ASPAI
MADEIRA • 2024

Advances in Signal Processing and Artificial Intelligence

Proceedings of the 6th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2024)

Edited by Sergey Y. Yurish

IFSA

ASPAI
MADEIRA • 2024

Advances in Signal Processing and Artificial Intelligence:

Proceedings of the 6th International Conference on Advances in Signal Processing and Artificial Intelligence

17-19 April 2024
Funchal (Madeira Island), Portugal

Edited by Sergey Y. Yurish

IFSA Publishing

Sergey Y. Yurish, *Editor*
Advances in Signal Processing and Artificial Intelligence
ASPAI' 2024 Conference Proceedings

E-mail (for orders and customer service enquires): ifsa.books@sensorsportal.com

Visit our Home Page on https://sensorsportal.com/ifsa_publishing.html

ISBN: 978-84-09-60540-8
ISSN: 2938-5350
BN-20240411-XX
BIC: UYQ

Contents

Foreword

On behalf of the ASPAI' 2024 Organizing Committee, I introduce with pleasure these proceedings devoted to contributions from the 6th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2024), 17-19 April 2024, Funchal (Madeira Island), Portugal.

Advances in artificial intelligence (AI) and signal processing are driving the growth of the artificial intelligence market as improved appropriate technologies is critical to offer enhanced drones, self-driving cars, robotics, etc. Today, more and more sensor manufacturers are using machine learning to sensors and signal data for analyses. The machine learning for sensors and signal data is becoming easier than ever: hardware is becoming smaller and sensors are getting cheaper, making Internet of things devices widely available for a variety of applications ranging from predictive maintenance to user behavior monitoring. Whether we are using sounds, vibrations, images, electrical signals or accelerometer or other kinds of sensor data, we can build now richer analytics by teaching a machine to detect and classify events happening in real-time, at the edge, using an inexpensive microcontroller for processing - even with noisy, high variation data.

The global artificial intelligence market size was valued at US $ 136.55 billion in 2022 and is projected to expand at a compound annual growth rate (CAGR) of 15.83 % from 2023 to 2030 to rich US $ 738.80 bn. Artificial intelligences currently transforming the manufacturing industry. Virtual reality, automation, Internet of Things (IoT), and robotics are some important features of AI that are benefitting the manufacturing industry. AI has been one of the fastest-growing technologies in recent years. The market growth is mainly driven by factors such as the increasing adoption of cloud-based applications and services, growing big data, and increasing demand for intelligent virtual assistants. The major restraint for such market is the limited number of AI technology experts.

The Series of ASPAI Conferences have been launched to fill-in this gap and to provide a forum for open discussion and development of emerging artificial intelligence and appropriate signal processing technologies focused on real-word implementations by offering Hardware, Software, Services, Technology (Machine Learning, Natural Language Processing, Context-Aware Computing, Computer Vision and Signal Processing). The goal of the conference is to provide an interactive environment for establishing collaboration, exchanging ideas, and facilitating discussion between researchers, manufacturers and users. The first ASPAI conference has taken place in Barcelona, Spain in 2019, the second (2020) and the third (2021) – in the virtual format due to the COVID-19 pandemic – in Berlin (Germany) and Porto (Portugal). In 2022 and 2023 we have returned to the in-person conference format in Corfu (Greece) and Tenerife (Canary Islands), Spain, respectively.

The conference is organized by the International Frequency Sensor Association (IFSA) - one of the major professional, non-profit association serving for sensor industry and academy since 1999, in technical cooperation with the media partners – IOS Press (journal *'Integrated Computer-Aided Engineering'*); World Scientific (*International Journal of Neural Systems*) and MDPI (journals 'Algorithms' and 'Electronics'). The conference program provides an opportunity for researchers interested in signal processing and artificial intelligence to discuss their latest results and exchange ideas on the new trends.

I hope that these proceedings will give readers an excellent overview of important and diversity topics discussed at the conference.

We thank all authors for submitting their latest work, thus contributing to the excellent technical contents of the Conference. Especially, we would like to thank the individuals and organizations that worked together diligently to make this Conference a success, and to the members of the International Program Committee for the thorough and careful review of the papers. It is important to point out that the great majority of the efforts in organizing the technical program of the Conference came from volunteers.

Prof., Dr. Sergey Y. Yurish
ASPAI' 2024 Conference Chairman

(003)

End-user Confidence in Artificial Intelligence Predictions

Z. Kam [1] L. Peraccio [2] and G. Nicora [2]
[1] Weizmann Institute of Science, Rehovot 76100, Israel
[2] University of Pavia, 27100 Pavia, Italy
Tel.: +972 545303136, +39 3381106288
E-mails: zvi.kam@weizmann.ac.il, giovanna.nicora@unipv.it

Summary: Scientific measurements always include their error-bars. This is not the practice for predictions given by Artificial Intelligence algorithms (AI). Thus, the credibility of their outputs is not known. We describe here a compact array structure, which is calculated once from the data set used for training supervised machine-learning-based applications. Discretization of feature values and histogram-binning of the multi-dimensional feature space allow to count the training events in each bin. Bin-counts provide directly readable estimate to the density of cases similar to each user-introduced test case, thus assigning a level of confidence to the AI prediction, and alerting users for badly supported prediction to outlier test cases.

Keywords: Artificial intelligence, Supervised machine learning, Training data, Error estimate, Credibility of AI predictions.

1. Introduction

An ever-increasing number of applications offered to the public are based on Artificial Intelligence (AI) algorithms. Supervised machine-learning-based applications (SML) are first trained on set of cases, each case consists of an array of features quantifying the case conditions, with the corresponding class. After this training phase, these applications output predicted class for each newly presented test case. The larger and more diverse the training set of cases is, the better the SML applications are expected to function. However, training cases include largely data from common conditions. When exposed to test inputs yet unseen or under-represented in the training set, SML generalization is poor and SML is bound to be fooled, yet always outputs a prediction. During the training phase of SML application development, evaluation of error statistics helps optimization and demonstration of its validity. We argue that also at the user-application phase, in addition to the prediction, there is need for an estimate of error probability, in order to assign a confidence level and alert for weakly-supported predictions.

2. The Algorithm

Nicora et. Al. [1] have recently reviewed methods for evaluating errors for AI algorithms output. SML prediction errors are derived from two properties of the set of training cases, namely, 1. The number of training cases similar to the test case (density of training cases); 2. Proximity to decision-borders. Low density implies that the SML application was not trained by a sufficient number of similar cases, and therefore what is named "Reliability" [2] of the prediction offered may be low. Proximity to borders means that small changes in the test case feature values can alter the prediction, which is named low "Local-Fit" [3]. Nicora et.al. [1] demonstrated, with simulated and medical data sets, that these two properties provide excellent estimates to errors. Yet, as much as we know, there are few practical procedures applicable at the point-of-use for estimating Reliability and Local-Fit for an individual test case. The training data are often too bulky to deliver to the end-user, even when they are open. In addition, users "out in the field" do not have the computer resources and the time to execute error estimation algorithms even when the training set and the algorithms are provided. We propose here a practical and compact array structure that allows user to directly estimate predictive reliability for each test case presented to a SML application.

2.1. Reliability

A continuous probability distribution function can be approximated by a discrete histogram. Repeated trials, with values following the probability distribution, increment a histogram bin counts, when the values fall within bin minimum and maximum ranges (bin edges). Similarly, a multi-dimensional histogram can count the number of training cases that fall, according to the values of all their features, within bin hyper-volume edges, and approximate their local density. Dougherty et al. [4] introduced discretization of continuous feature space in bins, for use by AI algorithms. Here we export the multi-dimensional histogram bins array to the end-user for directly reading the density of cases similar to the input test case.

Following is the formulation: The training set $S_n[k]$ includes K cases, and each case is defined by an array of N feature values:

$$S_n[k] \ (n = 1,N) \ (k = 1,K) \qquad (1)$$

Densities of the training data cases is approximated by counting the number of training cases that fall within discrete N-dimensional histogram bin edges defined as follows: We segment the values of each feature number n (n = 1,N) into M channels, each in a range between edges V_{nm-1} and V_{nm} (m = 1,M). For uniformly distributed feature values between 0 and 1, the edges can be, for example, linearly spaced:

$$V_{n,m} = m*DV \ (m = 0,M) \ DV = 1/M \quad (2)$$

Logarithmic edges can be defined by the equation:

$$V_{n,m} = ½^{(M-m)} \ (m = 1,M) \text{ and } V_{n0} = 0 \quad (3)$$

N-dimensional histogram bins, B[m[1]…m[N]] are defined by channel numbers m(n) for all n features and stored in an array B[mm], (mm = $1,M^N$), where bin index, mm, is derived from the channel numbers, m(n), and channel numbers can be uniquely calculated from bin index, since channel numbers are the digits of the bin index presented in base M:

$$mm = 1+\Sigma_{n = 1,N} \{ (m[n]*M^{(n-1)} \} \quad (4)$$

Bin-counts for B[mm] are incremented by a training case $S_n[k]$, if all feature values fall between the corresponding channel edges:

$$V_{n,m[n]-1} \geq S_n(k) > V_{n,m[n]} \ (n = 1,N) \quad (5)$$

Thus, the accumulated counts for the whole training set approximate the density of cases around the feature values, F_n, of each bin hyper-volume center:

$$F_n(m[1]…m[N]) = 1/2(V_{n,m(n)-1}+V_{n,m(n)}) \ (n = 1,N) \quad (6)$$

For a test cases $T_{m'(n)}$ the corresponding channel numbers m'[n] are similarly derived by (5) and bin index by (4), thus directly addressing N-dimensional histogram bin counts, approximating the density of training cases similar to the test case, and hence estimating Reliability of output prediction for this test case input.

For N = 10-dimensional feature space, and values segmented into M = 10 channels, $M^N = 10^{10}$ = =10 Gb array for the bins is required, which may be a bit heavy to handle. M = 6 channels require about 60Mbytes, easily handled by applications.

The number of channels may be optimized for each input feature, depending on its characteristics. Some features may be binary (exist or absent, such as a genetic mutation), while others may be quantified at higher number of channels, spread linearly, logarithmically (such as protein expression levels, typically quantified by 2- or 10-folds) or other channel-spacings that match the probability distribution of each feature. Application of Principal Component Analysis dramatically compacts the dimension of the bin-counts array.

Multi-dimensional data used in training applications are often clustered. This property provides ways to compress the size of the multi-dimensional bin storage space, for example by Sparse Matrix Presentation [5], increasing the number of features that can be analyzed simultaneously.

2.2. Local-fit

Decision-borders proximity to bin hyper-volume can be evaluated by counting the number of altered predictions for neighbors to bin hyper-volume center with features calculated by local variations [3]. If all neighbors produce a consistent prediction, Local-Fit is scored high. The larger the number of neighbors yielding an altered prediction, indicating close-by decision-borders, the less Local-Fitted is the prediction. The alterations are scored once in the multidimensional bin array and made available to the end-user via the same structure as for the Reliability.

4. Conclusions

Supplementing multi-dimensional bin array structure to SML applications provides a directly readable Reliability and Local-Fit estimate to each end-users test case input, adding to the output a level of confidence. This would gain faith to AI-based applications, presently considered by the public to be magical black boxes.

References

[1]. G. Nicora, M. Rios, A. Abu-Hanna, R. Bellazzi, Evaluating pointwise reliability of Machine learning prediction, *Journal of Biomedical Information,* Vol. 127, 2022, 103996.

[2]. S. Saria, A. Subbaswamy. Tutorial: safe and reliable machine learning, ArXiv, http://arxiv.org/abs/ 1904.07204.

[3]. J. A. Leonard, M. A. Kramer, L. H. Ungar, A neural network architecture that computes its own reliability, *Computers & Chemical Engineering*, Vol. 16, Issue 9, 1992, pp. 819-835.

[4]. J. Dougherty, R. Kohavi, M. Sahami, Supervised and unsupervised discretization of continuous features, in *Proceedings of the 12th International Conference on Machine Learning,* Tahoe City, California, July 9-12, 1995, pp. 194-202.

[5]. W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Sparse linear systems, numerical recipes in C, Chapter 2.7, in The Art of Scientific Computing, Third Edition, *Cambridge University Press*, 2007.

(004)

Cuffless Estimation of Arterial Blood Pressure Based on Heart Pulse Transmission Parameters Determined from Multi-channel PPG Signals

J. Přibil, A. Přibilová and I. Frollo
Institute of Measurement Science, Slovak Academy of Sciences, 841 04 Bratislava, Slovakia
Tel.: + 421-2-59104543
E-mail: jiri.pribil@savba.sk

Summary: The paper describes an experiment with indirect cuffless estimation of arterial blood pressure (ABP) from two/three-channel photoplethysmography (PPG) signals. It is important when the actual ABPs cannot be measured, e.g. during scanning inside a magnetic resonance imager. The proposed procedure uses heart pulse transmission parameters (HPTPs) extracted from the second derivative of the PPG signal. The linear regression method is used to calculate the relation between the determined HPTPs and the ABPs measured in parallel by a blood pressure monitor. The ABP values are estimated by the inverse conversion characteristic calculated from these linear relations. Final estimation errors obtained from this first-step experiment achieve acceptable values of -2.6/-3.5 mm Hg for systolic/diastolic ABPs.

Keywords: Arterial blood pressure, Estimation by linear regression, Heart pulse transmission parameters, Photoplethysmography signal.

1. Introduction

Persons examined in a magnetic resonance imager (MRI) are exposed to noise and vibration causing them stress that manifests mainly by heart rate (HR) and arterial blood pressure (ABP) changes [1]. The HR changes can be detected from a photoplethysmography (PPG) signal, while the systolic/diastolic blood pressure (SBP/DBP) values are measured by a blood pressure monitor (BPM) [2]. However, this type of a measurement arrangement is less comfortable for tested persons, and it causes problems with practical realization of experiments. In addition, the BPM device cannot be used for measurement inside the scanning area of a running MRI device due to interaction with a working magnetic field and a strong RF disturbance. It is well known that the second derivative PPG wave (SD-PPG) [3] consists of five areas corresponding to the time domain features [4]. These parameters are then fed into the inference function of a regression version of the least squares support vector machine algorithm [5]. The estimation error of this method typically achieves about 5-10 % [6].

The precision of ABP estimation may be improved by the method based on heart pulse transmission parameters (HPTP). Originally, the pulse transmission time (PTT) represented the time difference between R peak of the electrocardiogram and the systolic peak of the PPG measured by sensors located at a known distance [7]. The PTT can be also determined from two or more PPG waves sensed in parallel [8]. Another parameter describing current state of a human cardiovascular system of a tested person is the pulse wave velocity (PWV). We present also usefulness of derived parameters: relative PTT (rPPT) and relative PWV (rPWV).

The main motivation of this work was to test whether these HPTPs are suitable for ABP estimation and whether they give sufficient estimation accuracy. This paper describes the procedure for HPTPs determination from the preprocessed two/three-channel SD-PPG signals. The current experiments use two small databases of PPG records collected in the frame of our previous research [9-10]. The linear regression method is used to perform ABP estimation from the HPTPs. The numerical comparison based on a relative estimation error (REE) is performed to verify estimation accuracy of SBP/DBP values. Partial results determined separately from two used databases were compared by the scatter plots mapping correlations between the measured and estimated ABP values. Final estimation errors for both databases together were also graphically evaluated using the Bland-Altman plots.

2. Methods

The proposed method of SBP and DBP values estimation from HPTP parameters determined from the multi-channel PPG signal records can be divided to four phases:

1. Creation of a database of the HPTPs from the pre-processed SD-PPG signal records together with the HR and ABP values measured in parallel;
2. Application of a linear regression method to find a linear relation between the determined HPTPs and the HR/ABP_{BPM} values;
3. Calculation of inverse conversion characteristics for estimation of the SBP/DBP values from the HPTP parameters;

4. Testing the correctness and evaluation the precision of the proposed estimation method (see the block diagram in Fig. 1).

The algorithm used for HPTP values determination is relatively simple – but stable, and sufficiently precise. The PPG signal analysis after pre-processing phase starts with the systolic peaks P_{SYS} localization procedure. Next, the heart pulse period T_{HP} and the pulse amplitude (Ap) are determined from the PPG wave signal – as demonstrated in Fig. 2. The PTT and other derived parameters are next calculated from the difference ΔP_{SYS} in samples between adjacent P_{SYS} positions of two/or more PPG waves (see an example in Fig. 3). Using the sampling frequency f_s in kHz the PTT in ms is determined as

$$PTT = \Delta P_{SYS} / f_s \tag{1}$$

The pulse wave velocity represents the relationship between the PTT and the measuring distance Dx

$$PWV = Dx / PTT \tag{2}$$

The relative parameter rPTT defined as a percentual ratio

$$rPTT = (PTT / T_{HP}) \times 100 \tag{3}$$

is invariant on the current HR value in beat per minute (bpm), which can be calculated as

$$HR = 60 \times f_s / T_{HP} \tag{4}$$

Like to (3), the relative PWV is defined as

$$rPWV = Dx / rPTT \tag{5}$$

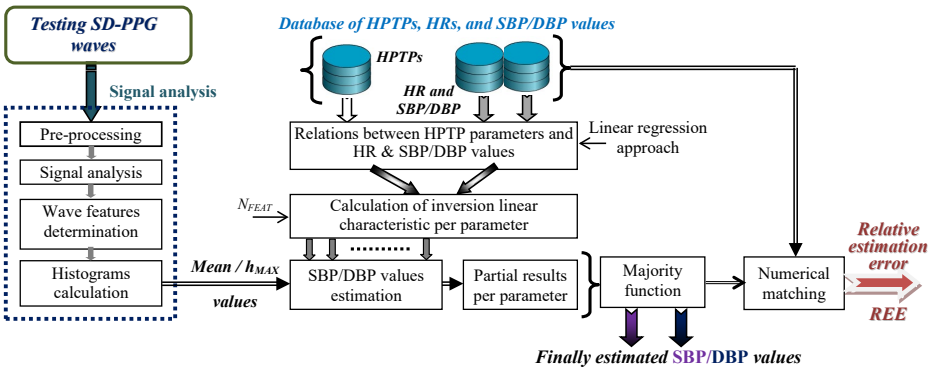


Fig. 1. Block diagram of the method for testing the correctness and evaluation the precision of estimated SBP/DBPs.

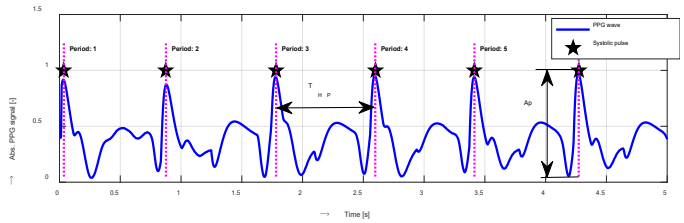


Fig. 2. An example of 5-sec part of a PPG wave
with localized systolic heart peaks (with amplitudes Ap) and determined heart pulse periods T_{HP}.

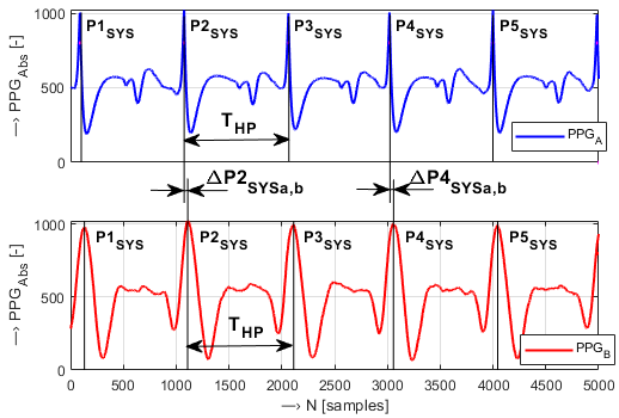


Fig. 3. An example of determining the time differences between systolic pulses $\Delta P_{SYSa,b}$ from two PPG waves sensed in parallel (PPG_A and PPG_B); f_s = 1 kHz.

HPTPs determined in this way are then statistically processed to obtain representative values (one per the whole analyzed PPG signal record) for further use in the estimation process. The simplest method for calculation of the representative value is to use the mean value. A situation often occurs in which the distribution of the analyzed parameter has a non-Gaussian character, so the mean value will not provide a correct result. In this case it is better to determine the value h_{MAX} corresponding to the maximum occurrence o_{MAX} [%] in the histogram however, o_{MAX} must be relatively high (typically more than 25 % – see the left graph in Fig. 4. Otherwise, better precision is achieved by calculation using the mean method (see the right graph in Fig. 4).

All these HPTPs are used to find a linear relation with BP_{BPM} in mmHg measured in parallel by the BPM device to obtain linear regression characteristics – see an example in Fig. 5. Next, the inverse conversion characteristic is calculated to estimate ABP values from the used HPTPs. Finally, the SBP/DBP values are estimated by applying the majority function (one set of values per one tested PPG record). For numerical evaluation of estimation results the relative percentage error is used

$$REE_{ABP} = (ABP_{EST} - BP_{BPM})/BP_{BPM} \times 100, \qquad (6)$$

where ABP_{EST} represents the estimated SBP/DBP and BP_{BPM} is the real value measured. For the purpose of visualization of the estimation error a simple absolute difference $\Delta SBP/DBP = ABP_{EST} - BP_{BPM}$ is applied. These differences are necessary for creation of the Bland-Altman plots and for mapping correlations between the measured and estimated ABP values based on the scatter plots.

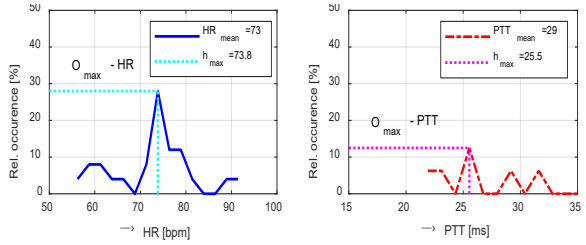


Fig. 4. Histograms of HR and PTT parameters demonstrating different maximum occurrence o_{MAX}.

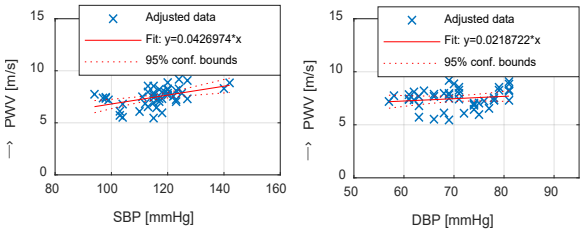


Fig. 5. Fitted linear relations between determined PWV and measured SBP/DBP values.

3. Material, Experiments, and Results

Two small PPG signal databases were used in this work: the first PPG corpus (DB_1) consists of two-channel PPG signals originated from 7 male and 3 female volunteers (aged from 22 to 60 years) [9]. The second PPG corpus (DB_2) contains three-channel PPG signals from 12 subjects (8 males and 4 females, with a mean age of 50 years) [10]. All PPG signals were picked with the help of special prototypes of wearable PPG sensors based on the Arduino micro-controller board with the processor ATmega328P and using the Pulse Sensors Adafruit 1093 working in a reflectance mode and directly producing the SD-PPG wave as an output. Data transfer to the control device (laptop, tablet, etc.) was realized via Bluetooth (BT) serial connection working in the 4.1 standard at 4.2 GHz [9], [10]. In the case of PPG records included in the DB_1 corpus, the first optical PPG sensor was always placed on the wrist artery and the second one was worn successively on each of the fingers of the left/right hand (P1-P5). For PPG records from the database DB_2 holds that the first optical sensor was again placed on a wrist, the second one on a pinkie (P1), and the third one on a forefinger (P4). A typical duration of each PPG signal record was 64 sec, so about 60÷80 PPG cycles can be localized and a similar amount of PPG features can be determined (the first and the last cycle are ignored). The real number of PPG cycles depends on the current HR, was always sufficient to obtain stable and credible statistical results necessary for final successfulness of the whole estimation process.

Sensing of PPG signals in our experiments was accompanied by parallel measurement of BP/HR values by the portable BPM device (automatic blood pressure monitor BP-A150-30 AFIB by Microlife AG). To prevent any possible negative influence of an inflated pressure cuff of the BPM on a tested person's blood system, the PPG signal was picked up from the fingers of the opposite hand.

The proposed method of SBP and DBP estimation from the PPG signal works in four phases, as described in the previous section. The PPG signal processing and implementation of the whole estimation algorithm were realized in the Matlab environment (ver. 2019a).

Table 1 compares numerical results of the obtained *REE* separately for the databases DB_1 and DB_2, for particular HPTPs and for all parameters together.

Table 1. Mean REE percentage values per a HPTP type, separately for the used databases DB_1 and DB_2.

HPTP	SBP $_{DB1}$	DBP $_{DB1}$	SBP $_{DB2}$	DBP $_{DB2}$
PTT	-2.5±2.3	-3.6±2.4	0.1±9.4	2.9±10.1
PWV	-2.7±2.4	-4.2±2.8	0.1±9.6	3.3±9.8
rPTT	-3.5±2.3	-3.9±2.5	1.5±8.9	3.2±10.0
rPWV	-3.1±2.2	-4.0±2.6	2.1±8.8	3.5±10.1
all	-3.1±2.2	-3.9±2.6	0.9±9.0	3.2±9.9

Fig. 6 contains scatter plots showing the correlation between the measured and the estimated SBP/DBP values for the used databases DB_1 and DB_2 separately. Fig. 7 presents the Bland-Altman plots of the final estimation accuracy for SBP/DBP using data of both databases together.

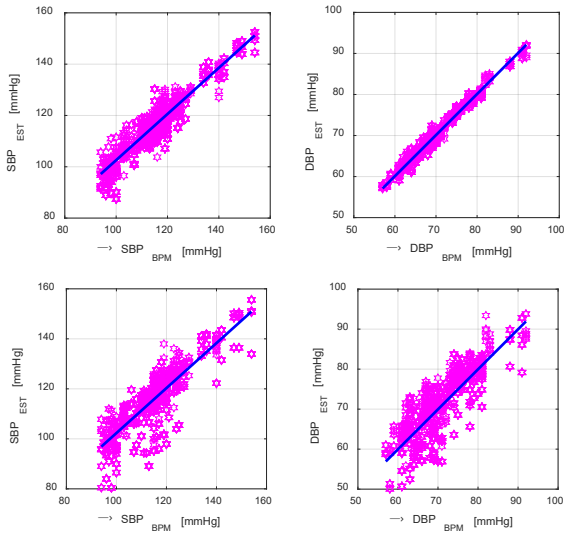


Fig. 6. Scatter plots of correlations between measured (SBP/DBP_{BPM}) and estimated (SBP/DBP_{EST}) values for: DB_1 (upper set of two graphs), and DB_2 (lower two graphs).

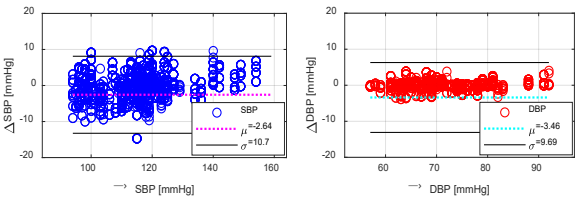


Fig. 7. Bland-Altman plots of the final estimation accuracy for SBP/DBP values, and merged DB_1, DB_2 databases.

4. Discussion and Conclusion

While f_s = 125 Hz is sufficient in sensing the PPG signals used for determination of basic features of PPG waves, higher $f_s \geq 1$ kHz must be applied to obtain correct values of HPTP parameters. Otherwise, the distances ΔP_{SYS} determined in samples can be too short which causes high errors in the time domain, i.e. heavy influence on the accuracy stability of the determined PPT and other derived parameters.

The preliminary analysis of mutual positions of HPTP and ABP/HR_{PPG} values performed prior to practical experiments have shown a detectable data grouping effect dependent on the position of PPG signal sensing (left/right hands) and the gender of tested subjects (male/female) as documented in graphs in Fig. 8. For this reason, common values obtained from all tested subjects, from PPG signals of all five fingers of both hands, without any differentiation were used in this work. Next simplification lies in the fact that the precision of *D*x distance measurement – which also affected the accuracy of the PWV and rPWV values – was omitted here. In addition, the quality of the sensed PPG signals depends essentially on the actual state of the skin at the place of an optical sensor. It means the skin surface temperature, but also the skin color and humidity together with the tested subject gender can have influence on the obtained PPG signal.

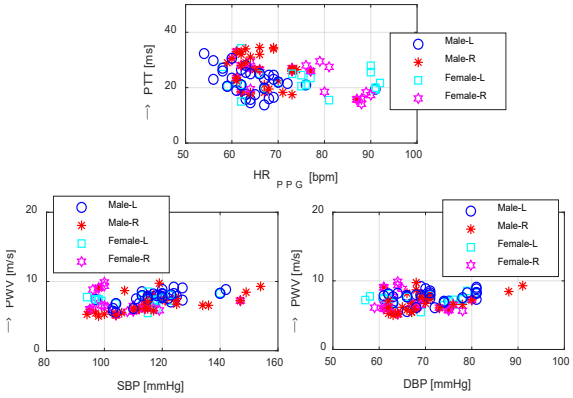


Fig. 8. Graphs of mutual positions grouped by subject's genders and left/right hands: $PTT\text{-}HR_{PPG}$ (upper graph) and PWV-SBP/DBP (lower two graphs) for database DB_1.

Therefore, in the near future, we plan to collect another PPG signal database with attached temperature values measured by a contact method. The applied thermo-element should be integrated directly to the optical sensor part to receive the current skin temperature at the place of sensor wearing (on a finger or a wrist).

From the point of view of the obtained ABP estimation results, the numerical comparison in Table 1 shows negative *REE* ($ABP_{EST} < BP_{BPM}$) and low std (up to 3 %) for DB_1. For DB_2, *REE* is positive, with higher std (more than 10 %). In both cases, the estimation errors are higher for DBP. The final mean estimation errors for both databases were following: ΔSBP = -2.6±10.7, ΔDBP = -3.5±9.7 mmHg. These results are acceptable for this first-step experiment, but further improvements are necessary before practical usage of the proposed method.

Acknowledgements

This work was funded by the Slovak Scientific Grant Agency project VEGA2/0004/23.

References

[1]. M. C. Steckner, A review of MRI acoustic noise and its potential impact on patient and worker health, *eMagRes*, Vol. 9, Issue 1, 2020, pp. 21-38.

[2]. P. Celka, et al., Influence of mental stress on the pulse wave features of photoplethysmograms, *Healthcare Technology Letters*, Vol. 7, 2020, pp. 7-12.

[3]. M. Nitzan, Z. Ovadia-Blechman, Physical and physiological interpretations of the PPG signal, Photoplethysmography: Technology, Signal Analysis, and Applications (P. A. Kyriacou, J. Allen, Eds.), *Elsevier*, London, UK, 2022, pp. 319-339.

[4]. M. Liu, P. Lai-Man, H. Fu, Cuffless blood pressure estimation based on photoplethysmography signal and its second derivative. *International Journal of Computer Theory and Engineering*, Vol. 9, 2017, 202.

[5]. J. A. K. Suykens, J. Vandewalle, Least squares support vector machine classifiers. *Neural Process Letters*, Vol. 9, 1999, pp. 293-300.

[6]. S. Mousavi, et al., Blood pressure estimation from appropriate and inappropriate PPG signals using A whole-based method, *Biomedical Signal Processing and Control,* Vol. 47, 2019, pp. 196-206.

[7]. M. Johnson, R. Jegan, X. Mary, Performance measures on blood pressure and heart rate measurement from PPG signal for biomedical applications, in *Proceedings of IEEE International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology (ICEEIMT'17)*, 2017, pp. 311-315.

[8]. C. O. Manlises, et al., Monitoring of blood pressure using photoplethysmographic (PPG) sensor with aromatherapy diffusion, in *Proceedings of the 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE'16)*, Penang, Malaysia, 25-27 November 2016, pp. 476-480.

[9]. J. Přibil, A. Přibilová, I. Frollo, Analysis of heart pulse transmission parameters determined from multi-channel PPG signals acquired by a wearable optical sensor, *Measurement Science Review*, Vol. 23, Issue 5, 2023, pp. 217-226.

[10]. J. Přibil, A. Přibilová, I. Frollo. Triple PPG sensor for measurement of heart pulse transmission parameters in weak magnetic field environment, in *Proceedings of the 14th International Conference on Measurement*, Smolenice Castle, Slovakia, 29-31 May, 2023, pp. 67-70.

(005)

Porting Large Language Models to Mobile Devices for Question Answering

Hannes Fassold
JOANNEUM Research – Digital, Steyrergasse 17, 8010 Graz, Austria
Tel.: +43 316 876-1126
E-mail: hannes.fassold@joanneum.at

Summary: Deploying Large Language Models (LLMs) on mobile devices makes all the capabilities of natural language processing available on the device. An important use case of LLMs is question answering, which can provide accurate and contextually relevant answers to a wide array of user queries. We describe how we managed to port state of the art LLMs to mobile devices, enabling them to operate natively on the device. We employ the llama.cpp framework, a flexible and self-contained C++ framework for LLM inference. We selected a 6-bit quantized version of the Orca-Mini-3B model with 3 billion parameters and present the correct prompt format for this model. Experimental results show that LLM inference runs in interactive speed on a Galaxy S21 smartphone and that the model delivers high-quality answers to user queries related to questions from different subjects like politics, geography or history.

Keywords: Deep learning, Large language models, Question answering, Mobile devices, Termux.

1. Introduction

Large Language Models (LLMs) [1] on mobile devices enhance natural language processing and enable more intuitive interactions. These models empower applications such as advanced virtual assistants, language translation, text summarization or the extraction of key terms in the text (named entity extraction).

An important use case of LLMs is also question answering, which can provide accurate and contextually relevant answers to a wide array of user queries. For example, it can be used for fake news detection on a smartphone by querying the LLM about the validity of dubious claims made in a news article.

Because of the limited processing power of a typical smartphone, usually the queries for an LLM on a mobile device are processed in the cloud and the LLM output is sent back to the device. This is the standard workflow for the ChatGPT app and most other LLM-powered chat apps. But often this is not possible or desired, for example for journalists operating in areas with limited connectivity or under strict monitoring and surveillance of internet traffic (e.g. in authoritarian regimes). In this case, the processing has to be done on the device.

In this work, we therefore demonstrate how to port LLMs efficiently to mobile devices so that they run natively and in interactive speed on a mobile device. In the following section, we will describe the software framework we employ for running LLMs natively on a mobile device (like a smartphone or tablet). Section 3 describes the specific model we have chosen and the proper prompt format for it. In Section 4 we provide information about qualitative experiments with the LLM and Section 5 concludes the paper.

2. LLM Framework for On-device Inference

Initially, we tried to do LLM inference natively on a mobile device via the *TensorFlow Lite* (TFLite) framework, as it is the most popular framework for on-device inference. But for LLMs, practically all finetuned models available on the *Hugging Face model hub*[1] provide only PyTorch weights, so a conversion to TFLite has to be done. Unfortunately, during our experiments we noticed that the conversion pipeline is quite complex (PyTorch = > ONNX = > TensorFlow = > TFLite) and not future-proof, as legacy TensorFlow 1.X versions have to be used in combination with code from several public GitHub repos which are not maintained anymore.

We therefore opted for the *llama.cpp* framework[2], a very flexible and self-contained C++ framework for LLM inference on a variety of devices. It can run state of the art models like *Llama* / *Llama2* [2], *Vicuna* [3] or *Mistral* [4] either on CPU or GPU/CUDA and provides a lot of options (e.g. for setting temperature, context size or sampling method). It supports a variety of sub-8bit quantisation methods (from 2 bits to 6 bits per parameter), which is crucial for running models with billions of parameters on a smartphone with limited memory.

In order to build the C++ libraries and executables of the llama.cpp framework, a standard Linux build toolchain is needed consisting of a terminal (shell), command-line tools, CMake/Make, C/C++ compiler and linker and more. For *Android*, fortunately such a build toolchain is available via the *Termux* app[3]. It can be installed via the Android open-source software package manager F-Droid[4] and does not need root access to the device. It has been used already for deep

[1] https://huggingface.co/docs/hub/models-the-hub
[2] https://github.com/ggerganov/llama.cpp
[3] https://termux.dev/en/
[4] https://f-droid.org/

learning tasks, for example for on-device training of neural networks as described in [5].

After installation of Termux, we open a console there and install the necessary tools (like wget, git, cmake, clang compiler) via *pkg*, the Termux package manager. We install also the Android screen mirror software scrcpy[1] on the PC so that we can control the device directly on the PC and mirror its screen there.

For building the llama.cpp binaries, we now clone its latest sources from the respective GitHub repo. We invoke *CMake* to generate the Makefile and build all binaries via the *make* command. We compile the binaries with model inference done on the CPU, as GPU inference relies on CUDA which is not available on Android devices. After compiling, several binaries are available on the device. The most important ones are an executable for direct interactive chatting with the LLM and a server application with a REST-API which is similar to the OpenAI API for ChatGPT. The server application allows for further integration, for example into an on-device GUI app for question answering.

3. Model Selection and Prompt Format

On the Hugging Face model hub there are many pretrained large language models available, which differ in their network architecture, model size, training / finetuning procedure and dataset and their task (base model for text completion versus instruct model for instruction following and chat). After some experiments, we selected the *Orca-Mini-3B* model[2] with 3 billion parameters. It runs in interactive speed on a recent smartphone and provides decent responses to a user query due to finetuning via imitation learning with the Orca method [6]. We employ a quantized model with approximately 5.6 bit per parameter, which takes roughly 2.2 GB of CPU RAM on the device.

For an instruct model, it is important for a good performance of the model to use the same prompt format (system prompt, user prompt etc.) as was used for finetuning the model. For the Orca-Mini-3B model, this means that the prompt format has to be as shown in the following example:
"### System:\n You are an AI assistant that follows instruction extremely well. Help as much as you can.\n\n### User:\n What is the smallest state in India ?\n\n### Response:\n"

4. Experiments and Evaluation

We did a subjective evaluation of the selected model by testing its responses for user queries related to questions from different subjects like politics, geography, history and more. From the tests we can infer that the model provides accurate and faithful answers for most of the user queries. Of course, like every LLM it can hallucinate (provide false information) from time to time.

An example output of the LLM application for direct chat can be seen in Fig. 1. The LLM provides correct answers for questions from different domains. The output of the model is generated fast enough for an interactive chat on a Samsung Galaxy S21 smartphone.

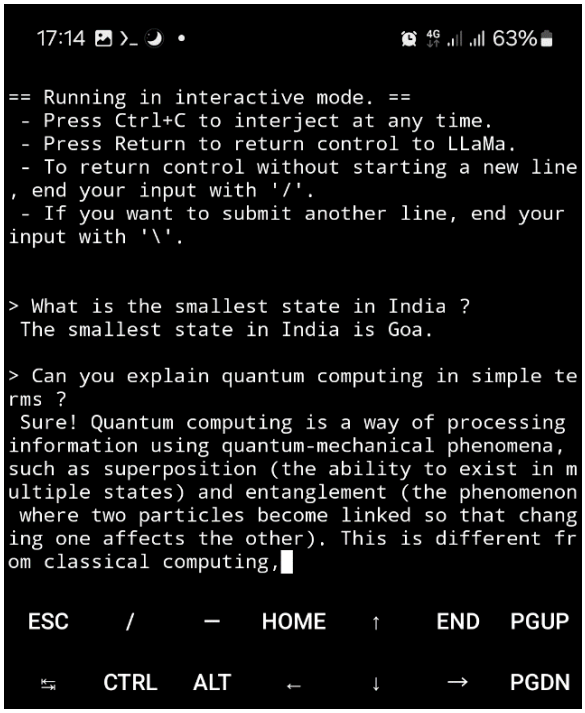


Fig. 1. Interactive chat application on a smartphone.

5. Conclusion

We demonstrated how to port large language models (LLMs) efficiently to mobile devices, so that they run natively on the device. We provided information on the LLM framework we employ as well as the model and proper prompt format for question answering. Experiments show that the model runs in interactive speed on a Galaxy S21 smartphone and provides high-quality answers for the user queries. In the future, we will explore recently introduced LLMs like *phi-2* [7] and GPU acceleration of the model via OpenCL or Vulkan on the device.

Acknowledgements

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 951911 – AI4Media.

[1] https://github.com/Genymobile/scrcpy

[2] https://huggingface.co/pankajmathur/orca_mini_3b

References

[1]. S. Minaee, T. Mikolov, et al., Large language models: a survey, *arXiv Preprint*, 2024, arXiv:2402.06196.

[2]. H. Touvron, L. Martin, Llama 2: open foundation and fine-tuned chat models, *arXiv Preprint*, 2023, arXiv:2307.09288.

[3]. L. Zheng, W. Chiang, et al., Judging LLM-as-a-Judge with MT-bench and chatbot arena, in Advances in Neural Information Processing System, Vol. 36, *Curran Associates, Inc.*, 2023, pp. 46595-46623.

[4]. A. Jiang, A. Sablayrolles, et al., Mistral 7B, *arXiv Preprint*, 2023, arXiv:2310.06825.

[5]. S. Singapuram, F. Lai, et al., Swan: a neural engine for efficient DNN training on smartphone SoCs, in *Proceedings of the IEEE Working Conference on Mining Software Repositories*, 2022.

[6]. S. Mukherjee, A. Mitra, et al., Orca: Progressive learning from complex explanation traces of GPT-4, *arXiv Preprint*, 2023, arXiv:2306.02707.

[7]. M. Javaheripi, S. Bubeck, et al., Phi-2: The Surprising Power of Small Language Models, https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models

(008)

Merging Outcomes of SAM Applied to RGB and Depth Images in Bin Picking Applications

M. Franaszek, P. Rachakonda, P. Piliptchak and K. S. Saidi
National Institute of Standards and Technology, Gaithersburg, MD 20899, USA
E-mail: marek@nist.gov

Summary: Segmenting images containing many objects stacked in unstructured piles is a challenging and vital task in robotic bin picking applications. Objects in such images are congested and occluded but, nevertheless, must be accurately segmented to calculate their 6DoF poses. For fast completion of automated tasks, many of these poses should be calculated and sent to the robot controller at once so that the path planning algorithm can prioritize which object to grasp. We show that the Segment Anything Model (SAM) can be used as the first step in processing such images to segment individual parts in a bin. However, applying SAM to RGB and depth images acquired from the same bin yields different results, with many segmentation masks present in only one type of image. Thus, merging two SAM outputs from both image types is suggested to maximize the number of segmented parts.

Keywords: Foundation models, Segment Anything Model (SAM), RGB and depth image, Robotic bin picking.

1. Introduction

Automated, robotic bin picking aims to pick up a part from an unstructured pile of parts in a bin and make it available for the next step of the automated process [1, 2]. This is usually achieved by using a robot with an integrated vision system. Data acquired by the vision system must be segmented so that the accurate 6DoF pose of the selected part from a bin can be calculated and provided to the path planning module of the robot controller.

Many different segmentation techniques that provide input for pose determination algorithms have been proposed [3-6]. Methods based on deep learning techniques require prior image annotations and training of a model [7, 8]. These tasks are not to be easily accomplished in low-volume and high-variability scenarios. Recent progress in the development of foundation models provides a chance to abandon these tasks, which are labor-intensive and require sufficient expertise [9]. While the segment anything model (SAM) [10] is agnostic to a particular part and still needs to be supplemented by some postprocessing, the difficulty of such an approach is expected to be much smaller than traditional labeling and training.

Fast completion of the picking task requires as many well segmented parts as possible. This enables the robot's path planner to select the best obstacle free path available, which increases the chances of successfully grasping a desired part. To maximize the number of accurately segmented parts in a bin, we merge two outcomes of SAM applied independently to RGB and depth images acquired from the same scene, as shown in Fig. 1. While many detections from both types of images overlap, we found that, on average, 40 % are observed only in one type of image.

Thus, merging two outcomes of SAM gives the robot's path planner more candidates to choose from.

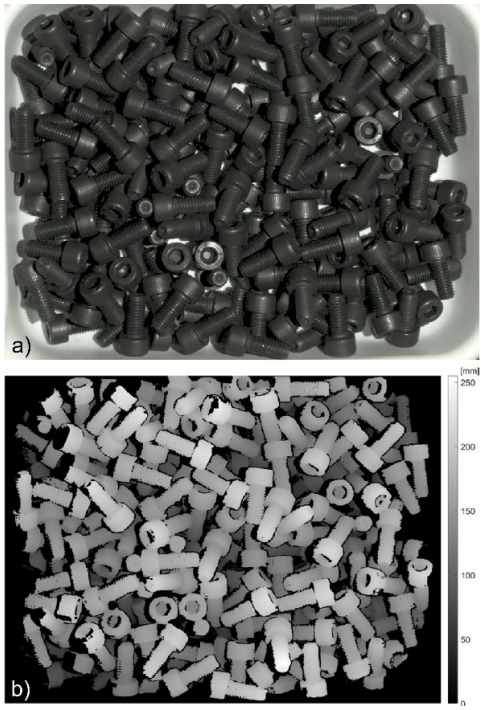


Fig. 1. Example of a) RGB; b) depth image.

2. Experiment

We used a structured light camera to acquire RGB images and the corresponding organized 3D point clouds of a bin filled with parts that are relevant in manufacturing (e.g., screws for which the CAD model was known). The organized point cloud format ensures that each of the three Cartesian coordinates (x, y, z) forms a 2D matrix of the same size as the corresponding RGB image. However, there is an essential difference between both types of images: while every pixel in an RGB image has three valid

entries for red, green, and blue components, some pixels in the organized 3D point clouds may not have entries and are labeled zero or not-a-number (NaN) as a placeholder. In Fig. 2, the RGB image and the corresponding binary map with the locations of NaN pixels are shown. For each 3D point cloud, the matrix of Cartesian z coordinates is converted into a grayscale depth image by replacing NaN pixels with z_{min}, i.e., the smallest of all valid z values.

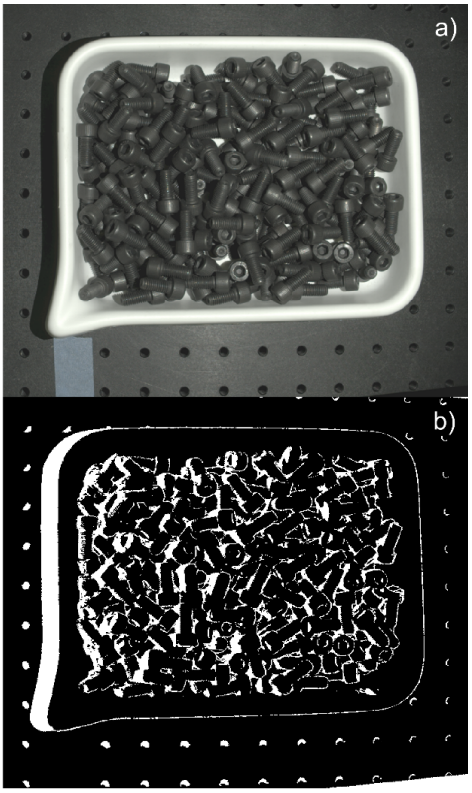


Fig. 2. Example of a) RGB image; b) binary map with locations of NaN pixels (white) in depth image.

Data from ten instances of random, unstructured piles of parts in the bin are acquired, and for each data, the RGB and depth images are processed by SAM; an example output for both types of images is shown in Fig. 3. For all ten acquired piles configurations, roughly the same total number N of segmented binary masks is observed for RGB and depth images, $N \approx 3{,}200$.

3. Data Postprocessing

Out of all N binary segmentation masks output by SAM, only less than a quarter are correct, as shown in Fig. 4. To filter them, the oriented bounding box is calculated for each subset of 3D points defined by a given mask, as shown in Fig. 5. Then, its normalized length L and width W are calculated, where

$$L = L_{EXP}/L_{CAD}\,, W = W_{EXP}/W_{CAD}, \qquad (1)$$

and EXP, CAD denote experiment and CAD model, respectively. Only length and width are used to filter correctly segmented masks. The third dimension (height H_{EXP}) depends strongly on the actual pose of the bounding box and is severely affected by noisy 3D points. The three dimensions $(L_{EXP}, W_{EXP}, H_{EXP})$ of bounding boxes in Fig. 5 are: a) $(25.8, 12.4, 5.3)$ [mm]; b) $(26.4, 12.6, 11.8)$ [mm]. In Fig. 6, the distributions of the normalized bounding box dimensions (L, W) are plotted. Three distinct clusters of points can be seen: cluster A is centered around $(L, W) \approx (1,1)$. Visual inspection of the corresponding masks output by SAM reveals they are in the category of correctly segmented masks, such as displayed in Fig. 4(d-f). The two other clusters marked in Fig. 6 correspond to incomplete segmentation masks: cluster B groups masks similar to those shown in Fig. 4b while cluster C groups masks displayed in Fig. 4c.

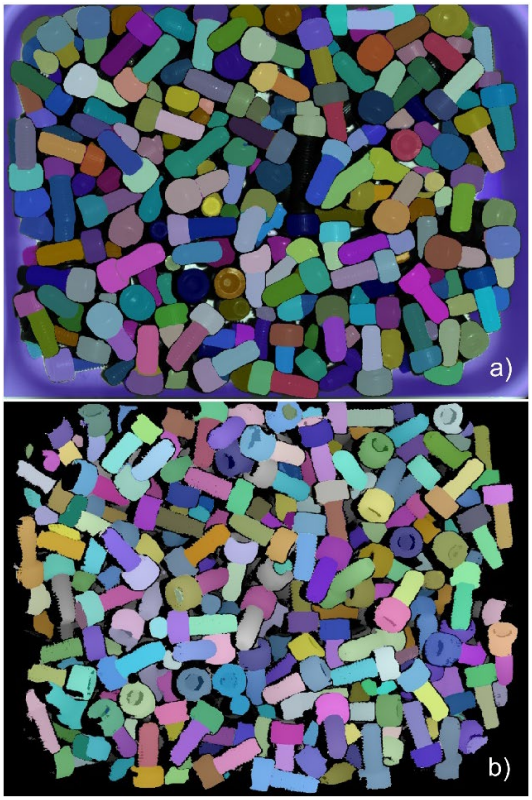


Fig. 3. Output from SAM applied to a) RGB; b) depth images shown in Fig. 1. Arbitrary color coding for both images.

For further processing, only the detections that passed the filter $\mathcal{F}$ were accepted, where

$$\mathcal{F}(L, W) = \begin{cases} 1 \; if \; L \in \tilde{\boldsymbol{L}} \text{ and } W \in \widetilde{\boldsymbol{W}}; \\ 0 \text{ otherwise}; \end{cases} \qquad (2)$$

and $\tilde{\boldsymbol{L}} = [0.904, 1.096]$, $\widetilde{\boldsymbol{W}} = [0.808, 1.077]$.

These steps are repeated for all ten RGB and depth images. The resulting detections are analyzed and

grouped into three categories, as shown in Fig. 7: a) detections found only in depth images; b) detections found in RGB images only; c) detections found in both RGB and depth images. A detection is declared common for RGB and depth images when the overlapping ratio of two binary masks is at least 80 %. In Fig. 8, the depth image with overlaid segmentation masks is shown for data corresponding to pile number $n = 7$.

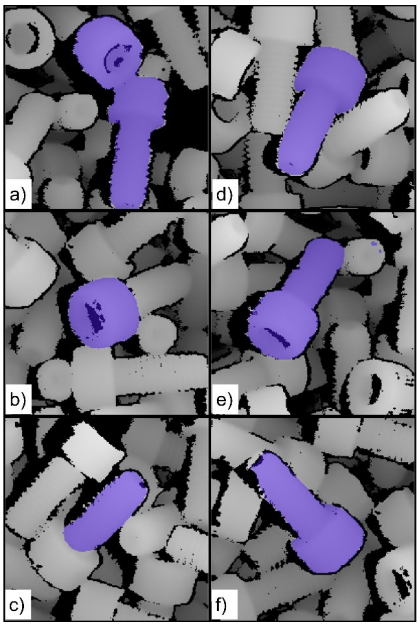


Fig. 4. Examples of individual masks output by SAM applied to depth images, left column (a-c): over- or under-segmented; right column (d-f): correct, complete.

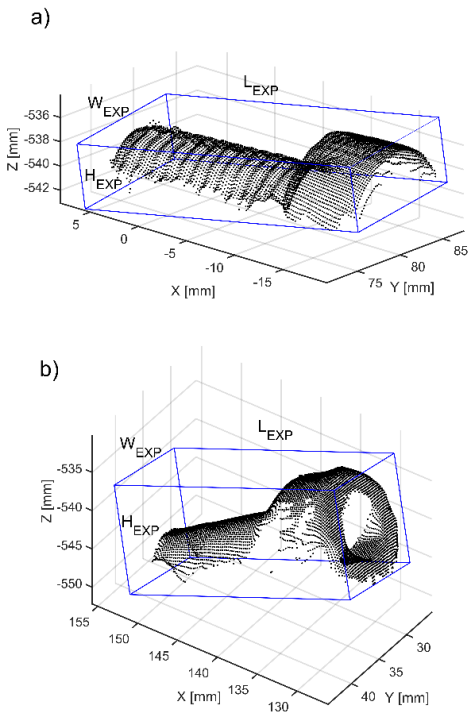


Fig. 5. Two examples of the segmented subset of 3D points and oriented bounding box.

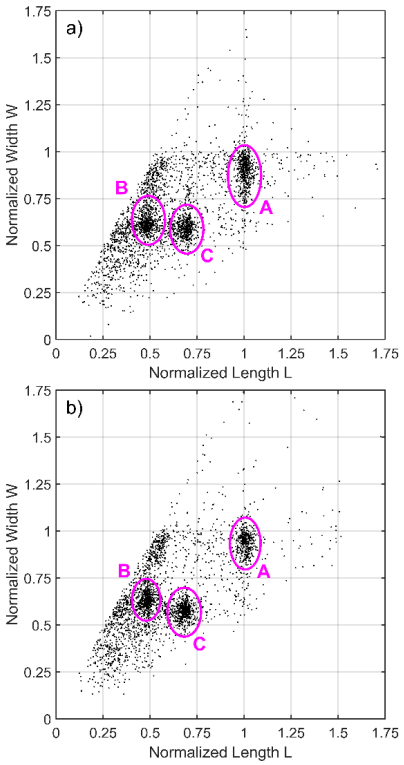


Fig. 6. Distribution of dimensions $(\boldsymbol{L}, \boldsymbol{W})$ for 3D points segmented by SAM applied to a) RGB; b) depth images.

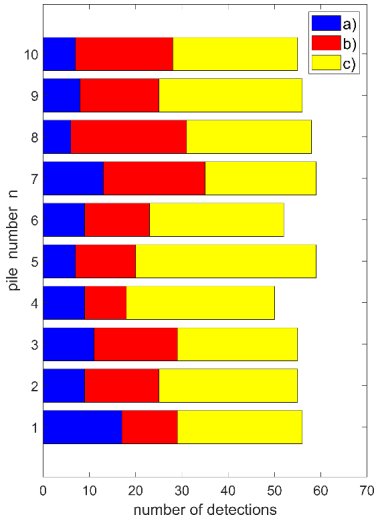


Fig. 7. Number of detections: a) in depth images only; b) in RGB only; c) in both RGB and depth.

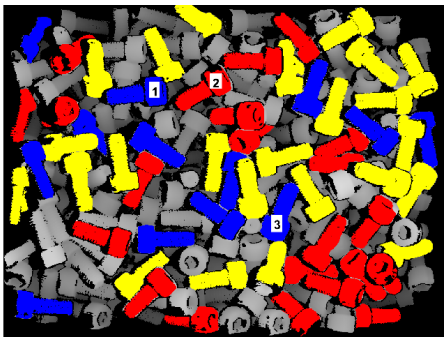


Fig. 8. Merged filtered SAM detections are overlaid on depth image, with color coding as in Fig. 7.

4. Discussion

Subtle differences between RGB and depth images of the same scene cause many SAM detections to be found in only one of the two images. Data in Fig. 7 indicate that, on average, segmentation of RGB only images yields 46 detections versus 39 for depth images only. When results from processing both types of images are merged, the average number of detections is 55.

To better understand why some segmentation masks are seen only in one type of image, a membership map $\mathcal{M}$ is created. It is a matrix of the same size as the original images processed by SAM, and each element in $\mathcal{M}$ stores the number μ indicating how many times a given pixel is a member of different segmentation masks. In Fig. 9, examples of membership maps are shown for RGB and depth images corresponding to the data displayed in Fig. 8. The largest observed value of the membership μ for this data is $\mu_{max} = 4$. Table 1 shows percentage of image area occupied by pixels with particular values of μ.

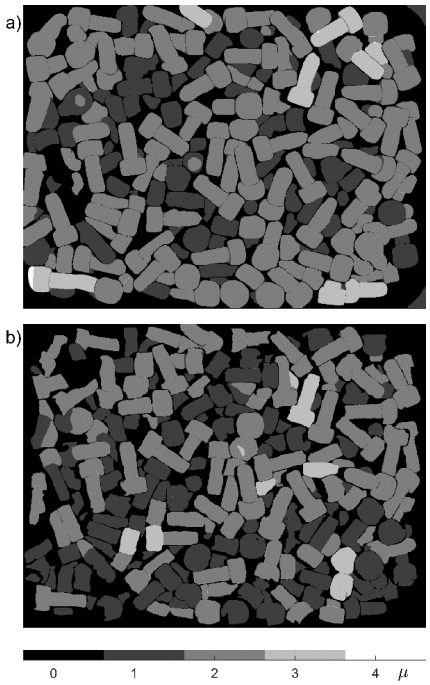


Fig. 9. Membership maps for output of SAM applied to: a) RGB; b) depth image.

With such different patterns in $\mathcal{M}$ maps observed for RGB and depth images, some complete segmentation masks are likely to be found in only one type of image.

Examples of such masks are marked in Fig. 8: detection 1 is reported only in the depth image, while detection 2 is found only in RGB image. In Fig. 10(a-c), the masks corresponding to detection 1 are shown, while in Fig. 10(d-f), the masks corresponding to detection 2 are shown. Note that the membership values $\mu = 1$ in Fig. 9a for individual segmentation masks shown in Fig. 10(a, b) and, similarly, $\mu = 1$ in Fig. 9b for masks displayed in Fig. 10(d, e).

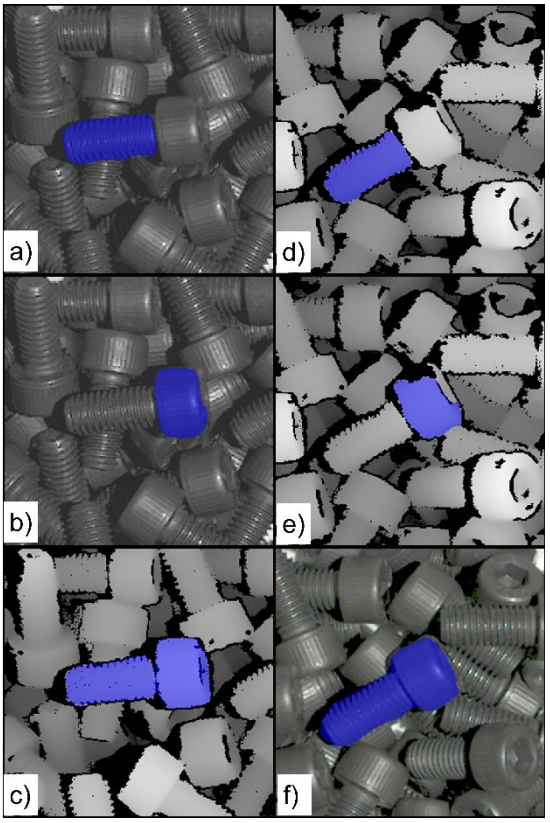


Fig. 10. Left column: examples of incomplete detection in RGB image in a) and b), and the complete corresponding detection in depth image in c). Right column: incomplete detections in depth image in d) and e) and the complete corresponding detection in RGB in f).

Table 1. Percentage of image area with given $\boldsymbol{\mu}$.

$\boldsymbol{\mu}$	0	1	2	3	4
RGB	23 %	20 %	51 %	5 %	< 1 %
Depth	32 %	27 %	37 %	3 %	< 1 %

For two complete masks shown in Fig. 10(c, f), the corresponding membership values $\mu = 2$ in Fig. 9. This means that in addition to complete masks for the selected parts, SAM also outputs two other incomplete masks, similar to those shown in Fig. 10(a, b) and Fig. 10(d, e).

In the described scenario, the correct, complete mask is present only for one type of image because SAM fails to provide complete masks for both RGB and depth images. However, there are also instances when SAM outputs complete masks for both images, but the detection is reported only for one type of image. This happens when the filter $\mathcal{F}$ in (2) accepts one detection but rejects detection from another image. An example of such detection, reported only for the depth image, is marked as 3 in Fig. 8. The masks for both images are displayed in Fig. 11 using two colors: blue color labels pixels belonging to the RGB mask in

Fig. 11a and to the depth mask in Fig. 11b. The yellow color in Fig. 11a indicates pixels that are also part of the RGB mask but are not in the depth mask. Similarly, the yellow color in Fig. 11b is used to indicate pixels belonging to the depth mask that are not part of the RGB mask. As can be seen, segmentation masks for both image types look complete and almost identical; their overlapping ratio is 95 %. However, only one of them, marked as 3, is displayed in Fig. 8 using blue color for detections present only in depth images.

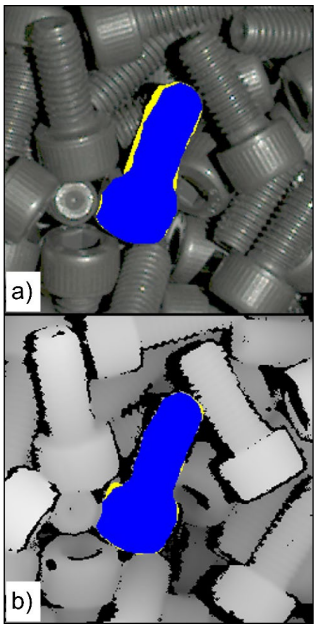


Fig. 11. Output of SAM applied to: a) RGB image; b) depth image. Color pixels mark segmentation masks.

As shown in Fig. 12, tiny differences in segmented masks visible in Fig. 11 are sufficient to generate two sets of 3D points for which the corresponding bounding boxes have quite different widths: $W = 1.12$ for box in Fig. 12a and $W = 1.0$ for box in Fig. 12b. Since the accepted range for width W in the filter $\mathcal{F}$ in (2) is set to $\widetilde{\boldsymbol{W}} = [0.808, 1.077]$, only the depth detection passes the filter as $W_{RGB} \notin \widetilde{\boldsymbol{W}}$ and $W_{Depth} \in \widetilde{\boldsymbol{W}}$.

A slight increase in the upper bound of the range $\widetilde{\boldsymbol{W}}$ would prevent this particular RGB detection from being rejected. Then, its color label would be changed from blue to yellow in Fig. 8 to indicate the detection reported for both types of images. However, filter $\mathcal{F}$ in (2) was introduced to eliminate most incorrect SAM detections, such as shown in Fig. 4(a-c). In bin picking applications, as many as possible accurately determined 6DoF poses of individual parts should be provided to the robot controller to select the best candidate for grabbing. This may suggest using in filter $\mathcal{F}$ more relaxed ranges $\tilde{\boldsymbol{L}}$ and $\widetilde{\boldsymbol{W}}$. However, the risk of passing incorrectly segmented 3D data (which will result in a failed attempt to grab the corresponding part) outweighs a loss of wrongly rejected mask, especially if the overall number of accepted masks is sufficiently large.

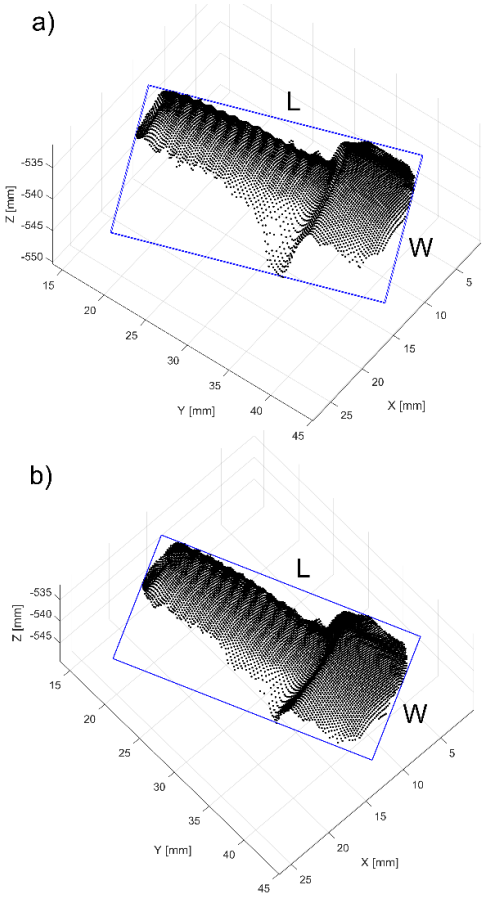


Fig. 12. Segmented 3D points with their respective bounding boxes for SAM masks obtained from: a) RGB image; b) depth image.

5. Conclusions

The organized 3D point cloud format enables SAM, originally developed for segmenting 2D images, to segment 3D data. Most segmentation masks are found in both types of images, but a portion of masks is reported in one type only: RGB or depth. Therefore, outputs from both types of images could be merged to maximize the number of detections.

In bin picking applications, segmentation of an individual part is needed to fit a CAD model and get a well-estimated 6DoF pose of a part. Fitting a model to 3D data is a rather time-consuming procedure that may fail if a starting pose for fitting is not well selected [11]. This usually happens when a subset of 3D points is not accurately segmented. For the techniques that rely on the segmentation of 2D images to get 3D points, the problem starts with either under or over-segmented 2D masks. Thus, having a feature that could gauge the quality of 2D segmentation before the slow model fitting procedure is invoked would be helpful. Since SAM provides slightly different outputs for RGB and depth images, exploring these differences may provide a useful hint about the quality of segmented 2D masks. That would reduce the cycle time in bin picking by avoiding attempts to fit a model to inaccurately segmented 3D points.

References

[1]. H. Alzarok, S. Fletcher, A. P. Longstaff, Survey of the current practices and challenges for vision systems in industrial robotic grasping and assembly applications, *Advances in Industrial Engineering,* Vol. 9, Issue 1, 2020, pp. 19-30.

[2]. A. Cordeiro, L. F. Rocha, C. Costa, P. Costa, M. F. Silva, Bin picking approaches based on deep learning techniques: A state-of-the-art survey, in *Proceedings of the IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC'22)*, 2022, pp. 110-117.

[3]. O. Skotheim, J. Thielemann, A. Berge, A. Sommerfelt, Robust 3D object localization and pose estimation for random bin picking with the 3DMaMa algorithm, *Proceedings of SPIE*, 2010, Vol. 7526, 75260E.

[4]. V. Kozak, R. Sushkov, M. Kulich, L. Preucil, Data-driven object pose estimation in a practical bin-picking application, *Sensors,* Vol. 21, 2021, 6093.

[5]. N. D. H. Nguyen, L. H. N. Nguyen, P.-T. Pham, Q. C. Nguyen, P. T. Ly, Bin-picking solution for industrial robots integrating a 2D vision system, in *Proceedings of the Int. Conference on High Performance Big Data and Intelligent Systems (HDIS'22)*, Tianjin, China, 2022.

[6]. P. Raj, L. Behera, T. Sandhan, Scalable and time-efficient bin-picking for unknown objects in dense clutter, *IEEE Trans. on Automation Science and Engineering,* 2023, pp. 1-13.

[7]. K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask R-CNN, *IEEE TPAMI,* Vol. 42, Issue 2, 2020, pp. 386-397.

[8]. F. Sultana, A. Sufian, P. Dutta, Evolution of image segmentation using deep convolutional neural network: a survey, *Knowledge-Based Systems,* Vols. 201-202, Issue 106062, 2020, pp. 1-25.

[9]. C. Zhou, *et al.*, A comprehensive survey on pretrained foundation models: a history from BERT to ChatGPT, *arXiv Preprint*, 2023, arXiv:2302.09419.

[10]. A. Kirillov, *et al.*, Segment anything, *arXiv Preprint*, 2023, arXiv:2304.02643,

[11]. P. J. Besl, N. D. McKay, A method for registration of 3-D shapes, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1992*,* Vol. 14, pp. 239-256.

(009)

Identification of Patients with Congestive Heart Failure Using K-Nearest Neighbors Technique and Wavelet Packet Decomposition

A. Hossen
Sultan Qaboos University, Electrical and Computer Engineering Department,
Communication and Information Research Center, 123 Muscat, Oman
E-mail: abhossen@squ.edu.om

Summary: Heart failure is a condition occurs if the heart becomes unable to pump enough blood. Congestive heart failure (CHF) is a special case of heart failure in which fluid builds up around the heart and in the lungs, causing congestion. Machine learning algorithms have been implemented for identification of patients with various diseases. In this paper, the K-nearest neighbors algorithm is used to identify patients with CHF from normal subjects. The data is taken from the MIT data base and splitted into two groups: trial group and test group. The frequency-domain features are obtained from the wavelet-based spectral of the heart rate variability (HRV) data. The KNN algorithm is trained on the trial group and the performance of the identification is obtained on the test group. The three well-known metrics: sensitivity, specificity, and accuracy are used to evaluate the performance of the system. Different factors are used to optimize the identification efficiency such as the wavelet filter type and the value of K (number of neighbors) in the KNN algorithm and the distance metric of the KNN algorithm. The best identification accuracy of 88 % is obtained using dmey wavelet filters with K equals 3 and with city block distance metric.

Keywords: Congestive heart failure, KNN, Wavelet packet decomposition, Identification, Frequency domain.

1. Introduction

1.1. Heart Failure

Heart failure is a common condition that usually develops slowly as the heart muscle weakens and needs to work harder to maintain a normal organ blood supply. It is often recognized at a more advanced stage of the disease, commonly referred to as Congestive Heart Failure (CHF). In which, failure of both left and right ventricles causes fluid to accumulate in the lungs, lower limbs, liver and sometimes the abdominal cavity [1].

According to the New York Heart Association (NYHA) system, which relates symptoms to everyday activities and the patient's quality of life, heart failure is classified into 4 classes [1]:

1. Class I (Mild), Symptoms with more than ordinary activities;
2. Class II (Mild), Symptoms with ordinary activities;
3. Class III (Moderate), Symptoms with minimal activities;
4. Class IV (Severe), Symptoms at rest.

The most important diagnosis test of heart failure is the Echocardiogram, which is a noninvasive technique using ultrasound to image the heart as it is beating in real time. It can determine the degree of failure, some of the causes and whether it is on the left ventricle, the right ventricle, or both [2]. The information from the Echocardiography is also used for calculating the ejection fraction (EF), which is the percent of the blood pumped out during each heartbeat. EF is a simple important measure for determining the severity of heart failure. People with a healthy heart usually have an EF of 50 percent or greater. Most people with heart failure, but not all, have an EF of 40 percent or less [2].

The echocardiogram is the most accurate diagnostic test but an expensive one. So, there is a need to a noninvasive simple test that helps in determining patients who most likely do not need an echocardiogram test [3].

1.2. Heart Rate Variability

Heart rate variability (HRV) is referred to as the beat-to-beat variation in heart rate. Instantaneous heart rate is measured as the time in seconds between peaks of two consecutive R waves of the ECG signal. This time is referred to as the RRI [3]. The variation of heart rate accompanies the variation of several physiological activities such as breathing, thermoregulation and blood pressure changes [3]. HRV is a result of continuous alteration of the autonomic neural regulation of the heart i.e. the variation of the balance between sympathetic and parasympathetic neural activity. The increase of sympathetic tone or decrease of parasympathetic activity will increase heart rate [3].

Several HRV abnormalities have been described in patients with CHF and it has been shown that patients with heart failure have decreased HRV [3].

Frequency-domain analysis approaches use one of the signal transformations such as FFT, wavelet transform to estimate the power spectral density of the RRI data. The frequency spectrum of the RRI data is divided into three main bands [3]:

- The very low-frequency band (VLF): $f \in (0.0033 - 0.04)\ Hz$;
- The low-frequency band (LF): $f \in (0.04 - 0.15)\ Hz$;

- The high-frequency band (HF): $f \in (0.15 - 0.4)\ Hz$.

A wavelet-decomposition with soft decision algorithm [4] is used to estimate an approximate power spectral density (PSD) of R-R-intervals (RRI) of ECG data for screening of congestive heart failure (CHF) from normal subjects [5]. The ratio of the power in the low-frequency (LF) band to the power in the high-frequency (HF) band of the RRI signal is used as the classification factor. Both trial and test data are drawn from MIT database [6]. The classification factor is determined from the trial data and then used to classify the test data to evaluate the performance of the technique. The receiver operating characteristics (ROC) is used to determine the threshold value of the classification factor. This technique showed a classification efficiency of 96.30 % on trial data and 88.57 % on test data using dmey wavelet filters. In this work machine learning algorithm as KNN is implemented to make the system fully automatic. The wavelet-packet, which is already present in Matlab toolboxes is to be used to simply the technique.

1.3. Data

The CHF records and normal records (NSR) were drawn from the Physionet website [6]. Two groups of CHF and NSR records are used as described below.

1.3.1. Trial Group

This group contains 15 CHF and 12 NSR records from MIT-BIH database. These records are used to train the machine learning algorithm.

The subjects of CHF are 11 men with age between 22 and 71 years, and 4 women with age between 54 and 63 years. The duration of each record is about 20 hours.

The subjects of the NSR records are 5 men, with age between 26 and 45 years, and 7 women with age between 20 and 50 years. The subjects were found to have no significant arrythmias.

1.3.2. Test Group

This group contains 17 CHF and 53 NSR recordings that are used to test the machine learning algorithm to find its performances. The subjects for the CHF records are 8 men, aged 39 to 68, and 2 women aged 38 and 59; gender is unknown for the 7 remaining records, but aged between 35 and 64 years.

The NSR data of this group contains 53 long-term (about 24 hours) RRI records. The subjects are 30 men, aged 28.5 to 76, and 23 women aged 58 to 73.

2. Methods

Wavelet-packet decomposition is used to decompose the RRI signal into approximation (a(n): low-pass component) and details (d(n): high-pass component) using a basic wavelet decomposition shown in Fig. 1. Both approximation and details can be divided more into smaller and smaller bands with a specified depth.

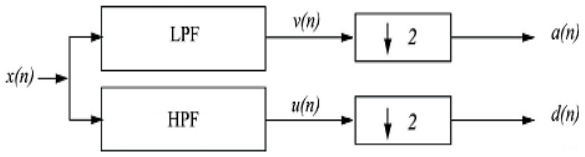


Fig. 1. Simple wavelet-decomposition.

In this paper, the number of decomposition stages is selected to be 5 yielding 32 bands, each covering a frequency range of 0.0156 Hz.

Three features: power of the (VLF, LF, and HF) bands can be obtained as below:

$$VLF = \sum_{i=2}^{i=3} P(B_i), \quad (1)$$

$$LF = \sum_{i=4}^{i=10} P(B_i), \quad (2)$$

$$HF = \sum_{i=11}^{i=25} P(B_i), \quad (3)$$

and then three ratios can be determined:

1. **r1** = The ratio of the power of LF band divided by the power of HF band **(LF/HF)**;
2. **r2** = The ratio of the power of VLF band divided by the power of LF band **(VLF/LF)**;
3. **r3** = The ratio of the power of VLF band divided by the power of HF band **(VLF/HF)**.

The KNN algorithm is trained on the trial group and the performance of the system is obtained from the test group. Different values of K and different types of distance metrics are applied in the KNN algorithm. Different wavelet filters are also implemented and their results are compared.

3. Results

Results are to be shown in terms of the three well-known metrics: sensitivity, specificity and accuracy [7].

The three metrics are defined in equations (4)-(6), and Table 1 defines the terms in the equations in a form of confusion matrix.

$$\text{Sensitivity (\%)} = TP \times 100/(TP+FN), \quad (4)$$

$$\text{Specificity (\%)} = TN \times 100/(TN+FP), \quad (5)$$

$$\text{Accuracy (\%)} = (TN+TP) \times 100/T, \quad (6)$$

where T is the total number of subjects in the test data, positive and negative are CHF and normal, respectively.

Table 1. Confusion Matrix.

	Predicted Class	
Actual Class	Positive (P)	Negative (N)
Positive (P)	TP	FN
Negative (N)	FP	TN

Table 2 shows the results of identification using different FFT-based features and the KNN algorithm with K = 3 and using city block distance metric. Tables 3-6 show the results for the wavelet-based features using the KNN algorithm (with K = 3 and with city block distance metric) and different wavelet filters (dmey, db4, coif3, and fk12), respectively.

Table 2. FFT results.

	r1 r2	**r1 r3**	**r2 r3**	**r1 r2 r3**
Sen.	94.11	88.23	94.11	88.23
Spe.	67.92	73.58	64.15	71.69
Acc.	74.28	77.14	71.42	75.71

Table 3. Wavelet (dmey) results.

	r1 r2	**r1 r3**	**r2 r3**	**r1 r2 r3**
Sen.	88.23	82.35	82.35	82.35
Spe.	73.58	90.56	84.90	79.24
Acc.	77.14	88.57	84.28	80.00

Table 4. Wavelet (db4) results.

	r1 r2	**r1 r3**	**r2 r3**	**r1 r2 r3**
Sen.	88.23	76.47	76.47	82.35
Spe.	75.47	83.01	79.24	77.35
Acc.	78.57	81.42	78.57	77.46

Table 5. Wavelet (coif3) results.

	r1 r2	**r1 r3**	**r2 r3**	**r1 r2 r3**
Sen.	88.23	76.47	76.47	82.35
Spe.	73.58	83.02	81.13	73.58
Acc.	77.14	81.42	80.00	75.71

Table 6. Wavelet (fk14) results.

	r1 r2	**r1 r3**	**r2 r3**	**r1 r2 r3**
Sen.	88.23	76.47	76.47	82.35
Spe.	75.47	83.01	83.01	73.58
Acc.	78.57	81.42	81.42	75.71

From Tables 2-6, it is to be concluded that the wavelet results are better than FFT and the features r1r3 are the best features. It can be also noticed that the best wavelet filter is dmey filter.

Table 7 shows the results for the dmey wavelet-based r1r3 features using the KNN algorithm at different K values.

Table 7. Results of different K values.

K	**Sen.**	**Spe.**	**Acc.**
3	82.35	90.56	88.57
5	82.35	79.24	80.00
7	82.35	88.67	87.14
9	82.35	86.79	85.71
11	82.35	86.79	85.71

It can be seen from Table 7 that K = 3 yields in the best result. Table 8 shows the results for the dmey wavelet-based r1r3 features using the KNN algorithm with the best K = 3 and using different distance metrics.

The best distance metric appears to be the city block followed by Euclidean distance and Minkowski distance metric.

Table 8. Results of different distance metrics.

Distance	**Sen.**	**Spe.**	**Acc.**
Cityblock	82.35	90.56	88.57
Chebychev	82.35	86.79	85.71
Euclidean	82.35	88.67	87.14
Minkowski	82.35	88.67	87.14

To test the consistency of the algorithm, interchanging of the trial data with test date is done.

Table 9 shows the result of this step for dmey filter with r1r3 as features and using city block distance metric and different K values. It is to be noted that the algorithm even performs better and the results are independent of data.

Table 9. Results of consistency test

K	**Sen.**	**Spe.**	**Acc.**
3	80	100	88.88
5	86.66	100	92.59
7	86.66	100	92.59
9	86.66	100	92.59
11	86.66	100	92.59

4. Conclusions

In this paper the wavelet-packet features of the RRI signal are used with the KNN machine learning algorithm to identify patients with CHF from normal subjects. Different wavelet filters are used in the system and different K values in the KNN algorithm are also implemented with different distance metrics of the KNN algorithm. The best identification accuracy obtained is 88.57 % using dmey wavelet filters with K = 3. The combination of features r1 and r3 gives the best result. This result is obtained using the KNN algorithm automatically with no need to use ROC or obtaining the threshold manually from the trial data.

This validates also the use of the wavelet-packet instead of the soft-decision wavelet decomposition algorithm. Consistency of the algorithm is also tested by interchanging the trial group with the test group. Results shows that even a higher accuracy of 92.59 % is obtained in this step for almost all K greater than 3.

References

[1]. Heart Failure Society of America, http://www.abouthf.org

[2]. Congestive Heart Failure, University of Maryland Medicine, http://www.umm.edu

[3]. Task force of the European society of cardiology and the North American society of pacing and electrophysiology, Heart rate variability, standards of measurements, physiological interpretation, and clinical use, *Circulation*, Vol. 93, 1996, pp. 1043-1065.

[4]. A. Hossen, B. Al-Ghunaimi, M. O. Hassan, Subband decomposition soft decision algorithm for heart rate variability analysis in patients with OSA and normal controls, *Signal Processing*, Vol. 85, 2005, pp. 95-106.

[5]. A. Hossen, B. Al Ghunaimi, A wavelet-based soft decision technique for screening of patients with congestive heart failure, *Journal of Biomedical Signal Processing and Control*, Vol. 2, Issue 2, 2007, pp. 135-143.

[6]. Physionet, Congestive Heart Failure RR Interval Database, http://www.physionet.org/physiobank/database

[7]. R. M. Rangayyan, Biomedical Signal Analysis: A Case-Study Approach, *John Wiley & Sons Inc*., U.S.A., Dec. 2001, pp. 466-472.

(011)

Hacking Visual Positioning Systems to Scale the Software Development of Augmented Reality Applications for Urban Settings

A. Giannakidis [1], M. Häcker [2], F. Sulzmann [1] and J. Frohnmayer [2]
[1] Fraunhofer Institute of Industrial Engineering IAO, Immersive Participation Lab
Nobelstr. 12, 70569 Stuttgart, Germany
[2] University of Stuttgart, Institute of Human Factors and Technology Management IAT
Nobelstr. 12, 70569 Stuttgart, Germany
E-mail: alexandros.giannakidis@iao.fraunhofer.de

Summary: Developing Augmented Reality (AR) applications for urban environments is notably labor-intensive. This is largely because verifying the functionality of these applications in their intended settings is required for developers, a process often repeated to minimize errors. Such a cycle not only extends the development timeline but also escalates the associated costs, affecting location-based AR projects in urban areas. Identifying optimal spots for AR markers presents its own set of challenges. To streamline development, the development team leverages Visual Positioning Systems (VPS), utilizing 3D models of urban landscapes as test environments. This approach allows for a preassessment of AR applications' performance and accuracy within a lab setting, using various mobile and wearable devices, thus eliminating the need for constant site visits and disruption of the development workflow.

Keywords: Augmented reality, Virtual reality, Visual positioning systems, Software development, Multi-stereo projection, Cave, convolutional neural networks, Digital twins, System hacking.

1. Introduction

Developing Augmented Reality (AR) software for urban settings typically involves numerous site visits to test the accuracy and performance of virtual objects within the real world. Each trip aims to identify and rectify errors, a cycle repeated until satisfactory results are achieved. This not only prolongs the development process but significantly increases its costs. By adopting a virtual testing environment, our team has seen considerable improvements in efficiency, enabling faster and more accurate development of AR applications.

2. Relevant Work

2.1 Augmented Reality Pose Estimation and Tracking

AR technology enhances our perception of the real world by overlaying virtual objects onto physical elements. Pose estimation and tracking determine the position and orientation of the user's viewpoint in relation to the virtual objects within a three-dimensional space. These processes ensures that 3D content is accurately anchored to the real world, providing users with convincing immersive experiences.

Recent advancements in AR technology have led to the development of sophisticated tracking algorithms that utilize RGB and depth sensors for inside-out tracking. These sensors, integral to AR glasses and handheld devices, analyze the surrounding environment to resolve accurate positioning of virtual objects [1]. For static environments, such tracking mechanisms have become standard. However, dynamic urban settings, where users or objects are in motion, present additional challenges.

One of the main challenges in AR pose estimation is maintaining accurate tracking in diverse and complex urban landscapes. Fast-moving users, such as those in vehicles, necessitate advanced tracking solutions that can adapt to rapid changes in the environment. To address this, researchers are exploring the integration of machine learning techniques with traditional sensor data [2]. These approaches aim to improve the robustness and accuracy of pose estimation, enabling more reliable AR experiences in a variety of settings.

2.2 Simulation of AR in VR

The use of virtual environments to simulate AR allows for experimentation and usability evaluation, provide complete control in the AR environment and advantages over testing with true AR systems. AR simulations have the potential to provide great benefit to AR research, allowing for the investigation of the effects of registration errors on task performance and accurate manipulation of augmented objects before deployment in real-life situations [3]. CAVE systems are ideal immersive environments for simulating AR scenarios and have been used with success to evaluate usability in applications or extend visualization space of graph visualizations [4]. Limitations of AR simulations in luminance fidelity may affect the replication of outdoor scenarios. Modern AR applications include the use of Light Detection and Ranging (LiDAR) to sense the environment. Such applications are not suitable for evaluating in CAVE systems because the user is actually surrounded by

physical walls and this results to a mismatch between the sensor data and the projected 3D scene.

2.3. Visual Positioning Systems

Visual Positioning Systems (VPS) enable the precise location of an object or user within a given space, primarily through the analysis of visual data. Unlike traditional positioning systems, which can rely on satellite signals (such as GPS) or radio frequencies (such as Wi-Fi or Bluetooth), VPS uses image recognition and computer vision techniques to understand the environment visually. This approach can be much more accurate in environments where GPS signals are weak or non-existent, such as indoors or in densely built urban areas.

The evolution of VPS has been significantly influenced by the adoption of Convolutional Neural Networks (CNN) and deep learning algorithms. These methods analyze complex visual data, allowing VPS to accurately identify features in the environment and determine the user's position relative to these features. The use of CNNs has also facilitated real-time 6D object pose estimation, which calculates the location and orientation of a device in three-dimensional space [5].

VPS function by comparing the view captured through a camera with a pre-existing database of images or 3D models of the environment. By identifying specific landmarks or features in the captured video-feed and matching them with its database, the system can pinpoint the precise location of the camera relative to its surroundings. Also, VPS is particularly useful in robotics applications that require situational awareness with high accuracy in navigation tasks [6].

VPS elevate the capabilities of AR by enhancing the precision of location and orientation tracking in visually complex environments. VPS is useful in urban settings, where GPS signal degenerates at the concrete of buildings and infrastructure. By employing VPS, AR applications can deliver more immersive and contextually relevant experiences. Users can receive navigation aids anchored directly onto their real-world view, interact with location-specific information, or engage with location-based gaming experiences. For example, Niantic Inc., creator of *Pokemon GO*, provides VPS technology for game developers to use in custom AR applications.
However, the challenges of pose estimation in dynamic, uncontrolled environments are significant. Factors like lighting changes, occlusions, and the need for real-time processing complicate the task. AR systems typically use a combination of sensor data (from accelerometers, gyroscopes, and sometimes depth sensors) and visual input from cameras to achieve this. Recent advancements leverage machine learning models, especially CNN, to improve the accuracy and speed of pose estimation, enabling more seamless and contextually relevant AR experiences, like assembly instructions [7].

VPS are commercially available and provide location-based services and Application Programming Interfaces (API) for developers to use in their own applications. Google uses its VPS in combination with *Google Maps* to provide AR walking directions through mobile phones. Their developer tool *Geospatial Creator* streamlines the creation of location based AR applications for indoor and outdoor environments.

3. Materials and Methods

3.1. Laboratory Setup (*AR Simulator*)

The AR software development process, especially in the Architecture Engineering Construction (AEC) domain, is characterized by frequent travels to the AR site, measuring accuracy and performance of superimposed graphical objects onto physical settings, registering errors and then going back to the lab where these errors are corrected in the code. This process repeats until a satisfying minimal error is achieved and evidently increases the software development overhead significantly. To reduce this overhead, we propose a virtual testing field for AR application development. To simulate AR in VR, the CAVE of the location-based Immersive Participation Lab (IPLab) at Fraunhofer IAO, is used for life-sized projections of photorealistic 3D models surrounding AR devices, as shown in Fig.1. This way, actual urban locations are visualized (as virtual locations), in a controlled laboratory setting.

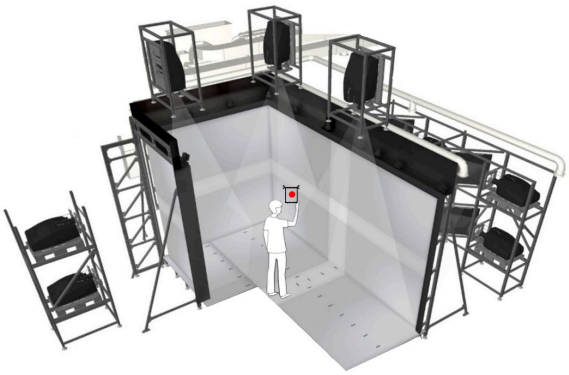


Fig. 1. CAVE of Immersive Engineering Lab at Fraunhofer IAO. Experimental setup to simulate AR Applications.

The IPLab is a work and presentation environment, which allows for accurately rendering immersive 3D graphics in real-time. Using VR, collaborative decision making is applied to a variety of industry sectors. The main component of the laboratory is a 3D projection system with a powerwall measuring 5.5 meters in length and 3.4 meters in height and a built-in four-sided CAVE. For these, 11 stereo projectors produce a picture of 25 million pixels rich in intensity and contrast, suitable for daylight conditions. Real-time visualization and a high-precision tracking

system allow users to immerse themselves in the virtual environment of life-size digital twins of urban areas and buildings. The big tracked space of this custom CAVE makes the IPLab suitable for communicating construction projects for the public and can host of up to 15 persons [8].

A single user is tracked and is presented with the correct perspective in 3D space. This presents a limitation to collaborative immersive experiences in groups and novel technologies are developed towards multiviewer CAVE systems at the Fraunhofer IAO [9]. This would also enable simulation of multiuser AR applications in surrounding projection systems.

A computer cluster of 11 PCs with 22 gaming GPUs is used for rendering up to 60 stereo images per second. These images are distributed to the 3D projectors using *UniCAVE,* a plugin that leverages the *Unity* game engine for non-head mounted virtual reality display systems [10]. This simplifies the process of adapting existing *Unity* projects for distributed visualization platforms and use modern gaming rendering technology in our CAVE, replacing complex workflows of custom outdated graphics engines used in the past.

The virtual environment, used to manipulate VPS localization in our experiments, is that of a visual digital twin of the Center for Virtual Engineering (Zentrum für Virtuelles Engineering, ZVE). This photorealistic 3D model has resulted from adapting the Building Information Modelling (BIM) model of the ZVE (that was deployed to actually construct the building in the past) for real-time rendering and interactive applications and is used as a testbed in a variety of projects and as an example of productive reuse of BIM Models.

3.2. Application Development and Testing

The introduced testing method exploits the inability of VPS to distinguish between reality and virtuality. For this research and to showcase the manipulation of Google's VPS using virtual environments, we developed two location-based mobile AR applications, one for indoor and one for outdoor experiments as is shown in Table 1. Both test applications are based on the Unity game engine and use *Google's ARCore* and *Geospatial Creator APIs* [11]. Additionally, we conducted an experiment using a commercial product that uses VPS technology.

Table 1. Experiments conducted to manipulate Google VPS in CAVE environments.

Experiment	API	Location	Application
1.	Geospatial Creator	Outdoor	Custom
2.	ARCore Cloud Anchor	Indoor	Custom
3.	Google Maps Live View	Outdoor	Commercial

We deployed those applications on latest generation AR-enabled iOS devices and tested them in the physical and virtual locations respectively (Table 2.). Each application has a red sphere as virtual object that appears at specific locations. In both cases (indoors and outdoors) we were able to simulate the AR experience in our CAVE. In parallel we documented our method with screen captures of applications running on devices and stills with the user inside the CAVE. The physical locations were chosen based on distinct characteristics and a short walking distance from the CAVE.

Table 2. AR devices used in experiments.

Device	Type	LiDAR	Experiment
1.	iPad Pro 6. gen	yes	1
2.	iPhone 13 Pro	yes	3
3.	iPhone 12	no	2

Location based virtual objects are commonly referred as anchors, cloud anchors, spatial anchors, persistent anchors, etc., and are stored either at the infrastructure of the VPS provider (in our case *Google Cloud*) or locally on users' devices.

3.3. Manipulation of Google's Geospatial Creator

The chosen outdoor location is at the front entrance of the Center for Virtual Engineering (Zentrum für Virtuelles Engineering, ZVE) which has a distinct architecture and also hosts the Immersive Participation Lab with the CAVE. This location also has sufficient VPS coverage, a precondition for *Google's Geospatial Creator* to function. VPS coverage is synonymous with *Google Maps and Street view* coverage. These two platforms provide the pre-existing database, i.e. the necessary data (terrain, images and 3D models of buildings, infrastructure or other landmarks) to compare against, during an AR-session and resolve the virtual objects (in our case a reflective red sphere with a diameter of 3 meters that floats 2 meters above ground) accordingly. In the first part of the experiment the AR-application is launched at about 20 meters distance of the chosen anchor points and after the localization process is completed, the virtual red sphere appears floating at the intended position in space. The localization process includes panning the AR device around in order to gather as much visual information about the environment as possible.

In the second step we terminate the application and relaunch it, but this time in the CAVE that displays a photorealistic 3D environment around us. Although no GPS signal is available inside the building, the localization process concludes with success and the virtual red sphere appears on the screen of the handheld AR device at the intended position. The CAVE environment in this case is fully utilized as it

allows for more than 180° pan movement of the AR device, often necessary for a successful localization of the VPS.

In Fig. 2, a side-by-side comparison of the physical and virtual location is shown. The left part is a screenshot of the AR application running on device 1 at the physical location. The right part is a screenshot of the same AR application running on device 1 inside the CAVE.

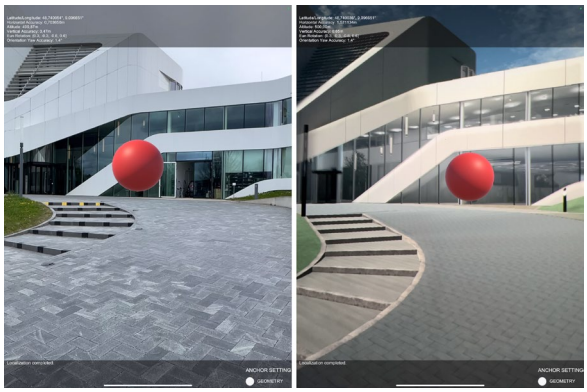


Fig. 2. Side by side comparison of outdoor experiment using Google Geospatial Creator. **Left** – device screen capture of resolved AR object in physical location. **Right** – device screen capture of resolved AR object in CAVE.

In the upper part of both screenshots localization information is displayed, like geolocation and orientation, with according error thresholds. The bottom part displays instructions and status updates. A first look at this comparison reveals good match between the actual and the simulated AR pose estimation. The red sphere appears at approximate the same position in front of the ZVE building. For this to succeed the AR device has to be carefully placed in the virtual environment resembling the position and orientation of the AR device at the physical location, as shown in the right part of Fig. 3.

Fig. 3. Experimental setup of user, AR device, tracking targets and CAVE projection of protorealistic 3D models. **Left** – indoor location using Google ARCore Cloud Anchors API. **Right** – outdoor location using Google Geospatial Creator.

Any mismatch results on slightly different AR pose estimations and final rendering of the virtual object. A closer look reveals these differences in pose estimation: the simulated AR pose estimation (Fig 2, Right) places the object with an offset of a few meters and also makes it appear smaller. Also, luminance is not the same. To accurately measure the offset of simulated AR experiences in combination with VPS, broader research should be conducted by measuring tracking errors.

3.3. Manipulation of Google's ARCore Cloud Anchor API

Google's ARCore Cloud anchor API also uses a VPS to map and resolve virtual objects in indoor locations. This VPS technology allows *ARCore* to recognize and understand the environment more accurately by comparing the visual features of a physical space to a previously mapped area stored in the cloud. By doing this, *ARCore* can determine the position and orientation of a user's device with high accuracy within environments that have been previously mapped and stored in the cloud. The *Cloud Anchors API* extends *ARCore's* environmental understanding by allowing developers to anchor virtual content to specific points in the environment that persist across app sessions and can be shared across devices. The visual positioning service supports these capabilities by recognizing specific features of the indoor environment and using them as reference points. The difference between the Geospatial Creator API, that uses references from existing databases, and the Cloud anchor API is that, in the latter, the user is required to map the environment manually first before being able to store a spatial anchor in the cloud. This process during an AR session is called mapping and leverages all available sensors to create a 3D mesh with distinguishable visual features, in close proximity of the user. Google has also shared that it uses Neural Radiance Fields (NeRF) to reconstruct indoor environments [12] but it is unclear if NeRF are used in the AR Cloud Anchor API. It certainly would make sense to use this additional source of easily accessible information.

The chosen indoor location to place a virtual object for this experiment is inside the ZVE building and at close proximity to the IPLab. Like the outdoor location, it is also rich on visual features that can be easily detected by the device sensors, to map the 3D space, and by the VPS find matches later. As can be seen in Fig.3 (Left), a white wall with several frames, text, doors on one side and a textured floor, provide enough reference points to anchor virtual objects in space and determine the AR Pose.

A diffuse red sphere with a diameter of 1 meter, attached to the floor, is chosen as the virtual object that is supposed to resolve in the physical and virtual environments accordingly. After the mapping process completes at the physical location and the cloud anchor is stored it can be successfully resolved, as

shown in Fig. 4 (Left). Restarting the application in the CAVE, that provides the virtual environment for our chosen indoor location, and after choosing to resolve the previously anchored red sphere with a specific id, the user is required to map his close proximity again. When enough reference points are found and the CAVE environment is matched to the physical space, the red sphere appears at the designated virtual location through the AR device as is shown in Fig.4 (Center and Right). While Fig.4 (Left and Center) are screen captures of ARCore Cloud Anchor applications, the right Fig.4 (Right) and the Fig.3 (Left) are photographs of the experimental setting in the CAVE, where also the projection wall edges are clearly visible.

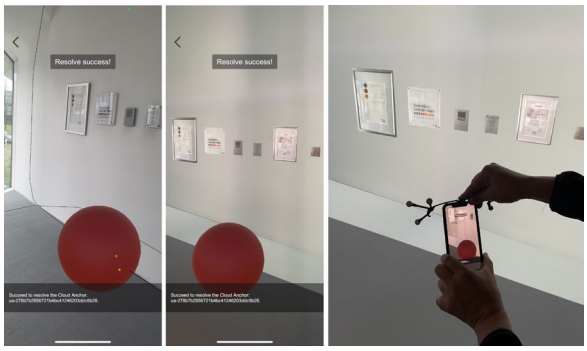


Fig. 4. Side by side comparison of indoor experiment using Google ARCore Cloud Anchor API. **Left** – Device Screen Capture of resolved AR object in physical location. **Center** – Device Screen Capture of resolved AR object in CAVE. **Right** – Experimental setup of AR device and tracking targets in CAVE.

3.4. Manipulation of Google Maps Live View

Live View is an AR feature of *Google Maps* that directs the user in which way to walk using arrows and directions overlaid on the real world through the phone's camera view. It is addressed to the average user of *Google Maps* with an *ARCore* compatible device [13]. Similar to the Geospatial Creator tool, Google leverages its data centers to process the spatial data and run queries in order to match the camera view against known 3d reconstructions and images of the built environment. This way the computational resources, required to complete the complex task of user localization based on pure visual data, are shifted away from the user's device. This shift of computational resources from user devices to data centers enables immersive experiences like AR at urban scale. At the same time Google leverages user generated content to enhance its VPS. Every time *Live View* is activated the user shares a live video feed from their surroundings with Google. In combination with other geolocation and sensor data, this video feed is used to run queries against *Street View*, *Maps and Google Earth* databases.

Live View can only be activated during a navigation session in *Google Maps*. First the user has to ask for directions towards a specific address and start the navigation guide. Then, if the user's device is *ARCore* compatible and VPS coverage is sufficient at the current geolocation, *Live View* is available to use. The graphical user interface consists of textual and visual walking directions. The primary AR object of Live View is a set of white and blue 3D arrows attached on the streets or sidewalks and oriented towards the suggested walking direction. The AR Pose of this virtual objects is constantly updated as the user walks and his geolocation changes. Another virtual object is the red 3D pin, a symbol of the target address and final destination of the current navigation route.

While it is possible to physically walk for a few meters in the CAVE, simulating a complete navigation with multiple nodes is not possible. The device can sense if a user is walking or not and does not update the remaining distance accordingly, even if the virtual environment of the CAVE is translated to a new location. Although the AR arrows adjust to the walking path, the distance to target is not updated accordingly. This means that Live View, in no time, relies purely on the VPS to determine walking distance and directions.

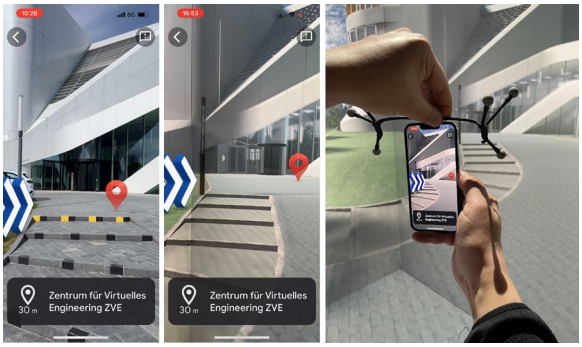


Fig. 5. Side by side comparison of Google Maps Live View manipulation. **Left** – Device Screen Capture of resolved AR navigation objects in physical location. **Center** – Device Screen Capture of resolved AR navigation objects in CAVE. **Right** – Experimental setup of AR device and tracking targets in CAVE.

For our third experiment we chose a short navigation route of 40 meters, starting from the sidewalk of Nobelstreet 12 and ending at the ZVE building, which can be selected as the destination in *Google Maps*. First, we setup the navigation at the physical location, launch Live View, capture the surroundings and wait for the VPS to resolve the virtual objects in the AR view of the device. This intended AR view is show in Fig. 5 (Left). We cancel this navigation, move to the ZVE building, in the CAVE virtual environment, where no GPS signal is available and setup a similar 30 meters long route. Start location is the sidewalk of Nobelstreet and the ZVE building is the destination. A screenshot of this

navigation route is shown in Fig. 6 (Left). The virtual environment of the CAVE has been adjusted to match the real start location outside of the ZVE building, this is shown in Fig. 6. (Right). This way, when *Live View* is activated and the VPS matches the CAVE virtual environment with an actual location from its databases the AR pose of the virtual navigation objects is resolved correctly oriented and scaled as is shown in Fig. 5 (Center) and Fig. 5 (Right). Also, the walking distance of 30 to 40 meters is approximately matched in our AR simulation environment.

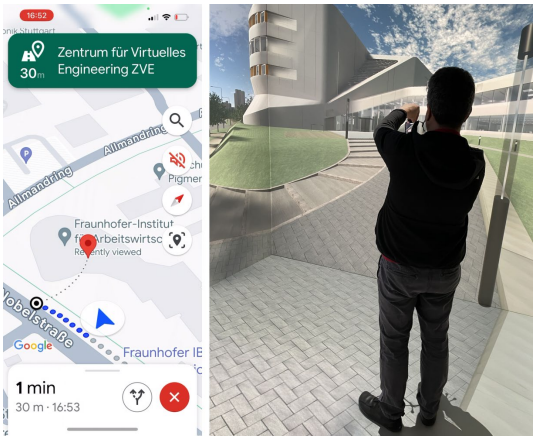


Fig. 6. Experimental setup of Google Maps Live View manipulation in CAVE. **Left** – Device Screen Capture of navigation route. **Right** – AR user getting navigation directions in virtual environment.

4. Conclusions and Future Work

Objective of this work is to share with scientist and developer communities findings about the opportunities and pitfalls of VPS based applications. On one side they offer a significant contribution to AR experiences in general and tools like the Geospatial Creator solve many developer problems by streamlining software architectures and digital asset management. On the other side they can be easily manipulated and should only be relied upon in combination with other sensor data, as we found out is the case in *Live View*, during our third experiment, where the walking distance failed to update during the CAVE simulation.

This work shows that simulation of AR in a CAVE virtual environment reduces overhead of the software development process for urban settings and buildings. If a photorealistic 3D reconstruction is available, then a first prototype of the AR application can be developed without even traveling to the remote site. This has big implications in projects where multiple physical laboratories and campus locations are required to be reconstructed and made AR-ready, in human-centered metaverse-like virtual worlds like in the *Instance Project* of Fraunhofer IAO [14].

In absence of CAVE facilities, the proposed AR test field can also be implemented using smaller VR projection areas, like a powerwall, or LED screens, making it attractive for more developers in smaller laboratories and studios. But tracking the AR device in space, using targets or other techniques, to calculate the correct 3D perspective for the AR device in the virtual environment is essentially for the simulation to succeed, so a tracking system is needed. This setup, with smaller projection areas, works but makes the localization process more difficult and additional time and runs are needed to resolve the AR pose, because pan movement of the device, required to map the surroundings, is limited to the available projection or screen area. Also, human life scale of the 3D models in a restricted projection area is difficult to achieve, whereas in the CAVE we render 3D reconstructions of physical locations in 1:1 scale by default.

When simulating AR applications within our CAVE, luminance and lighting conditions of virtual objects in the AR view of the device cannot be reliably rendered. Modern lighting estimation techniques can approximately provide ambient lighting information of physical environments in real-time, and use that information to correctly illuminate virtual objects in AR [15]. The artificially lit CAVE space fails to provide accurate luminance information and so photorealistic material properties of virtual objects should only be tested in the actual AR site and not in the simulator. We found out that the projector (or screen) resolution plays a vital role in correct registration of AR. Higher display or projection resolutions are also better suited for marker-based AR.

As is the case when the CAVE is used for immersive stereo visualizations a major breaker of immersion for VR users are the clearly visible projection wall edges [16]. This does not seem to be a hurdle for the VPS in order to localize the AR user and resolve the virtual objects. Although it sometimes helps the AR application if a virtual edge is aligned with the virtual one as is attempted in Fig. 3 (Left). The detection of these clearly visible CAVE edges can be a fail-safe mechanism for future updates of VPS. This way artificial environments can be distinguished from real-ones and classified accordingly.

On the other hand, a VPS designed from the ground up to be used in CAVE environments would greatly benefit the AR simulation and the scaling of the AR software development process. Decoupling the VPS from other localization and tracking algorithms during an AR-session will add more control to the simulator and enable testing of applications where locomotion is required. It will reduce sensor confusion that is introduced when some sensor data suggest an indoor location with no GPS signal and the camera "sees" an outdoor public square at street level for example. Using immersive environments in combination with photorealistic synthetic data, like a visual digital twin, enables deep learning approaches not previously possible [17].

Sharing our geolocation or "dropping a pin" to peers is a common practice in everyday life. In forensics a user's approximate geolocation is usually determined using triangulation of cell tower

connection records or GPS data extracted from devices or applications that use GPS for features such as navigation or location-based AR experiences like in our case. As our research shows, VPS can be hacked, be feeding it false information and manipulated into producing results intended for a remote physical location. This vulnerability can be misused and renders localization information, coming purely from VPS, unfit as forensic evidence. In autonomous navigation VPS have been research characterized as unreliable in a safety for life context. because they lack mature integrity frameworks [18]. Future research should determine if AR devices running applications that use VPS also share a resolved localization, in absence of GPS data, wrongfully as reliable geoinformation with third party applications that help users locate their lost devices for example or social networks features like location sharing and tagging.

References

[1]. Zhou, F., Duh, H. B.-L. and Billinghurst, M, Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR, in *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality,* 2008, pp. 193 – 202.

[2]. Marchand, E., Uchiyama, H. and Spindler, F, Pose estimation for augmented reality: A hands-on survey, *IEEE Transactions on Visualization and Computer Graphics*, 22, 12, 2016, pp. 2633–2651.

[3]. E. Ragan, C. Wilkes, D. A. Bowman and T. Hollerer, Simulation of Augmented Reality Systems in Purely Virtual Environments, in *Proceedings of the IEEE Virtual Reality Conference,* 2009, pp. 287-288.

[4]. Nishimoto, A. and Johnson, A. E, Extending virtual reality display wall environments using augmented reality, in *Proceedings of the Symposium on Spatial User Interaction*. New York, NY, USA: ACM, 2019.

[5]. Kim J-Y, Kim I-S, Yun D-Y, Jung T-W, Kwon S-C, Jung K-D., Visual Positioning System Based on 6D Object Pose Estimation Using Mobile Web, *Electronics*, 11, 6, 2022, 865.

[6]. Zhu, Y. et al, Target-driven visual navigation in indoor scenes using deep reinforcement learning, in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA),* 2017, http://vision.stanford.edu/pdf/zhu2017icra.pdf

[7]. Su, Y. et al, Deep multi-state object pose estimation for augmented reality assembly, in *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct),* 2019, pp. 222 - 227.

[8]. Abbaspour A., Wenzel, G., Immersive Besprechung mit Nutzerinnen, in Abbaspour, Amir (Hrsg.): Digitales Bauen mit BIM: Use Case Management im Hochbau, *Beuth,* Berlin (u.a.), 2021, pp. 61-63.

[9]. Hergert, S., Bues, M. and Riedel, O, 50-2: LED-based next generation immersive virtual reality, Digest of technical papers, *SID International Symposium*, 52, 1, 2021, pp. 687–689.

[10]. Tredinnick, R. et al, Uni-CAVE: A Unity3D plugin for non-head mounted VR display systems, in *Proceedings of the IEEE Annual International Symposium Virtual Reality,* 2017, pp. 393 - 394.

[11]. Silva, S., Create world-scale augmented reality experiences in minutes with Google's Geospatial Creator, 2023, *Googleblog.com*. Available at: https://developers.googleblog.com/2023/05/create-world-scale-augmented-reality-experiences-in-minutes-with-google-geospatial-creator.html (Accessed: April 9, 2024).

[12]. Seefelder M. and Duckworth Daniel, 2023, Reconstructing indoor spaces with NeRF, Research.google, Available at: https:// research.google/blog/reconstructing-indoor-spaces-with-nerf/ (Accessed: April 9, 2024).

[13]. Reinhardt T., Using global localization to improve navigation, Research.google. Available at: https://research.google/blog/using-global-localization-to-improve-navigation/ (Accessed: April 9, 2024).

[14]. Hölzle K., Aust M. and Braun S., Putting the human first and second: Challenges of building a human-centered industrial metaverse, *ERCIM News*, No. 137, 2024.

[15]. Agusanto, K. et al, Photorealistic rendering for augmented reality using environment illumination, in *Proceedings of the 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality,* 2003, pp. 208 – 216.

[16]. Freitag, S., Weyers, B. and Kuhlen, T. W, Examining rotation gain in CAVE-like virtual environments, *IEEE Transactions on Visualization and Computer Graphics*, 22, 4, 2016, pp. 1462–1471.

[17]. de Melo, C. M. et al, Next-generation deep learning based on simulators and synthetic data, *Trends in Cognitive Sciences*, 26, 2, 2022, pp. 174–187.

[18]. Zhu, C., Meurer, M. and Günther, C., Integrity of visual navigation-developments, challenges, and prospects, *Navigation,* Washington, 69, 2, 2022, navi.518.

(012)

Dynamic Analysis of 1 MW Steam Turbine during Run-up

R. Rzadkowski, L. Kubitz and A. Koprowski
Aeroelasticity Department, Institute of Fluid-Flow Machinery Polish Academy of Sciences,
Gdańsk 80-231, Fiszera 14, Poland
Tel.: + 48585225169
E-mail: z3@imp.gda.pl

Summary: A 1 MW steam turbine was designed and built to generate electricity from the waste steam of a chemical factory in Kedzierzyn Kozle, Poland. The dynamic of the turbine was analysed, including the vibration of the turbine casing, rotor blades, and bearings using accelerometers at the two bearing casings and two generator bearings. The two displacement sensors in each turbine bearing were used to analyse relative vertical and horizontal vibrations in the casing and rotor. A tip-timing system measured the vibrations of the third-stage bladed disc. During the first run-up, resonance appeared and the turbine was turned off by the control system. This paper analyzes the cause of this resonance using accelerometers on the turbine casing and bearings and inductive sensors for tip-timing blade vibrations.

Keywords: Signal processing, Tip-timing, Inductive sensors and accelerometers.

1. Introduction

A 1 MW steam turbine was designed and built to generate electricity from the waste steam of a chemical factory in Kedzierzyn Kozle, Poland. (Fig. 1). The dynamic of the turbine was analysed. The vibration of the turbine casing, rotor blades, and bearing were measured using accelerometers and tip-timing system. During the first run-up, resonance appeared and the control system turned off the turbine.

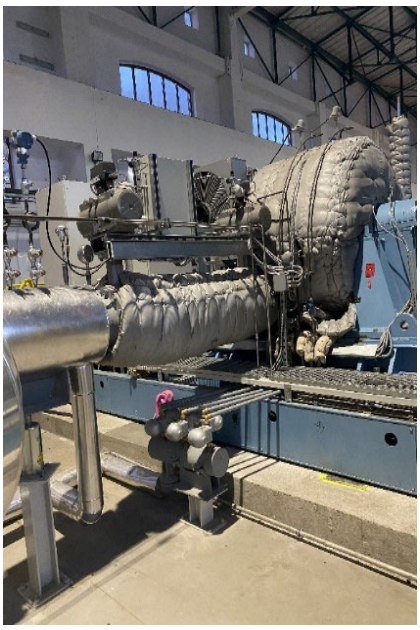


Fig. 1. Steam turbine in the chemical factory.

Measurements of steam turbine last stage LP rotor blade vibrations were presented by Rao and Dutta [1], using a noise sensor mounted in the casing to find the excessive blade vibration caused by flutter in high condenser pressure.

Donato et al. [2] used the tip-timing method with an optical sensor mounted in the casing to find blade vibration in an LP last stage.

The experimental and numerical results of the last-stage low-pressure rotor blade flutter were presented by Sanvito, et al. [3]. The dynamic behavior of blades was numerically investigated, and the grouping of blades in packs was optimized to avoid resonances. This led to the design of new blades, which were mounted in the turbine, and the measured results showed an improvement in the turbine's dynamic behavior.

Prochazka and Vanek [4] used tip-timing to show an increase in the vibration amplitude of a cracked blade in a 1000 MW turbine LP last stage.

Przysowa [5] used tip timing to analyze synchronous and nonsynchronous vibrations of a steam turbine LP last stage rotor blade. This showed how the pressure in the condenser influences rotor blade vibration in a nominal state.

Blade multi-modes using the non-uniform Fourier transform were identified by Kharyton and Bladh [6]. A sparse reconstruction of the blade tip-timing signal for multi-mode blade vibration monitoring was proposed by Lin et al. [7].

The application of the sparse representation theorem to find multimode blade vibration frequencies with uncertainty reduction was presented by Pan et al. [8].

The nonlinear least-squares Levenberg-Marquardt method used in paper [9] determines asynchronous multimode blade vibration components, where the blade can vibrate simultaneously with several modes and frequencies.

Paper [10] presented the method of finding multi-modes of synchronous and nonsynchronous blade vibrations from the tip-timing velocity time of arrival using the Least Squares Technique. This method requires only two sensors in the casing, and a

once-per-revolution sensor for synchronous vibrations but not for asynchronous vibrations.

2. Measuring System of Dynamic Vibration of 1 MW Steam Turbine

Two measuring systems were designed and produced.

The first used (vertical) accelerometers in the two bearing casings, horizontal and vertical ones in the turbine casing above the third stage, and two ones in the generator bearing casing.

The measurement system writes data as unsigned short int data (16 bits). To convert from digital values to physical quantities, the data from the accelerometer had to be decoded. The read values from the measurement system were processed into acceleration values. It was necessary to take into account the sensitivity settings of the accelerometer. To decode digital values into acceleration, the following formula was used:

$$a = \frac{r-o}{s},$$

where r is the acceleration value read from the measurements, o is the measured displacement values, s is the sensitivity of the accelerometer, i.e. conversion of digital values to acceleration values.

After applying the above formula and adjusting the displacement and sensitivity to the specific sensor parameters, the sensor acceleration data was obtained. Thus, the data was transformed from digital format to physical acceleration values, enabling further analysis and interpretation of the measurement results.

The blade displacements were calculated based on the measured times of blade arrival, using blade tip sensors and a once-per-revolution sensor. To find the harmonics of multimode rotor blade vibrations using the tip-timing method, the amplitudes may be assumed as follows:

$$A(t) = \sum_{i=1}^{n} A_i \sin(2\pi f_i + \varphi_i) + \tilde{0}, \quad (1)$$

where $A(t)$ refers to the known values of blade displacements in time, A_i is the amplitude for i-th harmonic, f_i is the frequency of the blade vibrations for i-th harmonics, and t is the known time for which the displacement $A(t)$ was calculated, φ_i is the phase shift for i-th harmonic, $\tilde{0}$ is the "0"- noise level of the blade vibrations.

Equation (1) adequately fits the measured data in the nonlinear least-squares method to obtain the amplitudes A_i, frequencies f_i and phases φ_i of i-th blade mode vibrations.

The nonlinear least-squares Levenberg-Marquardt algorithm (L-M) is used for the fitting [9].

This iterative algorithm is based on successive approximation of the analyzed parameters (i.e. frequency, amplitude and phase):

$$\beta_j^{k+1} = \beta_j^k + \Delta\beta_j, \quad (2)$$

where β_j^k is a parameter value of the blade mode (frequencies (j = 1) or amplitudes (j = 2) or phases (j = 3)), the superscript k is the iteration step, and the difference $\Delta\beta_j$ is called the shift value. At each iteration, the model is linearized using the Taylor series:

$$F(x_i, \boldsymbol{\beta}) = F^k(x_i, \boldsymbol{\beta}) + \sum_j \frac{\partial F(x_i, \boldsymbol{\beta})}{\partial \beta_j}\left(\beta_j - \right. \\ \left. -\beta_j^k\right) = F^k(x_i, \boldsymbol{\beta}) + \sum_j J_{ij}\Delta\beta_j, \quad (3)$$

where $\boldsymbol{\beta}$ is a vector of the parameters (frequency, amplitude and phase).

Jacobian J is a function of the constant (in this case, it is $\tilde{0}$ from Equations (1)–(4)), the independent variable (time) (x_i), and the parameters, and therefore changes from iteration to iteration.

The fitting error for each measurement is

$$r_i = y_i - F^j(x_i, \boldsymbol{\beta}) - \\ -\sum_{j=1}^{m} J_{ij}\Delta\beta_j = \Delta y_i - \sum_{j=1}^{m} J_{ij}\Delta\beta_j, \quad (4)$$

where y_i is the measured value (blade displacement), i = 1, …, n, n is the number of measurement points, F is a function of the model (in this case, right side of Equations (1), m is connected with the number of blade mode components m = 3 for one harmonic, 6 for two harmonics, etc., $\boldsymbol{\beta}$ is a vector of the parameters (frequency, amplitude and phase.

Jacobian J is a function of the constant (in this case, it is $\tilde{0}$ from Equations (1)–(4)), the independent variable (time) (x_i), and the parameters, and therefore changes from iteration to iteration.

The sum of squared fitting errors is minimized

$$S = \sum_i^n r_i^2 \quad (5)$$

The minimum value of S occurs when the gradient is zero

$$\frac{\partial S}{\partial \beta_j} = 2\sum_i^n r_i \frac{\partial r_i}{\partial \beta_j} = 0 \ (j = 1, \dots, m) \quad (6)$$

The number of parameters m, means that there are m gradient equations, β_j is approximated in each step using (11). Next, Jacobian J is linearized:

$$J_{ij} = -\frac{\partial r_i}{\partial \beta_j} \quad (7)$$

Substituting (4) and (7) into (6):

$$-2\sum_{i=1}^{n} J_{ij}(\Delta y_i - \sum_{k=1}^{m} J_{ik}\Delta\beta_k) = 0, \quad (8)$$

where n is the number of measurement points.

Equation (8) can be written as the m of linear equations:

$$\sum_{i=1}^{n}\sum_{k=1}^{m} J_{ij}J_{ik}\Delta\beta_k = \sum_{i=1}^{n} J_{ij}\Delta y_i \quad (j = 1,\dots,m) \tag{9}$$

Equation (9) is analogous to the linear least squares fitting algorithm and can be easily solved.

The nonlinear least-squares Levenberg–Marquardt algorithm requires a gradient of Equations (1) or (2) or (3) or (4) (Jacobian J in the linearized model). For Equation (1), the first component can be calculated by obtaining derivatives concerning each parameter:

$$\frac{\partial}{\partial A}(A\sin(2\pi ft+\varphi)) = \sin(2\pi ft+\varphi) \tag{10}$$

3. Experimental Results

To analyse the dynamic vibration of the turbine, two accelerometers were placed vertically in the two turbine-bearing casings and two generator-bearing casings. Two displacement sensors were placed in each turbine bearing to analyse the relative vertical and horizontal vibrations of the casing and rotor.

Fig. 2 presents the relative vibration of turbine bearing 1 in x direction (dark blue) and y direction (dark green), the relative vibration of turbine bearing 2 in x direction (red) and y direction (light blue), the absolute vibration of turbine bearing 1 (purple), bearing 2 (light blue), generator bearing 1 (dark violet), generator bearing 2 (orange). The rotation speed is black. It can be seen that, due to the extensive absolute vibration of the second turbine bearing at 2840 rpm, the control system turned the turbine off. The variable speed zig-zag up to 2840 rpm was caused by the speed regulator.

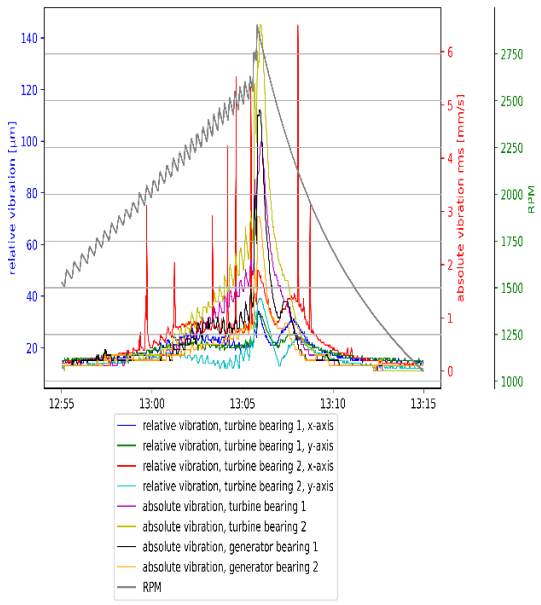


Fig. 2. Dynamic analysis of 1MW steam turbine.

The absolute vibration of bearing 2 was higher than the relative vibration of the casing and rotor, which means that the casing of bearing 2 or the frame was excessively vibrating. The red relative vibration peaks of turbine bearing 2 (Fig. 2) resulted from measurement errors.

A calculation of critical speed was carried out for the rotor. The Campbell diagram shows that there are no critical speeds up to 3000 rpm for 1EO. Another two accelerometers were installed in the casing in the x and y direction above the third bladed disc. Three inductive sensors were installed above the third bladed disc to measure the bladed disc vibration using the tip-timing method [3, 4]. It was found from the accelerometer measurements that a frequency of 51 Hz was independent of rotation speed. The tip timing measurements of the bladed disc show a very small level of vibration.

In the first step, the stiffness of bearing 2 was increased, but the turbine was turned off at 2940 rpm.

The vibrations of the frame were analysed numerically. It was found to have a natural vibration of 51 Hz. So, the frame was supported under the second bearing. The turbine reached a speed of 3000 rpm.

The Campbell diagram for the steam turbine rotor (Fig. 3) shows that the critical speed for 1EO is only 74 rad/s (706 rpm).

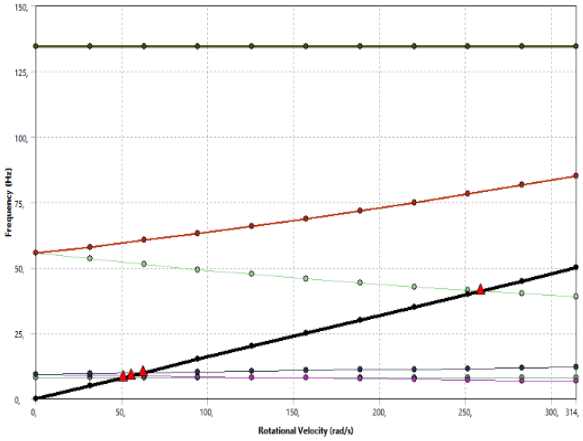


Fig. 3. Campbell diagram of 1 MW steam turbine rotor.

4. Conclusions

The dynamic of a 1MW steam turbine during run-up was analyzed. At 2840 rpm, the turbine was turned off by the control system because of the extensive absolute vibrations of the second turbine bearing. The accelerometer measurements show that the 51 Hz frequency was independent of rotation speed. The tip timing measurements of the bladed disc show a very small level of bladed disc vibration.

In the first step, the stiffness of bearing 2 was increased, but the turbine was turned off at 2940 rpm.

The vibration of the frame was analysed numerically. It was found to have a natural vibration of 51 Hz. Therefore, the frame was supported under the second bearing. The turbine reached a speed of 3000 rpm.

Acknowledgments

The authors wish to acknowledge NCBiR for the financial support of this work POIR.04.01.04-00-0116/17, 1 MW steam turbine powered by steam using waste and process heat, co-financed in 2018-2021 from the European Regional Development Fund under the Smart Growth Operational Program 2014-2020, Priority IV: Increasing the scientific and research potential Measure 4.1 "Research and development", Sub-measure 4.1.4 "Application projects"

References

[1]. A. R. Rao, B. K. Dutta, Blade vibration triggered low load and high back pressure, *Eng. Fail. Anal.*, Vol. 46, 2014, pp. 40-48.

[2]. V. Donato, R. L. Bannister, J. F. De Martini, Measuring blade vibration of large low pressure steam turbine, *Power Eng. Mech. Eng. Part A J. Power Energy*, Vol. 3, 1981, pp. 81-92.

[3]. M. Sanvito, E. Pesatori, N. Bachschmidt, S. Chatterton, Analysis of LP steam turbine blade vibration: experimental results and numerical simulations, in *Proceedings of the 10th International Conference on Vibrations in Rotating Machinery*, London, UK, 11-13 September 2012, pp. 189-197.

[4]. P. Prochazka, F. Vanek, Contactless diagnostics of turbine blade vibration and damage, *J. Phys. Conf. Ser.*, Vol. 305, 2011, 012116.

[5]. R. Przysowa, Blade vibration monitoring in a low-pressure steam turbine, in *Proceedings of the ASME Turbo Expo: Turbomachinery Technical Conference and Exposition*, Vol. 6, Oslo, Norway, 11-15 June 2018, V006T05A025.

[6]. V. Kharyton, R. Bladh, Using tip timing and strain gauge data for the estimation of consumed life in a compressor blisk subjected to stall-induced loading, in *Proceedings of the ASME Turbo Expo: Turbine Technical Conference and Exposition*, Düsseldorf, Germany, 16-20 June 2014, V07BT33A028.

[7]. J. Lin, Z. Hu, Z. S. Chen, Y. M. Yang, H. L. Xu, Sparse reconstruction of blade tip-timing signals for multi-mode blade vibration monitoring, *Mechanical System and Signal Processing*, Vol. 81, 2016, pp. 250-258.

[8]. M. Pan, Y. Yang, F. Guan, H. Hu, H. Xu, Sparse representation based frequency detection and uncertainty reduction in blade tip-timing measurements for multi-mode blade vibration monitoring, *Sensors*, Vol. 17, 2017, 1745.

[9]. R. Rzadkowski, P. Troka, J. Manerowski, L. Kubitz, M. Kowalski, Nonsynchronous rotor blade vibrations in last stage of 380 MW LP steam turbine at various condenser pressures, *Appl. Sci*., Vol. 12, 2022, 4884.

[10]. J. Manerowski, R. Rzadkowski, M. Kowalski, R. Szczepanik, Multimode tip-timing analysis of steam turbine rotor blades, *IEEE Sensor Journal*, Vol. 23, Issue 11, 2023, pp. 11721-11728.

(013)

A Review of 3D Object Detection Methods for Autonomous Driving

Haowei Yang [1], Yuanyao Lu [1] and Haiyang Jiang [2]
[1] School of Information Science and Technology, North China University of Technology, Beijing, PR China
[2] School of Electrical and Control Engineering, North China University of Technology, Beijing, PR China
E-mail: luyy@ncut.edu.cn

Summary: The purpose of 3D object detection is to predict a set of boundary boxes and category labels for each interested object in 3D space, serving as a fundamental task for realizing large-scale automated driving. In recent years, 3D object detection in the context of autonomous driving has become a hot research area in both academia and industry. However, due to limitations in datasets, underutilization of data, and low detection accuracy in multi-sensor fusion, achieving real-time and efficient 3D object detection is not an easy task. In this paper, we provide a review of the field of 3D object detection. This paper first provides a summary of several commonly used datasets for autonomous driving 3D object detection. Secondly, we categorize 3D object detection algorithms according to the type of data sources: LiDAR point cloud-based 3D object detection algorithms, camera image-based 3D object detection algorithms, and LiDAR-camera fusion-based 3D object detection algorithms. We conduct a deep analysis of each type of methods. Finally, we present potential opportunities and challenges for autonomous driving 3D object detection in the areas of data processing, feature extraction strategies, multi-sensor fusion, and dataset distribution. In conclusion, we hope that this paper can inspire further technical reflections among researchers.

Keywords: Autonomous driving, Multi-sensor fusion, 3D object detection, Deep learning, Computer vision.

1. Introduction

Autonomous driving 3D object detection algorithms refer to the use of 3D sensors (such as LiDAR) and 2D sensors (such as cameras) to acquire 3D point cloud and 2D image information in road scenes, thereby achieving automatic detection and recognition of various targets (such as vehicles, pedestrians, traffic signs, etc.) in road scenes.

In recent years, with the rapid development of autonomous driving technology, autonomous driving 3D object detection algorithms have gradually become an indispensable part of autonomous driving technology. Many excellent algorithms based on deep learning have emerged, such as the PointNet [1] series of algorithms based on point clouds and the Faster R-CNN [2] algorithm based on images. These algorithms have achieved significant detection results in their respective datasets. However, autonomous driving 3D object detection technology still faces some challenges, such as low detection accuracy for small targets, slow processing speed of point cloud data, and heavy dependence on the quantity and quality of datasets. These issues require further research and exploration. In addition, the application field of autonomous driving 3D object detection technology is also constantly expanding, extending from autonomous driving vehicles to intelligent transportation, intelligent manufacturing, intelligent security, and other fields, providing infinite possibilities for the realization of intelligent cities and life.

In summary, automatic driving 3D object detection technology is one of the core technologies for realizing autonomous driving technology, with significant application value and research significance. This paper will review the current research status and future development directions in this field, aiming to provide inspiration and ideas for relevant researchers and developers.

2. Datasets

Autonomous driving perception involves multiple datasets [3-5], but only three datasets are widely used, namely KITTI [6], Waymo [7], and nuScenes [8]. Here, we summarize the detailed characteristics of these datasets and list them in Table 1.

The KITTI [6] dataset is one of the widely used datasets in the field of autonomous driving, applicable for 2D, 3D, and bird's eye view detection tasks. Equipped with four high-resolution video cameras, a Velodyne LiDAR scanner, and a state-of-the-art localization system, the KITTI dataset collected a total of 7481 training images and 7518 test images, along with the corresponding point clouds. Within the dataset, only three types of objects (cars, pedestrians, and cyclists) are annotated, with over 200k 3D objects categorized by difficulty levels: easy, moderate, and hard. The common evaluation metric is the Average Precision (AP), used for comparing the performance of KITTI object detection tasks. Additionally, the Average Orientation Similarity (AOS) is utilized to evaluate the performance of joint detection of objects and estimation of their 3D directions.

Waymo [7] is the most widely used public dataset for benchmarking autonomous driving, collected using 5 LiDAR sensors and 5 high-resolution pinhole cameras. Specifically, Waymo comprises 798 training

scenes, 202 validation scenes, and 150 test scenes, each lasting 20 seconds, annotated with vehicles, cyclists, and pedestrians. For evaluating 3D object detection tasks, Waymo provides four metrics: AP/L1, APH/L1, AP/L2, and APH/L2. Among these, AP and APH represent two different performance measurement methods, while L1 and L2 contain targets with different levels of difficulty. Within these metrics, APH is calculated similarly to AP, but weighted to take into account directional accuracy.

Table 1. Survey of Commonly Used Open Dataset and Benchmarks.

Dataset	KITTI [6]	Waymo [7]	NuScenes [8]
Creation Year	2012	2019	2019
Used Radar	1 Velodyne LiDAR scanner	5 LiDAR sensors	1 rotating 32-beam LiDAR sensor
Used Cameras	4 high-resolution video cameras	5 high-resolution pinhole cameras	6 RGB cameras
Annotated Radar Frames	15K	230K	40K
Annotated 3D Object Boxes	80K	12M	1.4M
Annotated 2D Object Boxes	80K	9.9M	-
Traffic Conditions	Urban, suburban, highway	Urban, suburban	Urban, suburban
Task Scenarios	2D, 3D, and bird's eye view object detection tasks	Object detection and tracking	Object detection, semantic segmentation

NuScenes [8] is an open dataset consisting of 1000 driving scenes, divided into 700 for training, 150 for validation, and 150 for testing. The dataset is equipped with cameras, LiDAR, and radar sensors, with annotations for 23 object categories in each key frame, including various types of vehicles, pedestrians, and other objects. For performance evaluation, NuScenes [8] utilizes AP and TP as metrics.

3. 3D Object Detection Methods Based on LiDAR

Point cloud data obtained by LiDAR is crucial for autonomous driving. Point cloud data is a collection of three-dimensional coordinates obtained by LiDAR devices through scanning the surrounding environment. These points can provide precise information about the distance, position, and shape of objects around the vehicle. 3D object detection methods that use only LiDAR point clouds as input data can be divided into Voxel-based methods and Point-based methods, as shown in Fig. 1.

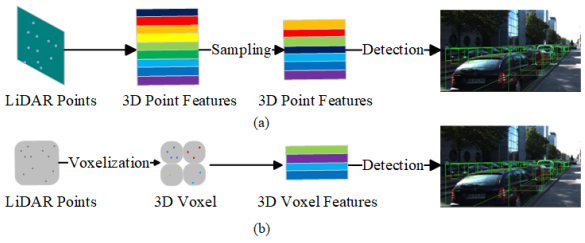


Fig. 1 Comparison of architectures for object detection using point-based and voxel-based methods: (a) Point-based method; (b) Voxel-based method.

The basic idea of voxel-based methods [9-13] is to partition the point cloud space into small voxels and then perform feature extraction and object detection within each voxel. Typically, these methods can be summarized into the following steps: (1) Voxelization: Partitioning the point cloud into a set of equally sized three-dimensional voxels. The voxelization process can utilize regular grids or adaptive methods to maintain appropriate resolution differences between different regions of the point cloud. (2) Feature extraction: Within each voxel, some feature extraction methods are used to obtain the surface features of the voxels, such as point coordinates, colors, normals, density, etc. (3) Object detection: Within the voxels processed for feature extraction, utilizing object detection algorithms to identify objects present within the voxels and predict attributes such as position, size, orientation, etc. (4) Post-processing: Projecting the detected object bounding boxes back into the point cloud space and performing non-maximum suppression (NMS) to eliminate redundant detection results.

Voxel-based methods typically can handle large-scale point cloud data and are easy to compute in parallel. However, due to limitations in voxel size and quantity, they may lose some spatial information and struggle to handle complex scenes and shapes.

The basic concept of point-based methods [14-16] is to directly process raw point cloud data without any preprocessing. Typically, these methods can be summarized into the following steps: (1) Point cloud feature extraction: Extracting features from each point in the point cloud. Usually, the initial features of the point cloud can be represented by coordinates, normals, colors, etc. When extracting features point by point, deep learning-based methods such as PointNet [1], PointNet++ [15], PointCNN [16], etc., can be used to extract local and global features. (2) Point cloud sampling: Point cloud data is often very dense, containing some redundant data. Therefore, it is necessary to sample the point cloud. Sampling methods can utilize random sampling or probability distribution-based sampling. After sampling, the number of points in the point cloud is usually

significantly reduced, thereby reducing computation and storage costs. (3) Object detection: Utilizing object detection algorithms to extract features of objects from the point cloud after feature extraction and sampling, and predicting attributes such as position, size, and orientation of the objects. Object detection algorithms typically employ 3D convolutional neural networks (3D CNN) or point-wise methods to achieve this, such as VoxelNet [17], SECOND [18], PointRCNN [19], etc. (4) Post-processing: Projecting the detected object bounding boxes back into the original point cloud space and performing non-maximum suppression (NMS) to eliminate redundant detection results.

Point-based methods can effectively preserve the spatial information of point clouds while reducing information loss during voxelization. However, due to the sparsity and noise of point clouds, they may face challenges such as high computational complexity and low real-time performance when dealing with large-scale point cloud data. Therefore, researchers are continuously exploring new 3D point cloud object detection methods to improve detection accuracy and efficiency.

4. 3D Object Detection Methods Based on Images

In the field of autonomous driving, early 3D object detection tasks were largely influenced by 2D detection algorithms, most of which were based on predicting 3D bounding boxes from 2D bounding boxes. In recent years, with the emergence of datasets such as KITTI, Waymo, and nuScenes, rapid development has been observed in single-camera [13, 20-22] and stereo-camera-based [23-25] 3D object detection algorithms. Common single-camera 3D object detection methods include the following: (1) 2D Detection with Depth Estimation: Initially, 2D object detection algorithms (e.g., Faster R-CNN [2], YOLO [25], etc.) are employed to detect objects in images and estimate their 2D bounding boxes. Then, depth estimation algorithms (e.g., those based on monocular image depth estimation) are used to estimate the distance to the objects, thus obtaining their 3D positions. (2) Deep Learning Approaches: In recent years, with the advent of point cloud datasets and improved computational power, deep learning-based single-camera 3D object detection methods have also seen significant advancement. For example, neural networks based on deep learning can be used to directly extract the 3D positions of objects from 2D images. These methods typically require large amounts of labeled data for training but often achieve higher detection accuracy. DETR3D [26] projected learnable 3D queries into 2D images, then sampled corresponding features, thus achieving end-to-end 3D object detection.

3D object detection methods based on stereo vision utilize images from left and right cameras. By calculating the disparity between the two cameras, depth information of the objects can be obtained. Common stereo-based 3D object detection methods include the following: (1) Traditional Computer Vision-Based Stereo 3D Object Detection: Early research on stereo 3D object detection relied mainly on traditional computer vision methods, such as disparity calculation and stereo image registration. These methods typically involve manual feature extraction, matching, and then depth computation of the objects. (2) Deep Learning-Based Stereo 3D Object Detection: In recent years, deep learning-based stereo 3D object detection methods have emerged gradually, often utilizing technologies like Convolutional Neural Networks (CNNs) to learn features from stereo images for object detection and depth estimation. For instance, an end-to-end CNN can be employed, taking stereo image pairs as input and producing output for object positions and depth information. (3) Optical Flow-Based Stereo 3D Object Detection: Another common stereo 3D object detection method is based on optical flow calculation. By computing the motion relationship between pixels in the left and right images, depth information within the images can be derived. These methods usually involve registration of images across frames and utilize motion patterns between pixels to calculate depth information.

Compared to monocular 3D object detection, stereo 3D object detection methods can acquire more precise depth information of objects. However, they require complex preprocessing such as stereo image calibration and have high hardware requirements, necessitating trade-offs in practical applications.

5. 3D Object Detection Methods Based on Fusion of LiDAR and Images

Since data collected by LiDAR and cameras usually contain complementary information, many algorithms for data fusion of these two sensors have emerged in the field of 3D object detection for autonomous driving in recent years. These algorithms can be categorized into three types based on fusion mechanism: Early fusion, Late fusion, Deep fusion, and Interactive fusion, as illustrated in Fig. 2.

Early fusion is a fusion mechanism that directly combines LiDAR and camera data at either the data or feature level, resulting in a larger tensor for processing. In Paper [27], an approach based on an end-to-end deep neural network was proposed. This approach involves inputting both types of data into a shared convolutional neural network to extract features, followed by concatenating them and feeding them directly into a fully connected layer for classification. Paper [28] utilized early fusion of RGB images and depth images for object recognition. Furthermore, Paper [29] integrated LiDAR and camera data for 3D object detection and tracking in dynamic environments.

Late fusion involves merging results from different data branches. For instance, LiDAR and camera data are separately input into their respective networks for processing, and the results from both networks are

merged in the end. Paper [30] utilized images captured by stereo cameras for object detection, improving detection accuracy by fusing information from two images. Paper [31] input images and point clouds into two separate networks and fused the outputs of both networks, achieving high-precision autonomous driving object detection. Paper [32] proposed a method by separately inputting LiDAR and camera data into two networks, then merging the outputs of both networks, yielding satisfactory detection results. Paper [33] performed feature fusion separately for point cloud and image data after feature extraction, obtaining two different feature vectors. These vectors were then concatenated as the final fusion feature for object detection.

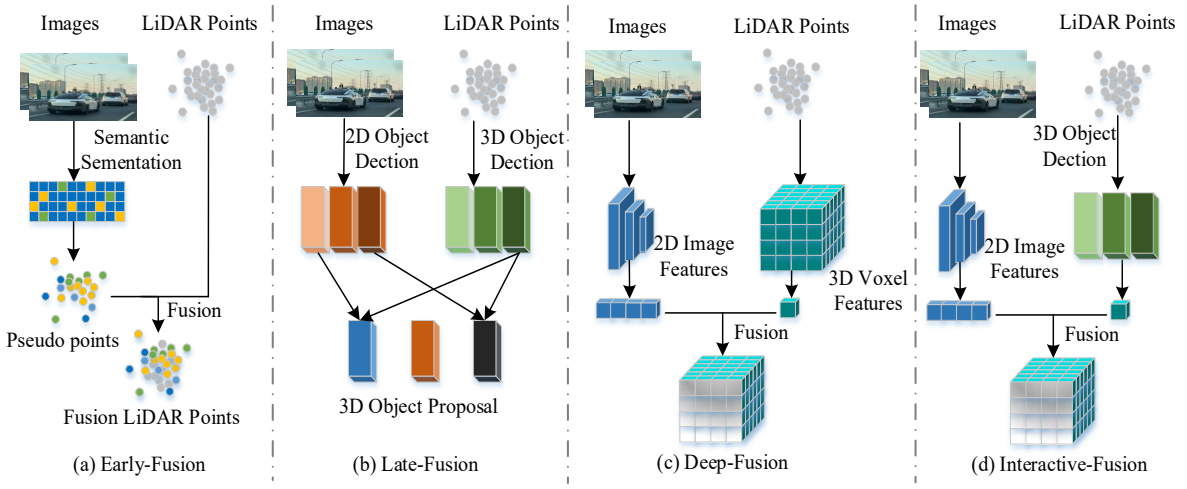


Fig. 2. Comparison of different LiDAR-Camera fusion methods: (a) Early-Fusion method fuses two modalities of data at the data level; (b) Late-Fusion method fuses two modalities of data at the result level; (c) Deep-Fusion method fuses two modalities of data at the feature level. (d) Interactive-Fusion method fuses two modalities of data at different level.

Deep fusion is a sensor fusion mechanism that integrates deep learning, performing multi-layer feature fusion of sensor data through multiple layers of deep neural networks to acquire richer feature representations. Compared to early fusion and late fusion, deep fusion can better integrate the complementarity of sensor information and high-level semantic features, thereby possessing superior representational capability and robustness. TransFusion [34] employs Transformer models to encode and fuse point cloud and image data, ultimately generating 3D bounding boxes and confidence scores for each object. 4D-Net [35] utilizes 3D convolutional neural networks (CNN) and 2D CNN for feature extraction from point cloud and image data respectively, then fuses the features of both networks for 3D object detection. BevFusion [36] employs two separate networks to extract features from point clouds and images respectively. These features are then fused together using a convolutional network, and finally, a detection method similar to the DETR [37] model is applied for 3D object detection.

Interactive Fusion is a fusion mechanism that integrates different data branches at both the result and feature levels. Inspired by DETR [37], we can extract features from the data and transform them into Object Queries containing object information, which are then fused with the features of another data branch. Such methods typically exhibit optimal fusion effects. In DeepFusion [38], two different networks are utilized to extract features from point clouds and images respectively, generating Object Queries. Subsequently, the Object Query from the point cloud is fused with the image features, followed by the fusion of the Object Query from the image with the point cloud features. Finally, convolutional networks are employed to generate 3D bounding boxes and confidence scores for each object.

6. Conclusions

In autonomous driving technology, 3D object detection algorithms play a crucial role. This paper categorizes datasets and existing 3D object detection algorithms in autonomous driving. Based on data sources, 3D object detection methods can be classified into those based on LiDAR, those based on cameras, and those based on the fusion of LiDAR and cameras. Each method has its own advantages and disadvantages, with some focusing more on improving detection accuracy while others prioritize real-time performance and low power consumption. With the continuous development of autonomous driving technology, future research directions include better data augmentation techniques, optimization and improvement of algorithms, and the fusion of 3D object detection algorithms with other sensors to achieve more efficient and accurate autonomous driving systems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (61971007 & 61571013).

References

[1]. C. R. Qi, H. Su, K. Mo, L. J. Guibas, PointNet: Deep learning on point sets for 3D classification and segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*, 2017, pp. 652-660.

[2]. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards real-time object detection with region proposal networks, in Advances in Neural Information Processing Systems, Vol. 28, *Curran Associates, Inc*., 2015.

[3]. M. F. Chang, J. Lambert, P. Sangkloy, J. Singh, et al., Argoverse: 3D tracking and forecasting with rich maps. in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, 2019, pp. 8748-8757.

[4]. W. Chen, Z. Liu, H. Zhao, S. Zhou, et al., CUHK-AHU dataset: promoting practical self-driving applications in the complex airport logistics, hill and urban environments, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'20)*, 2020, pp. 4283-4288.

[5]. T. Gruber, F. Julca-Aguilar, M. Bijelic, F. Heide, Gated2depth: Real-time dense lidar from gated images. in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV'19)*, 2019, pp. 1506-1516.

[6]. A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? The Kitti vision benchmark suite, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*, 2012, pp. 3354-3361.

[7]. P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, et al., Scalability in perception for autonomous driving: Waymo open dataset, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, 2020, pp. 2446-2454.

[8]. H. Caesar, V. Bankiti, A. H. Lang, S. Vora, et al., Nuscenes: A multimodal dataset for autonomous driving, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, 2020, pp. 11621-11631.

[9]. J. Ku, M. Mozifian, J. Lee, A. Harakeh, S. L. Waslander, Joint 3D proposal generation and object detection from view aggregation, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'18)*, 2018, pp. 1-8.

[10]. B. Yang, W. Luo, R. Urtasun, Pixor: Real-time 3D object detection from point clouds. in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*, 2018, pp. 7652-7660.

[11]. M. Liang, B. Yang, S. Wang, R. Urtasun, Deep continuous fusion for multi-sensor 3D object detection, in *Proceedings of the European Conference on Computer Vision (ECCV'18)*, 2018, pp. 641-656.

[12]. B. Yang, M. Liang, R. Urtasun, Hdnet: Exploiting HD maps for 3D object detection, in *Proceedings of the Conference on Robot Learning (PMLR'18)*, 2018, pp. 146-155.

[13]. A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, O. Beijbom, Pointpillars: Fast encoders for object detection from point clouds, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'19)*, 2019, pp. 12697-12705.

[14]. J. Ngiam, B. Caine, W. Han, B. Yang, et al., Starnet: Targeted computation for object detection in point clouds, *arXiv Preprint*, 2019, arXiv:1908.11069.

[15]. C. R. Qi, Y. Li, S. Hao, et al., PointNet++: deep hierarchical feature learning on point sets in a metric space, in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*, 2017, pp. 5099-5108.

[16]. Y. Li, R. Bu, M. Sun, W. Wu, et al., PointCNN: Convolution on x-transformed points, in Advances in Neural Information Processing Systems, Vol. 31, *Curran Associates, Inc*., 2018, pp. 820-830.

[17]. Y. Zhou, O. Tuzel, Voxelnet: End-to-end learning for point cloud-based 3D object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*, 2018, pp. 4490-4499.

[18]. Y. Yan, Y. Mao, B. Li, SECOND: Sparsely Embedded Convolutional Detection, *Sensors*, Vol. 18, Issue 10, 2018, 3337.

[19]. S Shi, X Wang, H. Li, PointRCNN: 3D object proposal generation and detection from point cloud, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'19)*, 2019, pp. 770-779.

[20]. Z. Liu, D. Zhou, F. Lu, J. Fang, L. Zhang, Autoshape: Real-time shape-aware monocular 3D object detection, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV'21)*, 2021, pp. 7769-7778.

[21]. A. Kumar, G. Brazil, X. Liu, Groomed-NMS: Grouped mathematically differentiable NMS for monocular 3D object detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 2021, pp. 8973-8983.

[22]. Y. Wang, A. Fathi, A. Kundu, D. A. Ross, C. Pantofaru, T. Funkhouser, J. Solomon, Pillar-based object detection for autonomous driving, in *Proceedings of the European Conference on Computer Vision (ECCV'20)*, 2020, pp. 18-34.

[23]. T. Yin, X. Zhou, P. Krahenbuhl, Center-based 3D object detection and tracking, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 2021, pp. 11784-11793.

[24]. J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, in *Proceedings of the European Conference on Computer Vision (ECCV'17)*, 2017, pp. 7263-7271.

[25]. C.-Y. Wang, A. Bochkovskiy, H.-Y. M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'23)*, 2023, pp. 7464-7475.

[26]. Y. Wang, V. Campagnolo Guizilini, T. Zhang, Y. Wang, H. Zhao, J. Solomon, Detr3D: 3D object detection from multi-view images via 3D-to-2D queries. in *Proceedings of the Conference on Robot Learning (PMLR'22)*, 2022, pp. 180-191.

[27]. S. Vora, A. H. Lang, B. Helou, O. Beijbom, Pointpainting: Sequential fusion for 3D object detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'20)*, 2020, pp. 4604-4612.

[28]. X. Chen, K. Kundu, et al., 3D Object proposals using stereo imagery for accurate object class detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 40, Issue 5, 1 May 2018, pp. 1259-1272.

[29]. X. Chen, H. Ma, J. Wan, B. Li, T. Xia, Multi-view 3D object detection network for autonomous driving, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*, 2017, pp. 1907-1915.

[30]. S. Pang, D. Morris, H. Radha, CLOCs: Camera-LiDAR object candidates fusion for 3D object detection, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'17)*, 2020, pp. 10386-10393.

[31]. G. Melotti, C. Premebida, N. M. M. da S. Gonçalves, U. J. C. Nunes, D. R. Faria, Multimodal CNN pedestrian classification: a study on combining LIDAR and camera data, in *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC'18)*, 2018, pp. 3138-3143.

[32]. G. P. Meyer, J. Charland, D. Hegde, A. Laddha, C. Vallespi-Gonzalez, Sensor fusion for joint 3D object detection and semantic segmentation, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'20)*, pp. 1230-1237.

[33]. S. Gu, Y. Zhang, J. Tang, J. Yang, J. M. Alvarez, H. Kong, Integrating dense LiDAR-camera road detection maps by a multi-modal CRF model, *IEEE Transactions on Vehicular Technology*, Vol. 68, Issue 12, 2019, pp. 11635-11645.

[34]. X Bai, Z Hu, X Zhu, et al., TransFusion: robust LiDAR-camera fusion for 3D object detection with transformers, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*, 2022, pp. 1080-1089.

[35]. A. J. Piergiovanni, V. Casser, M. S. Ryoo, A. Angelova, 4D-net for learned multi-modal alignment, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 2021, pp. 15435-15445.

[36]. T. Liang, H. Xie, K. Yu, Z. Xia, et al., Bevfusion: A simple and robust lidar-camera fusion framework, in *Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS'22)*, 2022, pp. 10421-10434.

[37]. N. Carion, F. Massa, G. Synnaeve, N. Usunier, et al., End-to-end object detection with transformers, in *Proceedings of the European Conference on Computer Vision (ECCV'20)*, 2020, pp. 213-229.

[38]. Y. Li, A. W. Yu, T. Meng, B. Caine, et al., Deepfusion: Lidar-camera deep fusion for multi-modal 3D object detection, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'22)*, 2022, pp. 17182-17191.

(015)

A Methodological Approach to Machine Learning for Forecasting Agricultural Commodity Prices

H. Schallner
Jade Hochschule, University of Applied Sciences Friedrich-Paffrath-Str. 101,
26389 Wilhelmshaven, Germany
Tel.: +49 1638717728
E-mail: harald.schallner@jade-hs.de

Summary: The accurate forecasting of agricultural commodity prices is essential for ensuring food security by improved planting planning. Traditional econometric models have been the backbone of such forecasting efforts; however, the advent of machine learning (ML) offers promising enhancements in predictive accuracy and adaptability to market dynamics. This paper proposes a comprehensive methodological approach to applying ML algorithms for the forecasting of agricultural commodity prices. This methodology encompasses data preprocessing, feature selection based on economic factors influencing agricultural prices, model training, and a rigorous evaluation framework that includes out-of-sample testing and cross-validation to assess forecast accuracy and robustness. Our findings indicate that especially transformer networks significantly outperform traditional econometric models and long short-term memory networks in forecast accuracy. This research contributes to the literature by providing a detailed methodological framework for the application of ML in agricultural price forecasting, offering insights for researchers, policymakers, and market participants.

Keywords: Forecasting, Agricultural commodity prices, Methodology, Machine learning, Transformer networks.

1. Introduction

Agricultural commodities, including grains, meat, and dairy products, form the backbone of global food systems. The ability to forecast these prices with higher accuracy holds significant implications for planting planning and food security. Forecasting agricultural commodity prices presents a multifaceted challenge, influenced by a variety of factors that make precise predictions difficult.

Agricultural markets are highly seasonal, with prices subject to short-term fluctuations caused by seasonal events, holidays, and harvest periods. Predicting these fluctuations requires detailed knowledge of seasonal patterns and the ability to respond to market changes quickly. Due to unpredictable long-term weather and pest infestations, it is difficult to forecast production quantities, which complicates accurate price forecasting. The lag between sowing, growing and harvest leads to delayed supply adjustments to changing market conditions, complicating price trend forecasting. Many agricultural products are highly perishable, leading to rapid price changes in response to supply and demand fluctuations. Incorporating perishability into forecasting models poses a significant challenge. The quality of agricultural products varies significantly and directly influences prices. Forecasting price trends thus requires accurate quality assessments, which depend on many variable factors. The market power of large retail chains can lead to price distortions that are difficult to predict. These entities often have the capacity to influence prices, complicating the modeling of price formation on agricultural markets.

Conventional statistical models, such as ARIMA (AutoRegressive Integrated Moving Average) and SARIMA (Seasonal ARIMA), are widely utilized in forecasting producer prices. These models analyze historical price data to identify patterns and trends, and based on these observations, predict future prices. However, they typically exhibit a linear structure and may not fully account for the complex interactions among various influencing factors and therefore with limited forecast accuracy [1].

A comprehensive analysis of recent research, discussing the strengths and weaknesses of various machine learning techniques concluded that machine learning has the potential to revolutionize agricultural price prediction [2]. This state-of-the-art study shows that, there is a lack of a systematic methodology, which describes a generic approach. In addition, currently invented transformer networks [3] have not been applied to improve prediction accuracy for time series data, yet. Therefore, this paper proposes a generic methodology to forecast agricultural commodity prices based on transformer networks.

2. Methodology Steps

The proposed methodology comprises a sequence of six main steps that are connected by two improvement loops. In the first step, relevant time series data has to be identified. These data have to be prepared and normalize in the following second step. In step three, a transformer network has to be configured by its model layers and hyperparameters. In step four, transformer network has to be trained. The forecast accuracy is validated based on out-of-sample testing and cross-validation in step five. Depending on this results step three has to be executed with tuned hyperparameters, iteratively. After achieving

sufficient forecast accuracy, the impact of time series data on prices has to be analyzed. Therefore, heatmaps give a quick and intuitive picture of temporal correlations. This helps to review selected time series data in step two in order to filter relevant data and reduce model complexity, thereby.

3. Methodology Evaluation

In a case study the proposed methodology was applied to oilseeds, potatoes, cereals, meat and milk prices that were provided by the German Lower Saxony Chamber of Agriculture [4].

Step 1: Relevant time series data are prices for energy, animal feed, seeds, fertilizer, pesticide, agricultural machinery, labor costs and land rental rates. In addition, weather data (rain amount, temperatures, sunshine hours), agricultural key figures (cultivated areas, livestock, number of farms and employees, subsidies), consumer behavior (per capita consumption, consumer spending on food), and macroeconomic key figures (inflation rates, exchange rates) have been identified.

Step 2: Normalizing time series data is a critical preprocessing step before feeding the data into transformer networks. Transformers, by design, are sensitive to the scale of the input data, and normalization helps in stabilizing the training process and improving model performance. Z-Score Normalization (Standardization) was chosen. This technique involves transforming the data to have a mean of 0 and a standard deviation of 1. Each feature was normalized independently. Series data was split into a sequence of fixed-length windows (here: 4 months). This involves creating input sequences that the transformer will learn from, typically using a sliding window approach. Time series data begins in December 2015 and ends November 2023.

Step 3: The choice of model size (number of layers, dimensionality of the feedforward network, number of attention heads) reflected the complexity of the task and the amount of available data. Overly large models may overfit when trained on limited financial data. Implemented Keras model comprises six transformer encoders each with MultiHeadAttention layers and four attention heads.

Step 4: Transformer network uses the Adam optimizer the for training, known for its effectiveness in handling sparse gradients and adaptive learning rates.

Step 5: The loss function Mean Absolute Error (MAE) calculates the accuracy in forecasting prices. Cross-validation computes MAE results between 0.0031 and 0.0124 for the best hyperparameter configuration of milk price prediction.

Step 6: In heatmap (Fig. 1), colors indicate the strength and direction of the Pearson correlation coefficient of some selected time series for milk price in Lower Saxony. The color gradient from blue color to red shows the range from negative -1.0 to positive +1.0 correlation, with more intense colors indicating stronger correlations. Grey color represents small correlation with value near 0.0. Clusters of highly correlated temporal variables are visualized which indicate groups of variables that behave similarly over time.

Fig. 1. Temporal Correlation Heatmap for Historical Milk Price in Lower Saxony (partial excerpt).

4. Conclusions

Based on the results of the evaluation, optimization strategies were developed to further enhance the performance of the models and ensure their applicability to real-world producer price forecasting problems. It is important to note that the successful application of the methodology depends on several factors, such as the quality of the available data, the adaptability of the transformer model, and the configuration of the appropriate hyperparameters. Summarized, the implemented pipeline for finding optimized hyperparameter values and the heatmap evaluation loop enables a continuous improvement process for forecasting prices.

References

[1]. G. E. Box, et al., Time Series Analysis: Forecasting and Control, *John Wiley & Son*, 2015.

[2]. N.-Q. Tran, et al., Predicting Agricultural commodities prices with machine learning: a review of current research, *arXiv Preprint*, 2023, arXiv:2310.18646.

[3]. A. Vaswani, et. al., Attention is all you need, *arXiv Preprint*, 2017, arXiv:1706.03762.

[4] Landwirtschaftskammer Niedersachsen, www.lwk-niedersachsen.de/markt-preise

(016)

Pressure Ulcers Monitoring with Combined Piezo- and Chemo-resistive Nanocomposite Sensors' Arrays

J. F. Feller [1], M. Castro [1], M. T. Tran [1] and W. Allègre [2]
[1] Smart Plastics Group, IRDL CNRS 6027 – Univ. South Brittany (UBS), Lorient, France
[2] Kerpape Centre for Functional Rehabilitation, Ploemeur, France
Tel.: + 33 2 97 87 45 84
E-mail: jean-francois.feller@univ-ubs.fr

Summary: The early detection of pressure ulcers is expected to improve the health of disabled patients and save hospital costs. Also, several existing devices have been developed to monitor wounds, we are presenting an original strategy that combines the different smart properties of conducting polymer nanocomposites (CPC) for the monitoring of the pressure applied on skin with piezo-resistive sensors (pQRS) and the evolution of volatiles emitted by bedsores with chemo-resistive sensors (vQRS). Piezo-resistive responses of the array of pQRS are converted with a Snowboard® card into coloured pixels to visualise the gradient of pressure applied on the skin, whereas chemo-resistive responses of the array of vQRS (e-nose) are analysed with a PCA algorithm to determine their discrimination ability between "normal" and "bedsore" olfactive imprints.

Keywords: Vapour sensors, Pressure sensors, Polymer nanocomposite transducer, Bedsores detection, Quantum resistive sensors, PCA treatment.

1. Introduction

Pressure ulcers (PU) also called bedsores are a serious global health challenge, affecting hundred million people in the world and putting immense pressure on healthcare systems [1]. Sensor-based diagnostic tools and monitoring systems have emerged as a non-invasive solution to reduce the occurrence of new cases of PU and promise a significant reduction in treatment expenditure and time [2]. In particular new technologies such as wearable sensors [3], [4], electronic skin [5], [6], smart dressing [7], [8] or epidermal electronics [9], [10] offer a wide range of integrated monitoring platform solutions.

Nonetheless, further innovations are necessary to associate multiple types of sensor arrays, particularly pressure and chemical sensor-based e-skins in a microsystem for performing real-time assessment of all the critical wound parameters, what is the objective of the present work.

1.1. Materials

For the synthesis of pQRS transducers, hybrids of carbon nanotubes from NANOCYL SA and graphene nanoplatelets architecture in house of card were stabilized by a poly(urethan) PU matrix [11].

For the synthesis of vQRS transducers, Fig. 1 shows the series of polymers of different chemical selectivities towards the biomarkers that has been chosen, i. e., poly(vinyl pyrrolidone) PVP, poly(vinyl acetate) PVA, poly(lactic acid) PLA, poly(ethylene glycol) PEG, poly(caprolactone) PCL, poly(methyl methacrylate) PMMA, poly(styrene-co-acrylonitrile) SAN, poly(styrene) PS and sulfonated poly(ether-co-ether-ter-ketone) PEEK, in order to functionalize nanocarbons [12].

All transducers were assembled hierarchically by spray layer by layer (sLbL) [14].

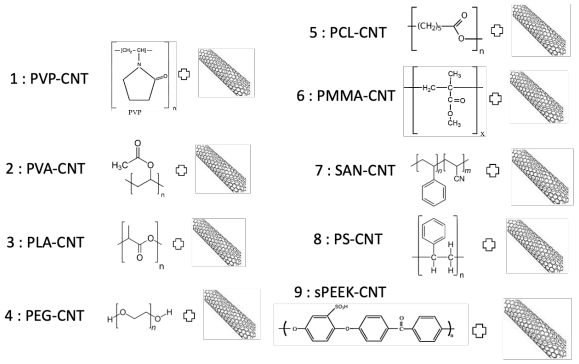


Fig. 1. Formulations of conductive polymer nanocomposite suspension used to sLbL transducers [13].

1.2. Processing

All nanocomposite transducers are processed by a home-made additive manufacturing technology, i.e., spray layer by layer allowing a good control of the conducting architecture from the nano- to the micro-scale, see Fig. 2.

This process allows to adjust the initial resistance R_0 between 50 to 500 kΩ, by varying either the amount of carbon in the CPC solution, and/or the number of sprayed layers. The nozzle forms a cone of atomized microdroplets of 50 µm diameter in which a good dispersion of carbon nanofillers and macromolecules into the solvent is preserved. When the microdroplets

hit and wet the surface, they are expected to weld together in 2D and subsequently in 3D during the evaporation of the solvent to form the transducer.

Typically, from 10 to 60 nanolayers about 40 nm thick are sputtered onto interdigitated electrodes (obtained by micro-capacity cleavage) to fabricate a 1 µm thick transducer.

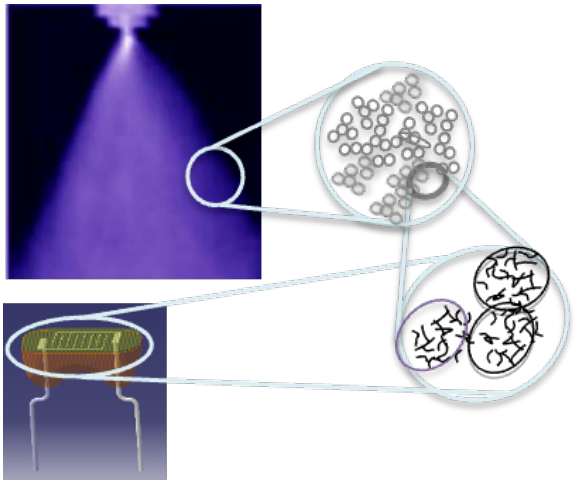


Fig. 2. Spray layer by layer process used for transducers' fabrication.

1.3. Characterization

To characterize the piezo-resistive behaviour of pQRS, we have assembled them into an array before submitting their individual surface of about 1 cm^2 to standard weights from 200 to 1600 g as in Fig. 3.

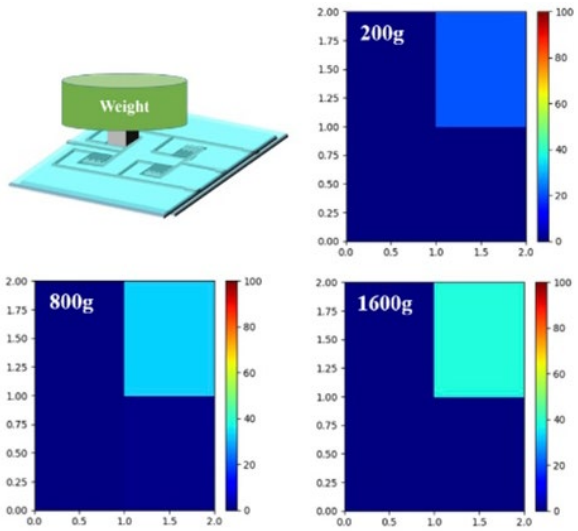


Fig. 3. Determination of pQRS sensitivity to pressure.

The piezo-resistive response Ar = $\Delta R/R_0$ is then converted into a colour by a Snowboard® card to better visualize the pressure gradient with pixels.

To characterize the chemo-resistive behaviour of the vQRS assembled into an array (e-nose), a vapour sensing device has been home-made (Fig. 4), using a carrier gas (dry nitrogen or air) driving a controlled quantity of volatile organic compounds (VOC) to be detected (target biomarkers) towards the cell containing the vapour sensors' array.

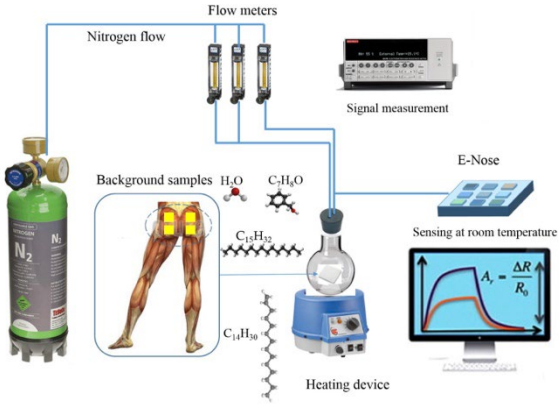


Fig. 4. Vapour sensing home-made device (e-nose).

In this device the VOC source can be either saturated vapours produced by bubbling into an Erlenmeyer containing a liquid or the head space composed of vapours desorbed from a patch heated into a balloon as in (Fig. 4).

In a first step the e-nose is trained with biomarker vapours to confirm that both sensitivity and selectivity of vQRS are appropriate. Then it is exposed to vapours desorbed from patches picked from the skin of a healthy volunteer after 11 hours of contact (and sealed in a vial before desorption) to qualify a representative background (Fig. 5).

Fig. 5. Patch samples preparation for the e-nose.

This reference is then compared to artificial samples made of patches on which drops of biomarkers had been deposited prior to desorption.

2. Results

2.1. Piezo-resistive Behaviour

A first proof of concept of pression mapping has been made with four quantum resistive pressure

sensors (pQRS) [15], assembled into an array to visualise the surface pressure applied on skin as can be seen in Fig. 6.

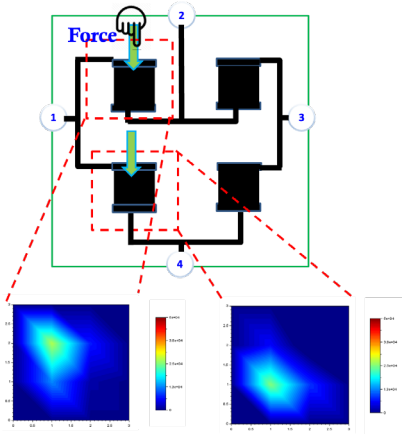


Fig. 6. Pressure mapping with an array of four pQRS.

To reach the ultimate objective, i.e., detecting when a disabled patient has seated too long in the same position, this matrix of four pixels could of course be extended to increase the precision of the diagnostic.

2.2. Chemo-resistive Behaviour

As an example, the chemo-resistive responses of four of the nine vQRS, PVP-CNT, PVA-CNT, PLA-CNT and PEG-CNT have been plotted in Fig. 7. Typical signals are obtained upon successive cycles of 5 min exposure to benzyl alcohol flow and dry nitrogen for rinsing. It can be noticed that vQRS respond all within the second, have reproducible and low noise responses, making unnecessary filtering. The desorption is complete as no drift is observed in R_0, the initial resistance, meaning that no vapour molecule is kept inside the transducer.

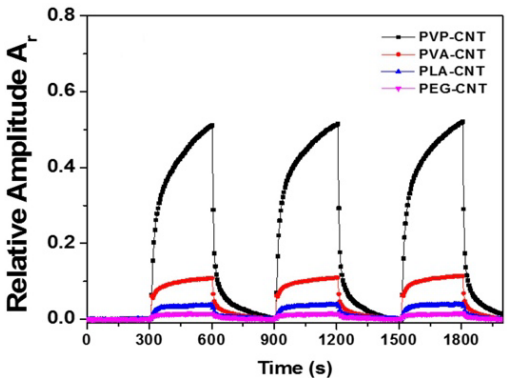


Fig. 7. Chemo-resistive response of an array of four vQRS to benzyl alcohol.

Moreover, in this peculiar case, it can be seen that PVP-CNT exhibits the larger response to saturated vapours of benzyl alcohol, one of the identified biomarkers of pressure ulcers.

To push vQRS closer to their limit of detection, in Fig. 8 the concentration of benzyl alcohol has been decreased from thousand ppm (parts per million) to hundred ppb (parts per billion), which shows that they still can detect molecules at that level which is closer to the application.

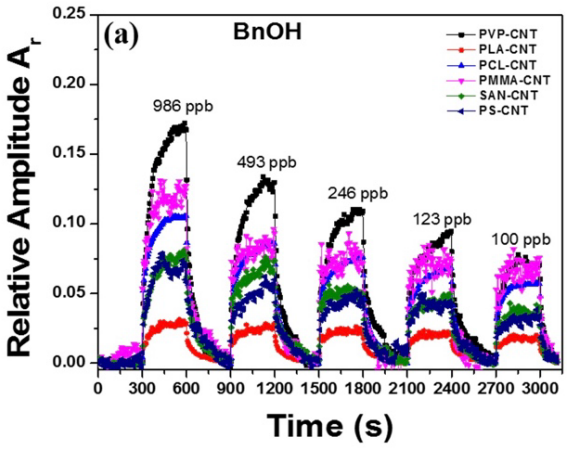


Fig. 8. Ar of 6 vQRS to ppm level of benzyl alcohol

Interestingly the amplitude of vQRS responses Ar max is found proportional to the analyte concentration.

3. Analysis

Integrating a set of pQRS into a cloth will allow triggering a preventive action to avoid a too long compression in the same area of the body and thus decrease the chances of bedsores' development.

In complement to this information on the nature of the patient's seating, the collection of VOC emitted by the body (part of the volatolome) can bring a pertinent analysis on the possible level of degradation of skin (4 steps can be identified until the bone is reached, but obviously being able to make an early diagnosis of the first one would prevent painful wound and long curing). During training, nine sensors of different nature have been exposed to eight VOC biomarkers of pressure ulcers (previously identified [16] by gas chromatography coupled with mass spectrometry) to confirm the complementarity of sensors and the full coverage of detection. Then in conditions closer to reality, the VOC desorbed from patches on which drops of biomarkers had been put, were analysed by the e-nose. Data treatment allowed the extraction of all maximum amplitudes Ar of sensors/vapours couples on five cycles to feed a PCA algorithm. This gave the graph of Fig. 9 where the discrimination ability of the e-nose to for different olfactive imprints, i.e., background, background + benzyl alcohol, background + pentadecane, background + tetradecene is clearly shown.

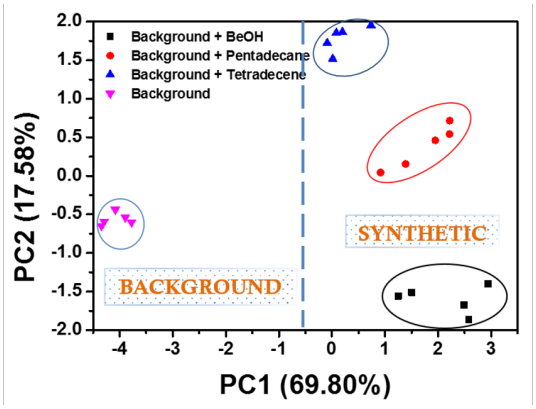


Fig. 9. PCA of vQRS response to express their discrimination ability.

To go further it will be necessary to combine several VOC on the same patch, to investigate the influence of water molecules on the detection of biomarkers and to collect patches on true patients suffering from bedsores, which are the next steps of this study.

4. Conclusion

The proof of concept of using conductive polymer nanocomposite (CPC) based quantum resistive sensors QRS) both to map the pressure applied on patient skin in combination of with the identification of olfactive imprint of their skin is showing promises in the prevention of bedsores which is a significant healthcare challenge.

However, using simple date treatments does not give yet a complete vision of the level of development of pressure ulcers, which would require to multiply experiments with true patches picked from patients with well-defined wounds.

Both types of sensors were found highly sensitive respectively to pressure (some kPa) and vapours (some 100 ppb) suggesting that these promising results represent a credible first step towards the prevention of pressure ulcers that will require further massive data collection, machine learning and artificial intelligence to build a sharper diagnostic.

Acknowledgements

The author would like to acknowledge Hervé BELLÉGOU for his input in the e-nose device development.

References

[1]. F. A. R. Mota, M. L. C. Passos, J. L. M. Santos, and M. L. M. F. S. Saraiva, Comparative analysis of electrochemical and optical sensors for detection of chronic wounds biomarkers: A review, *Biosens Bioelectron,* Vol. 251, No. 116095, May 2024, pp. 1–18.

[2]. M. T. Tran, A. Kumar, A. Sachan, M. Castro, W. Allegre, and J. F. Feller, Emerging strategies based on sensors for chronic wound monitoring and management, *Chemosensors*, Vol. 10, No. 311, 2022, pp. 1–29.

[3]. M. S. Brown, B. Ashley, and A. Koh, Wearable technology for chronic wound monitoring: Current dressings, advancements, and future prospects, *Front Bioeng Biotechnol*, Vol. 6, No. 47, Apr. 2018, pp. 1–21.

[4]. I. Texier et al., SWAN - iCare project: Towards smart wearable and autonomous negative pressure device for wound monitoring and therapy, in *Proceedings of the 4th International Conference on Wireless Mobile Communication and Healthcare - 'Transforming Healthcare Through Innovations in Mobile and Wireless Technologies' (MOBIHEALTH 2014)*, 2015, pp. 357–360.

[5]. Y. Liu, M. Pharr, and G. A. Salvatore, Lab-on-skin: A review of flexible and stretchable electronics for wearable health monitoring, *ACS Nano*, Vol. 11, No. 10, Oct. 2017, pp. 9614–9635.

[6]. M. L. Hammock, A. Chortos, B. C. K. Tee, J. B. H. Tok, and Z. Bao, 25th anniversary article: The evolution of electronic skin (e-skin): A brief history, design considerations, and recent progress, *Advanced Materials,* Vol. 25, No. 42, Nov. 2013, pp. 5997–6038.

[7]. M. Sharifuzzaman, et al., Smart bandage with integrated multifunctional sensors based on MXene-functionalized porous graphene scaffold for chronic wound care management, *Biosens Bioelectron,* Vol. 169, August 2020, p. 112637.

[8]. Q. Zeng, X. Qi, G. Shi, M. Zhang, and H. Haick, Wound Dressing: From Nanomaterials to Diagnostic Dressings and Healing Evaluations, *ACS Nano*, Vol. 16, No. 2, Feb. 2022, pp. 1708–1733.

[9]. R. C. Webb et al., Thermal transport characteristics of human skin measured in vivo using ultrathin conformal arrays of thermal sensors and actuators, *PLoS One*, Vol. 10, No. 2, Feb. 2015, p. e0118131.

[10]. D. H. Kim et al., Epidermal Electronics, *Science* (1979), Vol. 333, No. 6044, Aug. 2011, pp. 838–843.

[11]. T. T. Tung, C. Robert, M. Castro, J. F. Feller, T. Y. Kim, and K. S. Suh, Enhancing the sensitivity of graphene/polyurethane nanocomposite flexible piezo-resistive pressure sensors with magnetite nano-spacers, *Carbon N Y*, Vol. 108, Nov. 2016, pp. 450–460.

[12]. S. Nag, M. Castro, V. Choudhary, and J. F. Feller, Boosting selectivity and sensitivity to biomarkers of Quantum Resistive vapour Sensors used for volatolomics with nanoarchitectured carbon nanotubes or graphene platelets connected by fullerene junctions, *Chemosensors,* Vol. 9, No. 66, Mar. 2021, pp. 1–15.

[13]. S. Nag, M. Castro, V. Choudhary, and J.-F. Feller, Boosting Selectivity and Sensitivity to Biomarkers of Quantum Resistive Vapour Sensors Used for Volatolomics with Nanoarchitectured Carbon Nanotubes or Graphene Platelets Connected by Fullerene Junctions, *Chemosensors*, Vol. 9, No. 4, Mar. 2021, p. 66.

[14]. J. F. Feller, et al., Novel architecture of carbon nanotube decorated poly(methyl methacrylate) microbead vapour sensors assembled by spray layer by layer, *J Mater Chem*, Vol. 21, No. 12, 2011, pp. 4142–4149.

[15]. T. T. Tung, C. Robert, M. Castro, J. F. Feller, T. Y. Kim, and K. S. Suh, Enhancing the sensitivity of

graphene/polyurethane nanocomposite flexible piezo-resistive pressure sensors with magnetite nano-spacers, *Carbon N Y*, Vol. 108, Nov. 2016, pp. 450–460.

[16]. A. N. Thomas et al., Novel noninvasive identification of biomarkers by analytical profiling of chronic wounds using volatile organic compounds, *Wound Repair & Regeneration*, Vol. 18, No. 4, May 2010, pp. 391–400.

(017)

A Markov Chain-based Data Augmentation to Improve Balance and Posture Stability in Spinal Cord Injury Rehabilitation

Vibhuti [1,2], Neelesh Kumar [1,2] and Chitra Kataria [3]
[1] Academy of Scientific and Innovative Research (AcSIR), Ghaziabad-201002, India
[2] CSIR – Central Scientific Instruments Organisation (CSIR-CSIO), Chandigarh-160030, India
[3] ISIC – Indian Spinal Injuries Centre, New Delhi-110070, India
Tel.: + 91-1722672278
E-mails: vibhuti.csio19a@acsir.res.in; neel5278@csio.res.in; chitrakataria@yahoo.com

Summary: Spinal cord injuries (SCIs) often lead to significant limitations in daily activities, including difficulties with posture, balance, and brain-spine connectivity. Virtual Reality (VR) therapeutic intervention can improve motor function and lessen neuropathic pain through rehabilitation. The data collected from the designed and developed VR rehabilitation system on motor functions addresses standing balance in incomplete SCI individuals. However, adequate data in clinical and e-rehabilitation settings remain a challenge. To address this, Markov chain-based data augmentation technique is employed to generate simulated data emulating original parameters. 30 SCI individuals were divided into experimental (EG) and control groups (CG) and assessed before and after VR intervention using different outcome measures (Berg Balance Scale (BBS), Activities Specific Balance Confidence (ABC), Walking Index for Spinal Cord Injury (WISCII)). Results indicate that 20 % of individuals in the EG and 53.33 % in the CG showed significant improvements in functional tasks based on the BBS. Moreover, on the ABC, 13.33 % of the EG and 33.33 % of the CG exhibited improved balance confidence and daily living activities. Regarding the WISCII scale, 46.66 % of the EG showed better walking impairment results than 80 % in the CG. Thorough statistical techniques and comparisons enhance the validity of rehabilitation outcomes.

Keywords: Rehabilitation, Virtual reality, Spinal cord injury, Data augmentation, Markov chain.

1. Introduction

Neuromotor impairments, predominantly Spinal Cord Injury (SCI), impact 250000 and 500000 individuals every year, according to the World Health Organization [1] (WHO). The manifestations of impairment are characterized by sensory symptoms such as pain, numbness, paresthesia, motor symptoms, viz. weakness, paralysis [2, 3], spasticity, and autonomic symptoms like bradycardia, hypotension, hypothermia, and erectile dysfunction. However, maintaining balance while standing [4] is highly dependent on it. Enhancing one's ability to balance when standing and control one's weight is essential for better activities. The adaptive movements of the extremities and the amalgamation and modulation of information from the somatosensory, visual, and vestibular systems help to maintain balance [1, 5]. The somatosensory organs are connected via the spinal cord. Balance difficulties [6] are caused by injury to the spinal cord. The impairment of proprioception following a central nervous system injury appears to have an impact on the balance of individuals with SCI. Nevertheless, research has been done too far to identify the impacted components of the balance system and the degree of proprioception impairment. Due to this, both individuals and society are burdened financially by the cost of medical treatment and management [7]. Affording timely rehabilitation [8, 9] of an affected extremity of an individual is a feasible solution. The most common approach to evaluating difficulties with movement is evaluating daily living activities.

The optimal approach to activating the affected extremities employs a virtual reality (VR)-based rehabilitation process [10]. Due to technological advancements, the adoption of VR is currently growing. VR technologies are intriguing for various rehabilitation treatments and research fields [11, 12]. Many VR-based lower extremity rehabilitation technologies are focused primarily on posture and balance [6] to assist a diverse spectrum of individuals. On the other hand, task-oriented movements acknowledged relatively less attention. The article focuses on task-oriented movement, incorporating VR technology in medical care to distract individuals from pain, automating motor therapy by providing physical support for restricted motions and increasing motivation [13]. The hypothesis that using VR rehabilitation technology reduces pain and improves lower extremity motor function in individuals with incomplete SCI was assessed by pre and post-clinical assessments. Balance control was clinically evaluated using BBS (Berg Balance Scale), ABC (Activities-Specific Balance Confidence), and WISCII (Walking Index for Spinal Cord Injury) [9].

In the study, the rate of the severity of SCI in individuals is examined using the American Spinal Injury Association Impairment Scale (ASIA). There are some constraints to the collected data to compensate for the unbalanced database due to low availability. Considering the above limitation, the data augmentation approach has been introduced to generate new data samples based on statistical models. However, this article introduces Markov Chain-based

augmentation [14] to generate stimulated data from original data. With the help of this approach, the systematically simulated dataset is generated to enhance the stability and dependability of our research.

2. Related Work

A comprehensive review of the current state of research is discussed in this section. Prior studies have shown that data augmentation in rehabilitation with different methods is an area of the current research. As Isam Biukhennoufa, et al. [15] reported, post-stroke e-rehabilitation assessment with wearable health monitoring devices anticipated the Time Series Generative Adversarial Network (TS-GAN) model. It increased the activity recognition dataset's classification performance from 48.73 % to 90.8 % and the ARAT dataset's classification performance from 63 % to 98.2 %. Another author, Chengxuan Qin, et al. [16] demonstrated a spatial variation generation algorithm for Motor Imagery (MI) data augmentation. Scalable Vector Graphics (SVG) performs better than other data augmentation algorithms in terms of improvement. Additional findings from the ablation research confirm that every SVG element works. Using various sample sizes in the CG demonstrates that the SVG algorithm continuously raises the area under the curve, with increases ranging from roughly 0.02 to 0.15. However, the simulated dataset generated by a dual encoder variational autoencoder-generative adversarial network (DEVAE-GAN), as discussed by Chenxi Tian et al. [17], possesses a 97.21 % average accuracy across 15 individuals, a 5 % increase over the original dataset, and it is demonstrated that the generated data and the original data distribution are identical. Another finding by Yu Xie et al. [18] is that motor imagery electroencephalogram (EEG) signals with data augmentation technique were quite successful in raising the accuracy of training Visual Geometry Group (VGGNet), EEGNet, and the suggested model. The average accuracy of the suggested MI-EEG image classification approach is 97.61 %. This method is improved by designing two distinct Convolutional Neural Network (CNN) scales for the time domain and Continuous Wavelet Transform (CWT) mapping maps, resulting in a more thorough feature extraction. The approach can enhance the classification performance of Brain-Computer Interface (BCI) systems designed for individuals with disabilities and MI-based BCIs. Kang Yin et al. [19] illustrated a framework for target-centered subject transfer as a method for augmenting data by using a generative model. A method discussed by Seong Jin Bang et al. [20], human activity recognition in rehabilitation activities presents an imbalance problem. STO-CVAE has done this to improve the preciseness of the classification of disabilities. During exercise, it can promptly predict emergency halt scenarios based on the impairment type. Continued research in this area employed by Ping-Huan Kuo et al. [21] represented extensive datasets, and model instability caused by SMOTE data augmentation during training can potentially be lessened. The suggested system can schedule occupational therapy appointments and diagnose dementia. However, three data augmentation techniques – SMOTE, NearMiss, and Markov chain-based augmentation – were investigated in this study to eliminate the class imbalance in rehabilitation data. Markov chain-based augmentation formulated new variables to replicate the sequential patterns recognized in the original dataset [22]. In comparison, SMOTE [23, 24] and Near Miss [25] approaches emphasize oversampling and under-sampling, respectively. Markov chain-based augmentation preserves the data structure by representing the time progression of affected motor movements and therapy sessions through statistical modeling of data development processes. Markov chain-based augmentation provides a more reliable method for synthesizing realistic synthetic samples than SMOTE and NearMiss, especially when sequential dependencies are essential, like rehabilitation data analysis [26, 27].

The fundamental objective of this research is to generate a simulated dataset from the original dataset collected from incomplete SCI individuals who suffered from standing balance or posture impairment. The collected data has a small sample size and uses a data augmentation method that can aid medical professionals. This analysis aims to educate academics and practitioners on the best methods for augmenting imbalanced datasets in rehabilitation research by analyzing before and after treatment.

3. Materials and Method

3.1. Database Description

The information was collected from ISIC-Indian Spinal Injuries Center, New Delhi, India Institutional Review Board / Independent Ethics Committee. The records in the database originated from 30 standing balance SCI individuals comprising an EG (15 subjects) and a CG (15 subjects). However, this data was further categorized based on the AIS C and AIS D scale. The data was collected with the aid of VR therapeutic intervention as developed by CSIR-CSIO, Chandigarh. This semi-immersive VR activity has been integrated with health monitoring devices (postural stability assessment board and Inertial measurement units). The data was accumulated at the study's beginning upon individual enrolment and the completion of the rehabilitation program.

3.2. Clinical Assessment Protocol

The clinical evaluation tools employed to evaluate various aspects of an individual's daily functioning, medical conditions, and rehabilitation results can be used to characterize the clinical assessment used in the study. This involves delivering details about the

standardized clinical evaluation protocols, including the particular tests, scales, or questionnaires used to evaluate the relevant parameters, like motor function, balance, posture, and overall progress accomplished during rehabilitation. A systematic neurological examination called the ASIA measures motor and sensory function in individuals with SCI. ASIA A (complete), ASIA B (sensory incomplete), ASIA C (motor incomplete: more than half), ASIA D (motor incomplete: at least half), and ASIA E (normal) are the categories used to classify the severity of injuries. Regions from C2 to C4, C5 to T1, T2 to T12, L1 to L5, and S1 to S5 are among the sensory testing locations. According to the BBS, the quantification of static balance and fall risk ranges from 0 to 56, having 14 items balance measure administrated 15 to 20 minutes ranges from point 0 (lower level of activity) to point 4 (higher level of activity). The functional balance has been categorized into three stages: stage I (0-20), stage II (21-40), and stage III (41-56). However, the ABC has been used to assess confidence in balance for the activities of daily living, a 16-item administered 5 to 10 minutes ambits from 0 (no confidence) to 100 (complete confidence). It was segmented into three stages: stage I (0-49), stage II (50-79), and stage III (80-100). For the mobility assessment, assistive devices were assessed by one item administrated 5 minutes ambits from 0 (incapable of ambulation) to 20 (capable of ambulation without assistive devices); the WISCI assessment was used. The three different categories for evaluation were stage I (0-5), stage II (6-10), and stage III (11-20), addressing the identified challenges effectively.

3.3. Data Augmentation Process

An illustration in the mathematical theory of the Markov chain-based data augmentation method based on: Let us assume

$$X = \{X_1, X_2, X_3, \dots\dots, X_n\} \quad (1)$$

original dataset with n samples, where X_i represents the i^{th} sample [28].

Transition Probability Matrix (P)

Implement a transition probability matrix P with dimensions m × m, where m indicates the number of classes in the dataset [29].

$$\mathrm{P} = \begin{matrix} P_{11} & P_{12} \dots\dots & P_{1m} \\ P_{21} & P_{22} \dots\dots & P_{2m} \\ P_{m1} & P_{m2} \dots\dots & P_{mm} \end{matrix}, \quad (2)$$

where P is the transition matrix, P_{ij} is the transition probability from state i to state j.

Stationary Distribution (π)

Determine the states' and classes' long-term probability distribution in the Markov chain by computing the stationary distribution vector, or π.

By figuring out the equation:

$$\pi \mathrm{P} = \pi, \quad (3)$$

subject to the constraint

$$\sum_{i=1}^{m} \pi_i = 1 \quad (4)$$

Data Augmentation

The transition matrix P simulates transitions between clinical assessment scores to simulate new clinical assessment scores. From an initial clinical assessment score x_t, the next assessment score, x_{t+1} was determined by sampling from the probability distribution defined by the row corresponding to score x_t in the transition matrix. This process has been repeated to generate a sequence of clinical assessment scores. Mathematically, the equation for data augmentation in this context can be represented as:

$$x_{t+1} = x_t * P, \quad (5)$$

where x_t is the current assessment score at time t, x_{t+1} is the simulated assessment score at time t+1, P is the transition matrix representing the transition probabilities between different assessment scores.

This equation represents the process of simulating transitions between assessment scores according to the transition probabilities defined by the transition matrix P.

Perhaps the original dataset can be used to estimate the transition probabilities P_{ij}. The augmentation procedure aims to create simulated samples [30] with Markov chain transitions that introduce variety while closely resembling the underlying class distribution in the original dataset.

The transition probability matrix P and the stationary distribution π can be computed theoretically in the following ways [31]:

Transition Probability Matrix:

$$p_{ij} = \frac{Number\ of\ transitions\ from\ class\ i\ to\ class\ j}{Total\ number\ of\ transitions\ from\ class\ i} \quad (6)$$

Stationary Distribution

$$\pi \mathrm{P} = \pi, \quad (7)$$

$$\sum_{j=1}^{m} \pi_j p_{ji} = \pi_i, \quad (8)$$

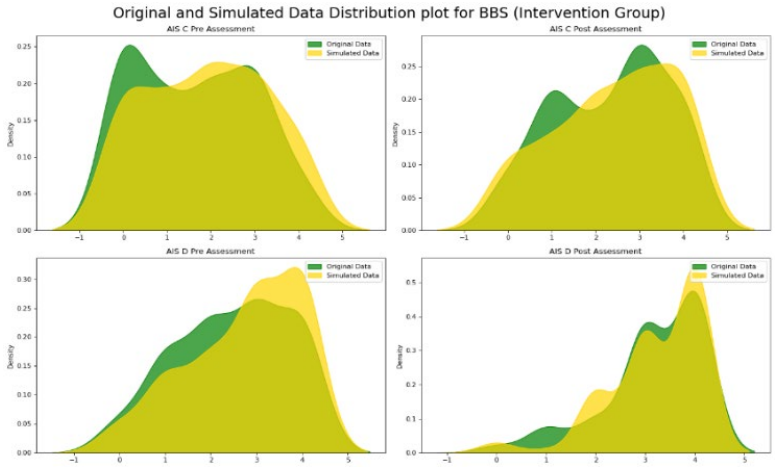
$$\begin{gathered} \pi_1 p_{11} + \pi_2 p_{21} + \pi_3 p_{31} + \cdots + \\ + \pi_m p_{m1} = \pi_1, \\ \pi_1 (p_{11} - 1) + \pi_2 p_{21} + \pi_3 p_{31} + \cdots + \\ + \pi_m p_{m1} = 0, \\ \pi_1 (p_{11} - 1) + \pi_2 (p_{21} - 1) + \pi_3 p_{31} + \cdots + \\ + \pi_m (p_{m1} - 1) = 0 \end{gathered} \quad (9)$$

The Markov Chain-Based Data Augmentation approach produces simulated data points that closely resemble the original dataset and successfully

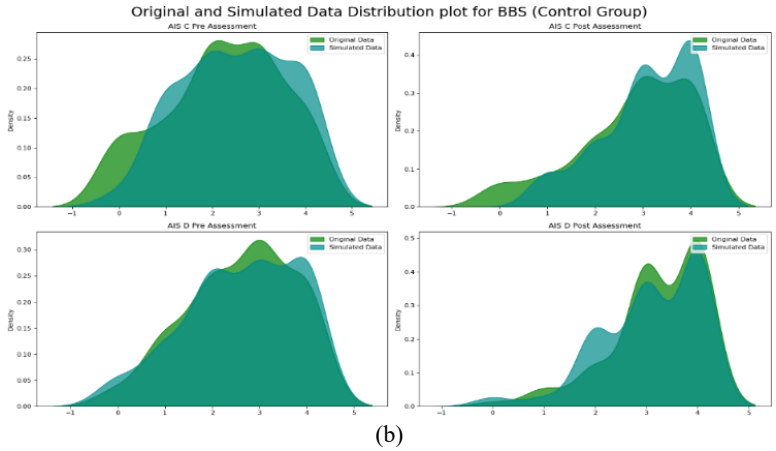
represent the underlying data distributions the state probability vector and the transition probability matrix.

4. Results and Discussion

The results after data augmentation [15, 16] are presented and discussed in this section. The significant methodological procedures to optimize the research design and represent the statistical robustness of the analysis include balancing the dataset and categorizing individuals according to their AIS C and D levels. It was known that BBS, ABC, and WISCII assessments employed discrete values to illustrate specific balancing functions. Markov chains are superior at showcasing state transitions. Transitions between a score of 10 and 20 on the ABC indicate an overall improvement in the balancing function rather than a progressive, continuous improvement. The gap between the discrete nature of the assessment and the model's assumption of continuous transitions may cause synthesized data points to be generated that are unrealistic [28] and fall between integer values, which could minimize the augmentation's usefulness for this particular application. To deliver synthetic data for assessments performed before and after intervention in individuals with standing balance, the Python code pertains to a Markov chain-based data augmentation technique [14, 22]. The simulated data samples are generated for the AIS C and AIS D categories to ensure that the quantity of synthetic and original samples were balanced. The original and synthetic data samples are finally shown for validation and comparison. The study affirmed balanced representation by including 15 participants in each group, with 8 AIS C and 7 AIS D individuals in the EG and 9 AIS C and 6 AIS D individuals in the GG, as illustrated in Fig. 1, respectively. Following data augmentation, the analysis demonstrated notable outcomes preferring the CG over the EG.



(a)

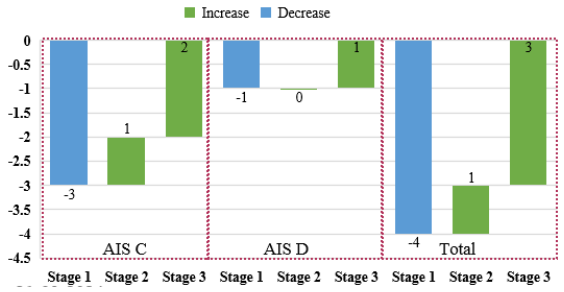


(b)

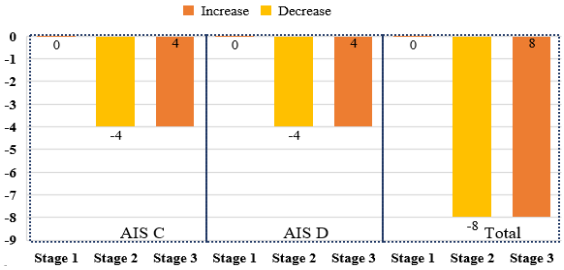
Fig. 1. (a) Original and Simulated Data Distribution Plot for EG; (b) Original and Simulated Data Distribution Plot for CG.

As illustrated in Fig. 2, none of the participants fell into Stage I after generating synthetic data from the BBS assessment. One participant fell into Stage II, while two fell into Stage III with AIS C. Only one fell into Stage III for the AIS D EG as in Fig. 2 (a). Similarly, four participants fell into Stage III for the CG for both AIS C and AIS D. Notably, the balanced dataset indicated significant results favoring the CG in

Fig. 2(b), with eight participants showing improved balance activities, compared to three participants in the EG.



(a)



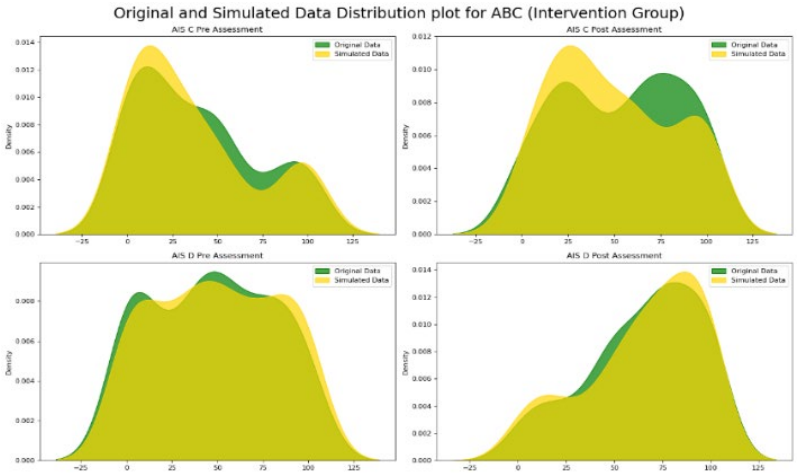
(b)

Fig. 2. (a) Individuals Improvements in EG; (b) Individuals Improvements in CG.

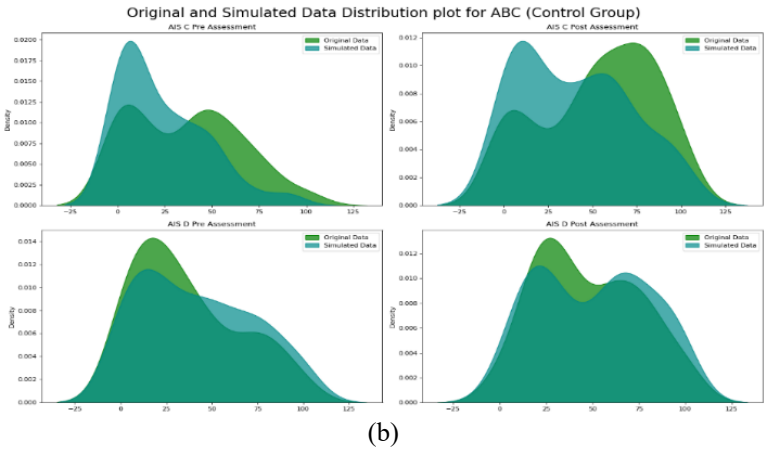
As depicted in Fig. 3, no individuals were categorized under Stage I post-augmentation with the ABC assessment. Two individuals transitioned to Stage II, while one moved to Stage III within AIS C. Only one moved to Stages II and III in the AIS D EG.

Similarly, within the CG, one individual transitioned to Stage II, while three moved to Stage III within AIS C. However, one individual in the AIS D CG moved to Stage II, and two transitioned to Stage III as in Fig. 4(a). Notably, the balanced dataset indicated a noteworthy advantage favoring the CG (Fig. 4(b)), with five individuals demonstrating enhanced balance activities compared to two individuals in the EG.

As observed in the data generated from the WISCII assessment in Fig. 5, none of the individuals remained in stages I & II following data augmentation. Three individuals advanced to Stage III, those classified under AIS C. In the AIS D EG, four individuals advanced to stage III in Fig.6(a). Similarly, within the CG, three individuals advanced to Stage III within AIS C. However, within the AIS D CG, nine individuals progressed to Stage III. Remarkably, the balanced dataset highlighted a significant advantage favoring the CG, with twelve individuals in Fig. 6(b) exhibiting improved walking impairment compared to seven participants in the EG.



(a)



(b)

Fig. 3. (a) Original and Simulated Data Distribution Plot for EG; (b) Original and Simulated Data Distribution Plot for CG.

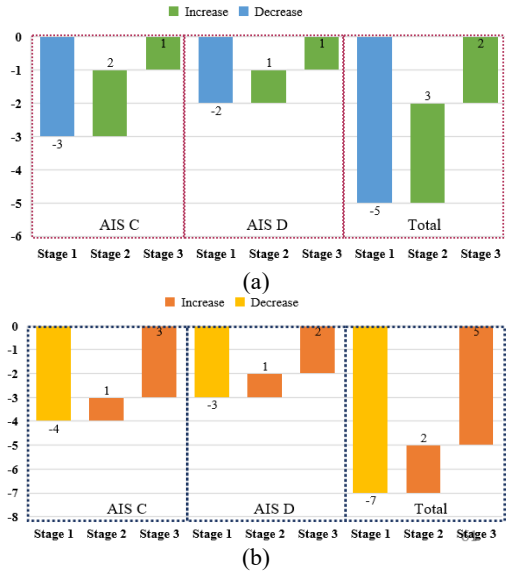


Fig. 4. (a) Individuals Improvements in EG; (b) Individual improvements in CG.

From these results, it has been concluded that the attributes of the analyzed dataset Markov chain-based data augmentation needed to be performed for our investigation. Discreteness has been shown by the findings, which consist of assessment scale scores for individuals with SCI. However, SMOTE [24] and NearMiss [25] were designed for continuous data and might not be able to manage the fundamental structure of discontinuous data well. The specified approach to data visualization demonstrates the data as a continuous function (i.e., a smooth line), although the real-world data points are discrete (i.e., whole numbers or specific categories). This provides key information on the progression of an individual rehabilitation and facilitates to identify underlying trends in the data and anticipate potential outcomes.

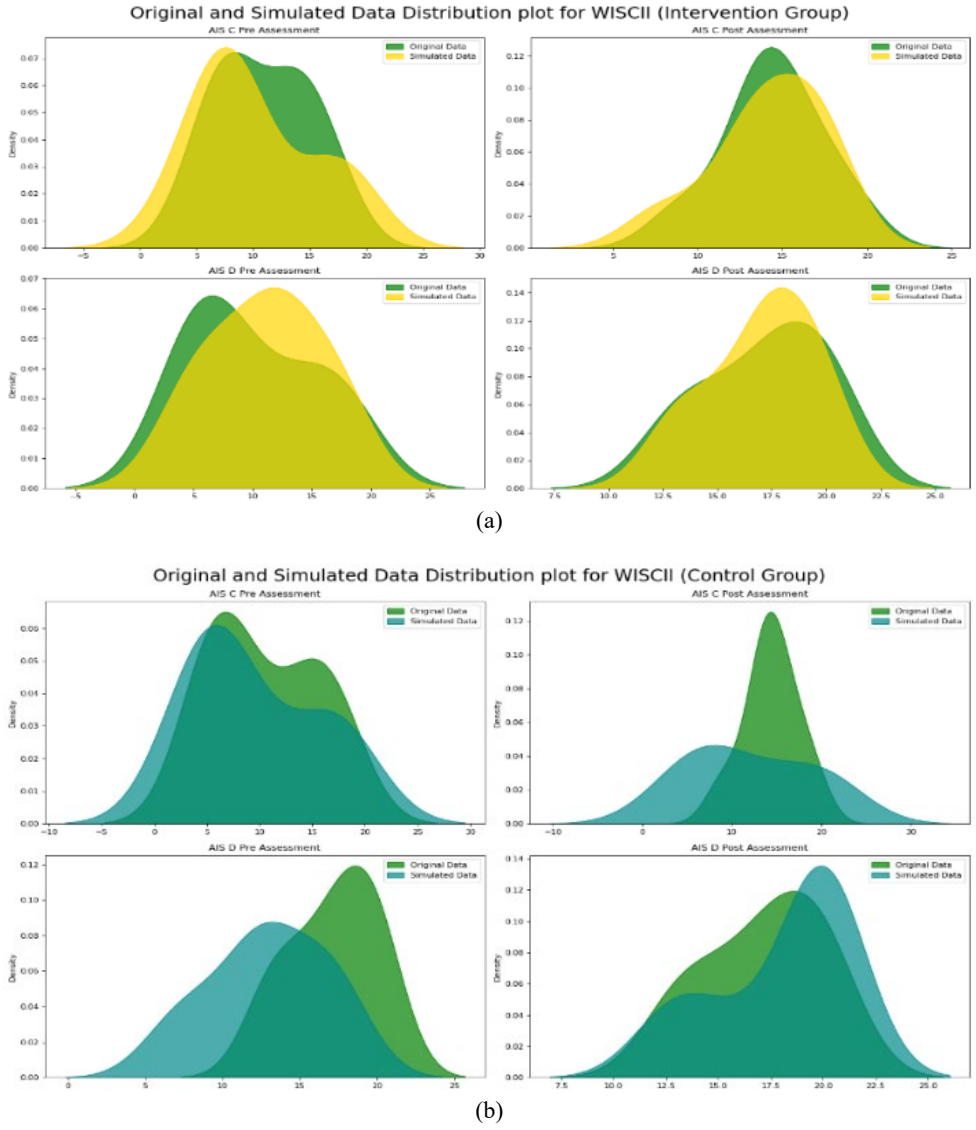


Fig. 5. (a) Original and Simulated Data Distribution Plot for EG; (b) Original and Simulated Data Distribution Plot for CG.

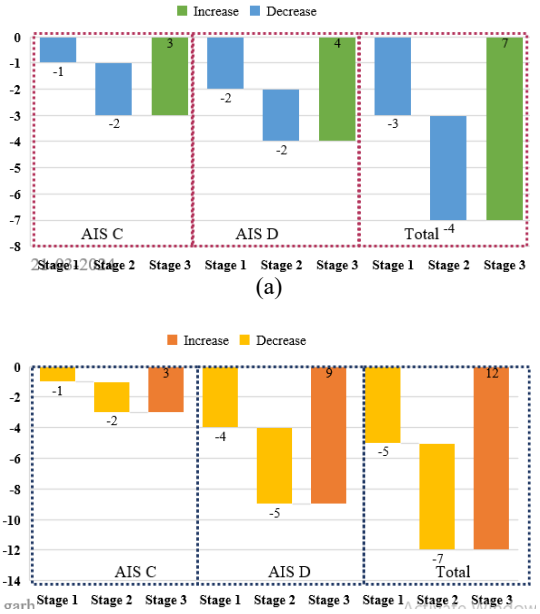


Fig. 6. (a) Individuals Improvements in EG; (b) Individual improvements in CG.

5. Conclusion

The study illustrated a statistical model approach of data augmentation to generate a synthetic dataset of standing balance SCI individuals. However, Markov chain-based data augmentation highlights overcoming the challenges of establishing random discrete data of assessment scales in the medical field. To balance out the AIS C and AIS D pre-post rehabilitation data and increase their relevance, stratified sampling processes were used. The EG and CG showed significant improvements in balance, confidence, and walking impairment. However, the CG was periodically outperformed by the balanced dataset on all assessment scales (BBS, ABC, and WISCII). It implies that compared to VR rehabilitation interventions, conventional rehabilitation produces significantly more significant improvements in functional performance. Of the 15 individuals in the EG, 20 % performed better on their balance, 13.33 % improved their balance confidence, and 46 % improved their walking ability. On the other hand, the same 15 individuals in the CG saw a 53.33 % advance in BBS, a 33.33 % advance in ABC, and an 80 % advance in WISCII. Furthermore, an aspect of the methodology is highlighted by the difference between the Markov chain model's assumption of continuous transitions and the discrete character of the assessment scale scores, which could result in the creation of fake synthetic data points. The study might need a more sufficient statistical ability to identify statistically significant group differences because there are only 15 individuals in each group. Examining alternative techniques tailored for discrete data could enhance the precision of forming synthetic data. However, using longitudinal data can convey a more detailed view of the effects of interventions by recording the steady shifts in performance over time. The resulting information provides you substantial knowledge on the current state of the individual's rehabilitation and facilitates you to recognize fundamental patterns in the data and predict future outcomes.

Acknowledgement

The authors acknowledge the Ministry of Social Justice and Empowerment (MSJE) and the Indian Council of Medical Research (ICMR) for providing us with the funds required to complete this research. We gratefully acknowledge the invaluable contributions of the Mr. Anuj Mishra, PhD student whose assistance was instrumental in this study.

References

[1]. G. Maresca, *et al.*, A novel use of virtual reality in the treatment of cognitive and motor deficit in spinal cord injury A case report, *Med. (United States)*, Vol. 97, Issue 50, 2018, e13559.

[2]. Y. Bin Oh, *et al.*, Efficacy of virtual reality combined with real instrument training for patients with stroke: a randomized controlled trial, *Arch. Phys. Med. Rehabil.*, Vol. 100, Issue 8, 2019, pp. 1400-1408.

[3]. A. Berton, *et al.*, Virtual reality, augmented reality, gamification, and telerehabilitation: Psychological impact on orthopedic patients' rehabilitation, *J. Clin. Med.*, Vol. 9, Issue 8, 2020, pp. 1-13.

[4]. S. Yoon, H. Son, Effects of full immersion virtual reality training on balance and knee function in total knee replacement patients: a randomized controlled study, *J. Mech. Med. Biol.*, Vol. 20, Issue 9, 2020, pp. 1-14.

[5]. L. I. E. Oddsson, R. Karlsson, J. Konrad, S. Ince, S. R. Williams, and E. Zemkova, A rehabilitation tool for functional balance using altered gravity and virtual reality, *J. Neuroeng. Rehabil.*, Vol. 4, 2007, pp. 1-7.

[6]. M. Goffredo, *et al.*, Non-immersive virtual reality telerehabilitation system improves postural balance in people with chronic neurological diseases, *J. Clin. Med.*, Vol. 12, Issue 9, 2023, 3178.

[7]. C. Gao, Y. Wu, J. Liu, R. Zhang, M. Zhao, Systematic evaluation of the effect of rehabilitation of lower limb function in children with cerebral palsy based on virtual reality technology, *J. Healthc. Eng.*, Vol. 2021, 2021, 6625604.

[8]. L. D. Duffell, S. Paddison, A. F. Alahmary, N. Donaldson, J. Burridge, The effects of FES cycling combined with virtual reality racing biofeedback on voluntary function after incomplete SCI: A pilot study, *J. Neuroeng. Rehabil.*, Vol. 16, Issue 1, 2019, pp. 1-15.

[9]. M. G. Maggio, M. Bonanno, A. Manuli, R. S. Calabrò, Improving outcomes in people with spinal cord injury: encouraging results from a multidisciplinary advanced rehabilitation pathway, *Brain Sci.*, Vol. 14, Issue 2, 2024, 140.

[10]. D. C. R. Papa, *et al.*, Cardiac autonomic modulation in response to postural transition during a virtual reality task in individuals with spinal cord injury: A

crosssectional study, *PLoS One*, Vol. 18, Issue 4, 2023, e0283820.

[11]. Vibhuti, N. Sharma, C. Kataria, N. Kumar, S. Walia, M. Singh, Effectiveness of virtual reality-based rehabilitation of osteoarthritis undergoing total knee arthroplasty: a pre-post study, *IETE J. Res.*, 2023.

[12]. V. Vibhuti, N. Kumar, C. Kataria, Efficacy assessment of virtual reality therapy for neuromotor rehabilitation in home environment: a systematic review, *Disabil. Rehabil. Assist. Technol.*, Vol. 18, Issue 7, 2023, pp. 1200-1220.

[13]. R. Maskeliūnas, R. Damaševičius, T. Blažauskas, C. Canbulut, A. Adomavičienė, J. Griškevičius, BiomacVR: A virtual reality-based system for precise human posture and motion analysis in rehabilitation exercises using depth sensors, *Electron.*, Vol. 12, Issue 2, 2023, 339.

[14]. M. Sánchez-Manchola, L. Arciniegas-Mayag, M. Múnera, M. Bourgain, T. Provot, C. A. Cifuentes, Effects of stance control via hidden Markov model-based gait phase detection on healthy users of an active hip-knee exoskeleton, *Front. Bioeng. Biotechnol.*, Vol. 11, 2023, pp. 1-15.

[15]. I. Boukhennoufa, *et al.*, A novel model to generate heterogeneous and realistic time-series data for post-stroke rehabilitation assessment, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 31, 2023, pp. 2676-2687.

[16]. C. Qin, R. Yang, M. Huang, W. Liu, Z. Wang, Spatial variation generation algorithm for motor imagery data augmentation: increasing the density of sample vicinity, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 31, 2023, pp. 3675-3686.

[17]. C. Tian, Y. Ma, J. Cammon, F. Fang, Y. Zhang, M. Meng, Dual-encoder VAE-GAN with spatiotemporal features for emotional EEG data augmentation, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 31, 2023, pp. 2018-2027.

[18]. Y. Xie, S. Oniga, Classification of motor imagery EEG signals based on data augmentation and convolutional neural networks, *Sensors*, Vol. 23, Issue 4, 2023, 1932.

[19]. K. Yin, B. H. Lee, B. H. Kwon, J. H. Cho, Target-centered subject transfer framework for EEG data augmentation, in Proceedings of the *Int. Winter Conference Brain-Computer Interface (BCI'23)*, 2023, pp. 1-4.

[20]. S. J. Bang, M. J. Kang, M. G. Lee, S. M. Lee, STO-CVAE: state transition-oriented conditional variational autoencoder for data augmentation in disability classification, *Complex Intell. Syst.*, 2024.

[21]. P. H. Kuo, C. T. Huang, T. C. Yao, Optimized transfer learning based dementia prediction system for rehabilitation therapy planning, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 31, 2023, pp. 2047-2059.

[22]. M. Capecci, *et al.*, A Hidden Semi-Markov Model based approach for rehabilitation exercise assessment, *J. Biomed. Inform.*, Vol. 78, 2018, pp. 1-11.

[23]. H. Woldesellasse, S. Tesfamariam, Data augmentation using conditional generative adversarial network (cGAN): Application for prediction of corrosion pit depth and testing using neural network, *J. Pipeline Sci. Eng.*, Vol. 3, Issue 1, 2023, 100091.

[24]. W. Wang, T. W. Pai, Enhancing small tabular clinical trial dataset through hybrid data augmentation: combining SMOTE and WCGAN-GP, *Data*, Vol. 8, Issue 9, 2023, 135.

[25]. M. Li, *et al.*, A benchmark for cycling close pass near miss event detection from video streams, *arXiv Preprint*, 2023, arXiv:2304.11868.

[26]. N. Sinha, M. A. G. Kumar, A. M. Joshi, L. R. Cenkeramaddi, DASMcC: data augmented smote multi-class classifier for prediction of cardiovascular diseases using time series features, *IEEE Access*, Vol. 11, 2023, pp. 117643-117655.

[27]. L. M. Lim, S. Sathasivam, M. T. Ismail, A. Sufril, Comparison using intelligent systems for data prediction and near miss detection techniques, *Pertanika J. Sci. & Technol*., Vol. 32, Issue 1, 2024, pp. 365-394.

[28]. F. M. Omar, M. A. Sohaly, Susceptible – exposed – infectious model using Markov chains, *J. Nonlinear Math. Phys.*, 2024, pp. 1-17.

[29]. D. Mizutani, Infrastructure deterioration modeling with an inhomogeneous continuous time Markov chain: A latent state approach with analytic transition probabilities, *Computer-Aided Civil and Infrastructure Engineering*, Vol. 38, 2023, pp. 1730-1748.

[30]. P. Lencastre, M. Gjersdal, L. Rydin, A. Yazidi, Modern AI versus century-old mathematical models: How far can we go with generative adversarial networks to reproduce stochastic processes?, *Phys. D*, Vol. 453, 2023, 133831.

[31]. M. N. Hassan, M. S. Mahmud, K. F. Nipa, M. Kamrujjaman, Mathematical modeling and COVID-19 forecast in Texas, USA: a prediction model analysis and the probability of disease outbreak, *Disaster Med. Public Health Prep.*, Vol. 17, Issue 1, 2023, e19.

(018)

Identification of Nonlinearities using Wavelet Transform

A. Klepka
AGH University of Science and Technology, Department of Mechatronics and Robotics,
Al. Mickiewicza 30, 30-059 Krakow, Poland
E-mail: klepka@agh.edu.pl

Summary: The paper presents an application of time–frequency technique for nonlinearity detection. The wavelet transform is used to transform an impulse response of the system into a time-scale domain. Ridge and skeleton definitions of wavelet transform for backbone curve estimation have been used. A statistical approach for ridge detection of wavelet transform has been applied. Estimation of dynamic characteristics of the system with the use of envelope of signal response has been presented. The possibility of using the method for nonlinearity detection has been shown based on the properties of wavelet transform. The method also makes it possible to define the nature of these nonlinearities. MATLAB package is used as a numerical tool for the computation of wavelet transform and dynamical characteristics of the system. The algorithm has been tested on simulated data and data from a test bed with dry friction.

Keywords: Wavelet transform, Nonlinear systems, Backbone curve.

1. Introduction

Nonlinearities are effects that very often occur in mechanical systems. Structural, geometrical, and mechanical properties can cause it. Detecting nonlinearities and defining their properties are very important. Unlike linear systems, nonlinear systems can behave differently depending on excitation. Usually, it works unpredictably, e.g., small changes in initial conditions can lead to big changes in trajectory. Treatment of this kind of mechanical system as linear can be the reason for the incorrect results of the analysis. The paper attempts to use the wavelet transform for the detection of nonlinearities.

2. Continuous Wavelet Transform

The wavelet analysis is a method of signal decomposition. As a result of the wavelet analysis, elementary signals – so-called wavelets – are obtained in contradiction to the Fourier transform. Wavelet curves are continuous and oscillated with various duration times and spectrums. From the mathematical point of view, a wavelet transform of a signal *x(t)* can be defined as [3, 7]

$$(W_g x)(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t) g^*\left(\frac{t-b}{a}\right) dt \qquad (1)$$

Using properties of the wavelet transform [4], it can be proved mathematically that his kind of time-frequency analysis decouples the natural frequency contained in the signal. It has been explained graphically in Fig. 1.

The Morlet wavelet (Fig. 2) is one of the most widespread and often used functions in wavelet analysis. The Morlet wavelet is defined as

$$g(t) = e^{j2\pi f_0 |t|} e^{-\frac{|t|^2}{2}} \qquad (2)$$

Additional information about wavelet transform can be found [2-4, 6].

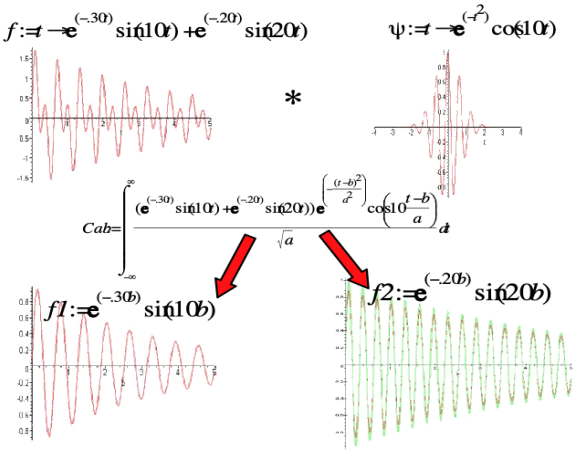


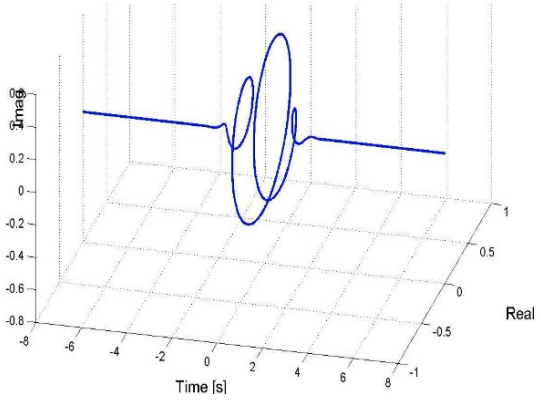
Fig. 1. Diagram of the analytical decoupling of natural frequencies.

3. Nonlinearities Detection using Wavelet Transform

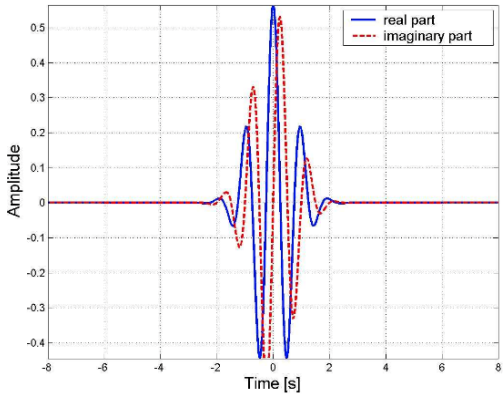
The procedure of nonlinearities detection is based on definitions of the skeleton and ridge of the wavelet transform. The ridge of the wavelet transform $(W x)(a,b)$ of the signal *x(t)* is a set of points *(a,b) g* in the wavelet transform domain, where phase $x(t)g_{a,b}(t)$ is stationary, which means that condition is fulfilled.

$$t_0(a,b) = b, \qquad (3)$$

where b is the translation (displacement) representing a region, a is the dilatation (expansion) or a scale parameter [3]. The skeleton of wavelet transform $W_g x(a,b)$ of signal $x(t)$ is the set of g coefficients of the wavelet transform calculated from ridge $W_g x(a_r(b),b)$, where a_r is the parameter of the ridge scale [6]. Graphically, the definition of the ridge and skeleton of the wavelet transform is shown in Fig. 3.



(a)



(b)

Fig. 2. An example of a Morlet wavelet is a) In an imaginary domain, and b) In a real and imaginary part.

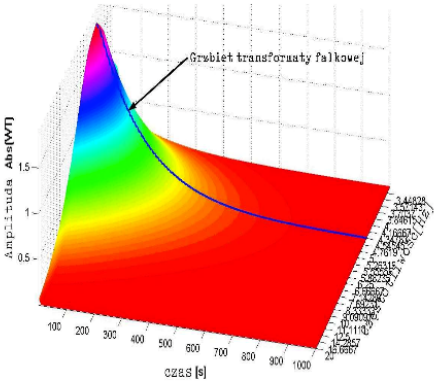
Based on the properties of the wavelet transform presented above, the algorithm of nonlinearity identification has been created. A diagram of the method is presented in Fig. 4.

In the first stage of the method, the wavelet transform matrix coefficients is calculated. This matrix can be interpreted as an energy distribution of the signal in time–frequency domain. From this matrix, the ridge curve is estimated. Detection of the ridge curve leads to an estimation of the skeleton of the wavelet transform. Based on the skeleton, the envelope of the given frequency component can be

For the linear signals, the envelope function can be written as estimated.

$$A(t) = A_0 e^{-\xi\omega_n t} \qquad (4)$$

a)



b)

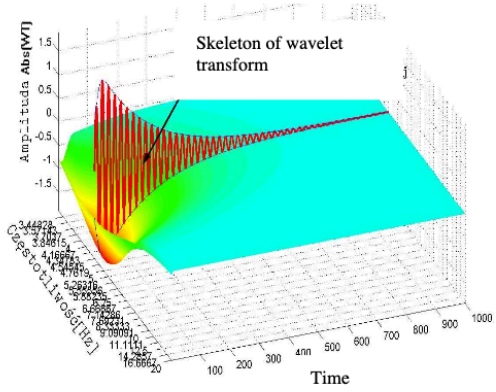


Fig. 3. a) Ridge of wavelet transform; b) Skeleton of wavelet transform.

Based on this, the system's modal parameters can be estimated. For nonlinear systems, the envelope function will depend on the type of damping and stiffness nonlinearities. Using ridge and skeleton definition, the characteristic called the Backbone curve can be determined. This characteristic shows the dependencies between the natural frequency and envelope functions of the system's impulse response. The backbone curve doesn't depend on envelope function and has a constant value for linear systems.

Using the curve fitting method for estimated characteristics, modal parameters of nonlinear systems can be estimated. The characteristics for linear and nonlinear systems presented are shown in Fig. 5.

The main problem with this method is estimating the wavelet transform's ridge curve. A method based on maximal values of wavelet coefficients for every section of the matrix in the time domain is often used [2, 6]. This method can work properly only if the signal-to-noise ratio has a big value. Otherwise, it is possible to appear local maxima connected with noise. In this case, the result of the analysis can be incorrect. For this reason, the algorithm of detection ridge of wavelet transform is based on the statistical approach of scalograms. The base of this method is the digitization of the wavelet coefficients matrix and the estimation of a three-dimensional histogram of these coefficients.

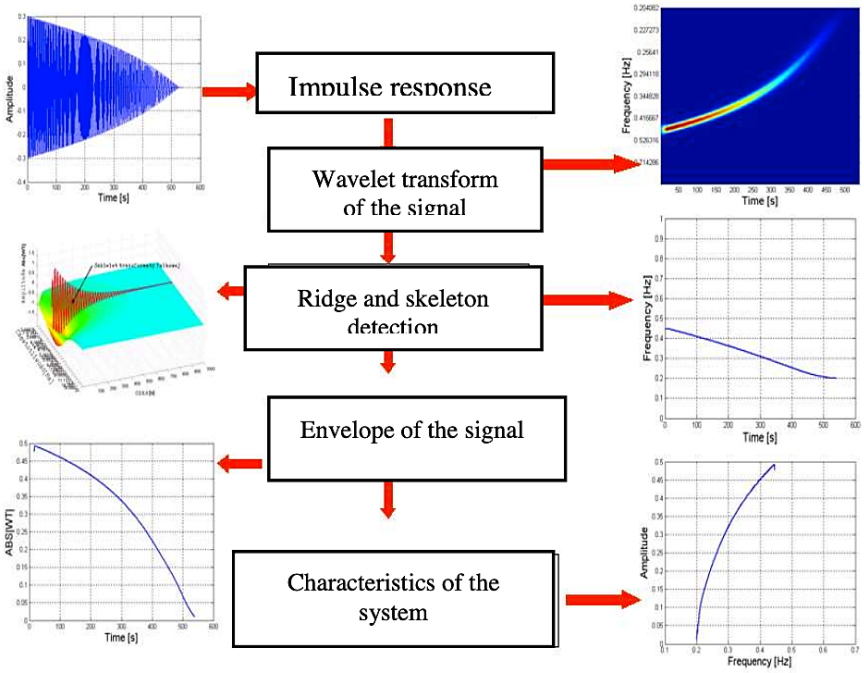


Fig. 4. Diagram of the nonlinearities identification with wavelet transform.

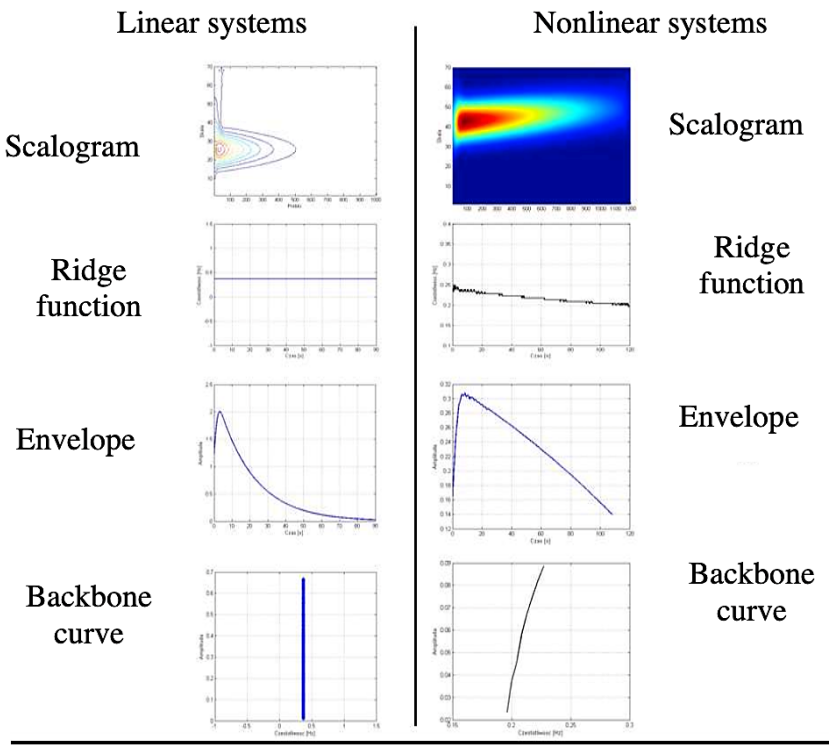


Fig. 5. Comparison of characteristics for linear and nonlinear systems.

This operation can effectively eliminate local maximums caused by noise. A comparison of results can be shown in Fig. 6 and Fig. 7.

Estimated ridge curve and envelope function for a given natural frequency are possible to determine amplitude–frequency characteristic (Backbone curve). The backbone curve gives information about stiffness nonlinearities in the system. The impulse response envelope function gives information about types of damping in the system. Analytical functions for different types of damping and stiffness can be found [1, 5].

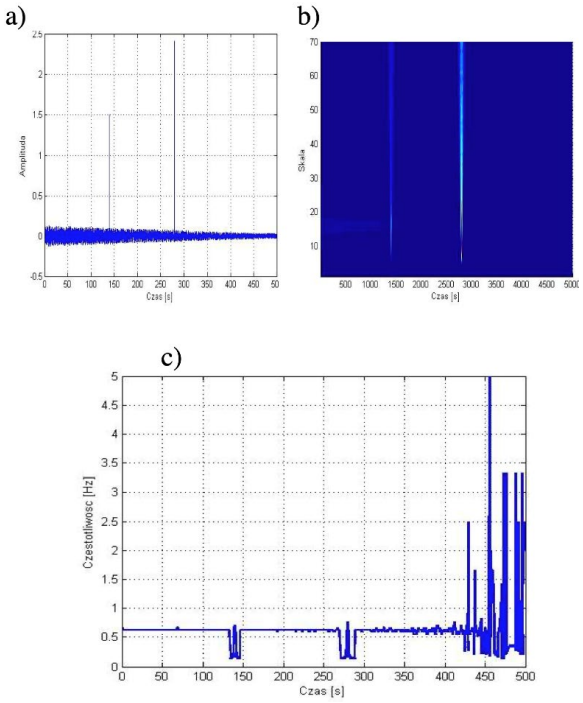


Fig. 6. a) Analyzed signal; b) Scalogram of the signal; c) Ridge curve obtained from scalogram ("maximum" approach).

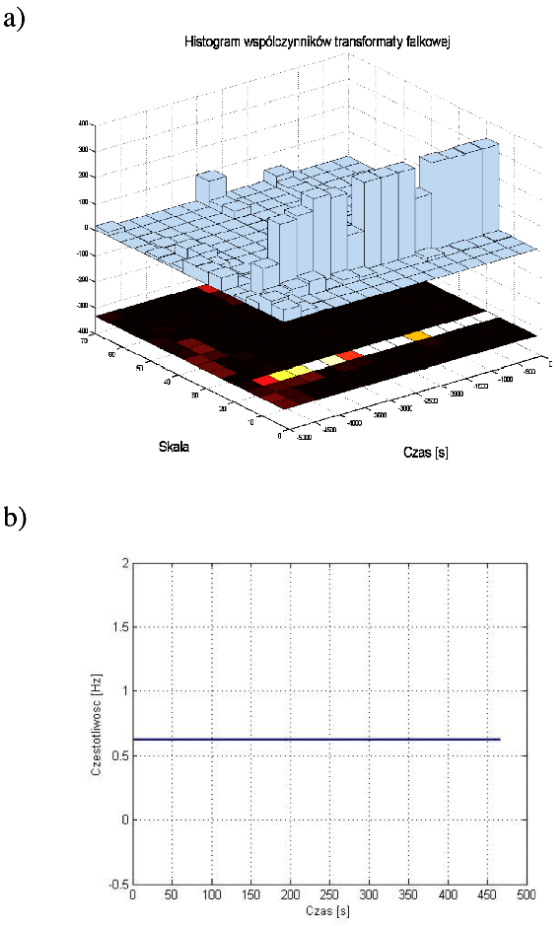


Fig. 7. a) Three – dimensional histogram of wavelet transform; b) Ridge curve obtained from scalogram ("statistical" approach).

4. Numerical Verification

Numerical verification of proposed algorithms has been carried out. Response signal of nonlinear single degree of freedom system with dry friction and cubic stiffness has been described by formula

$$\left(\frac{d^2}{dt^2}y(t)\right) + 0.001signum\left(\frac{d}{dt}y(t)\right) + 0.16\pi^2 y(t) + 100yy(t)^3 \quad (5)$$

Additionally, the signal was disrupted by noise. Time history of the signal, scalogram and ridge curve are presented on Fig. 8.

In the next step the character of nonlinearities has been determined. It has been done by curve fitting method, using known dependences between envelope function and type of damping.

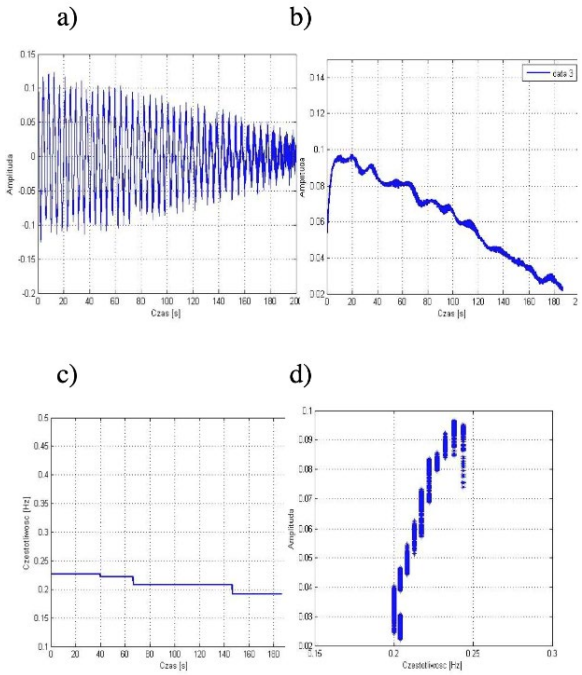


Fig. 8. a) Analyzed signal; b) Envelope function; c) Ridge curve; d) Backbone curve.

As a criterion Root Mean Squares Error has been applied. Result of analysis compare in the Tables 1-3 below.

Table 1. Modal parameters of the system.

Parameter	Real value	Identified value
ω[rad]	1,25	1,25
c[Ns/m]	0,001	0,00094
k1[N/m]	1,57	1,57
k3[N/m]	100	88,95

Table 2. Results (stiffness).

$kx\ \omega_n+\frac{k}{2\omega_n}$	$kx^3\ \omega_n+\frac{3k}{8\omega_n}A^2$	$kx^5\ \omega_n+\frac{5k}{16\omega_n}A^4$	$kx^7\ \omega_n+\frac{35k}{128\omega_n}A^6$
	k3 = 88.95 RMSE: 0.00482	k3 = 1.446e+004 RMSE: 0.006138	k3 = 0.8419 RMSE: 0.02242

Table 3. Results (damping).

Quadratic friction $\dot{x}\lvert\dot{x}\rvert$	Dry friction $sigmum(\dot{x})$	Viscous damping $c\dot{x}$
f(x) = a/(x+b) a = 9.777 b = 93.31 RMSE: 0.007325	f(x) = p1*x + p2 p1 = -0.0004333 p2 = 0.09572 RMSE: 0.002665	f(x) = a*exp(b*x) a = 0.101 b = -0.006947 RMSE: 0.004987

The last stage of the verification was analysis of test stand (Fig. 9). The system's impulse response is presented in Fig. 10.

Using the method the characteristic of the system has been estimated (Fig. 11).

Estimation of modal parameters has been carried out. Identified parameters has been used for creation an analytical model of the signal in order to method verification. Results has been presented in the Table 4 and has been shown on the Fig. 12.

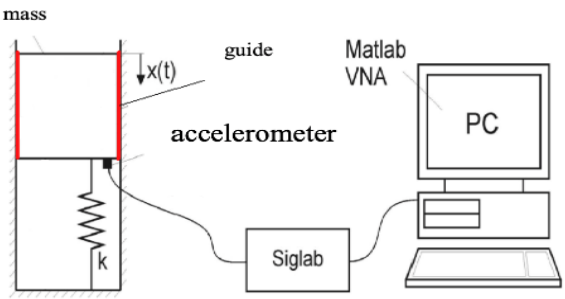


Fig. 9. Test stand.

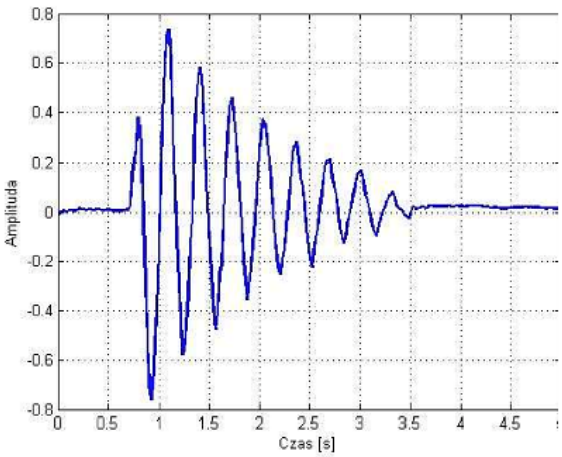


Fig. 10. Impulse response of the system.

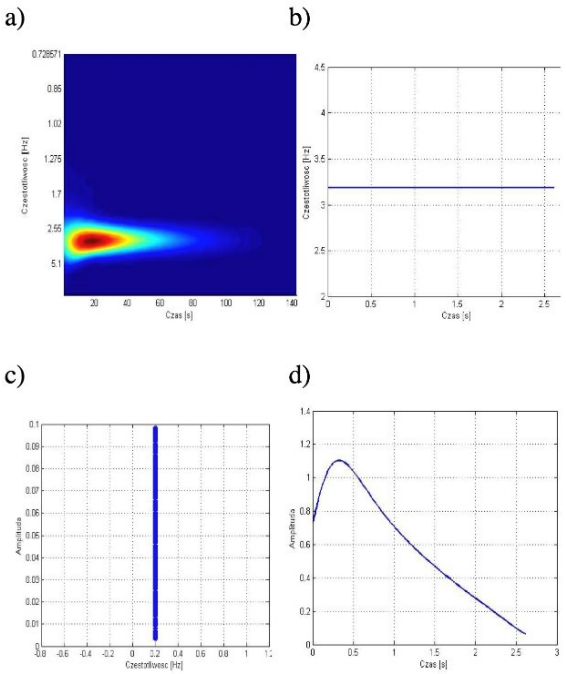


Fig. 11. a) Scalogram of the signal, b) Ridge curve, c) Backbone curve, d) Envelope.

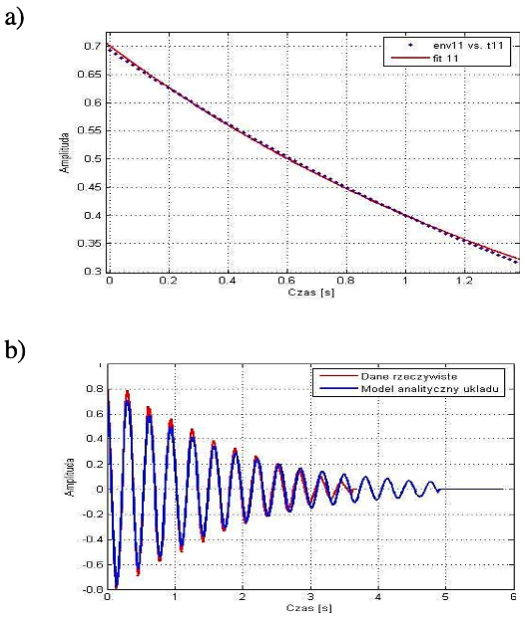


Fig. 12. a) Envelope of the signal; b) Comparison (model and signal).

Table 4. Identified parameters.

Parameter	Real value	Identified value
ω[rad]	19.3	20.1
c[Ns/m] (cẋ)	4,67	4,79
k[kN/m]	1,6	1,7

Assumed model was a linear model and result are correct only for initial part of the signal. The ending part of decay a difference between model and signal are visible. For this part of signal analysis with nonlinear model has been carried out. Result of estimation collected in the Table 5 and show Fig 13 and Fig 14.

Table 5. Identified parameters.

Parameter	Value
ω[rad]	20.1
$c[Ns/m](\ signum(\dot{x}))$	18
$k[kN/m]$	1,7

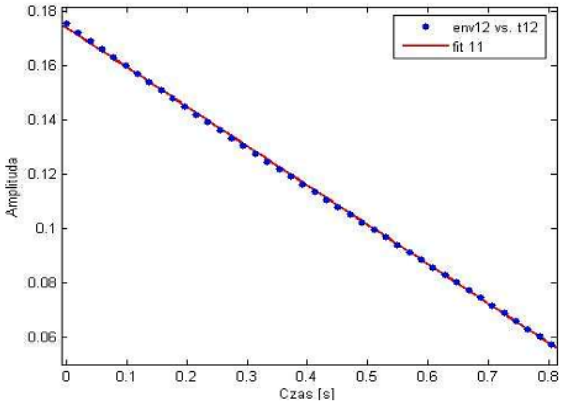


Fig. 13. Envelope of ending part of the signal.

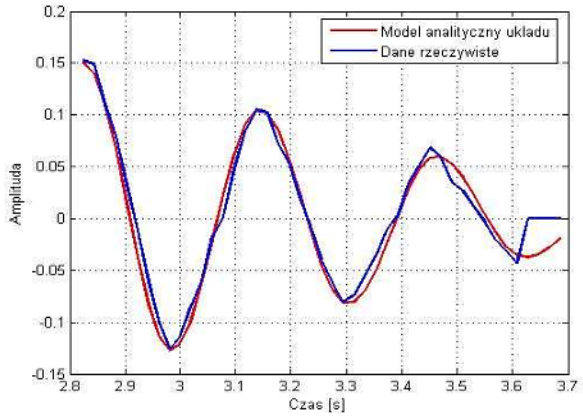


Fig. 14. Comparison (model and signal).

5. Conclusions and Further Works

The numerical analysis confirmed that applying wavelet transform make possible to detect nonlinearities of the mechanical systems and can define character of this nonlinearities. The method of ridge curve detection allows to identify dominant frequency components in the signal for noised signals. The algorithm estimates correct values of modal parameters.

It is necessary to carry out consider for natural frequency decoupling for multi degree of freedom systems. The next stage of the researches should be creation a method for nonlinear systems with operational excitation.

Acknowledgments

The author would like to acknowledge that this research was supported by AGH University of Krakow, Department of Robotics and Mechatronics subsidy.

References

[1]. Feldman M., Braun S., Analysis of typical nonlinear vibration system by using Hilbert transform, in *Proceedings of the 11th IMAC Conference*, 1993, pp. 799 – 805.

[2]. Joseph L., Ta Minh-Nghi, A wavelet-based approach for the identification of damping in non – linear oscillators, *International Journal of Mechanical Sciences,* 47, 205, pp. 1262 – 1281.

[3]. Klepka A., Uhl T., Zastosowanie transformaty falkowej do wyznaczania współczynnika łumienia konstrukcji, *Zeszyty Naukowe Politechniki Rzeszowskiej. Mechanika,* 60, 197, 2002, pp. 289-296.

[4]. Klepka, A., & Uhl, T., Zastosowanie transformaty falkowej do rozprzęgania postaci drgań oraz do wyznaczania współczynnika tłumienia, *Górnictwo odkrywkowe,* 45, 2-3, 2003, pp. 61-64.

[5]. Nayfeh, A. H., Perturbation methods in nonlinear dynamics, In Lecture Notes in Physics: Nonlinear Dynamics Aspects of Particle Accelerators, *Springer Berlin Heidelberg,* Berlin, Heidelberg, 2007, pp. 238-314.

[6]. Staszewski W. J., Identification of Non-Linear System Using Multi-Scale Ridges and Skeletons of the Wavelet Transform, *Journal of Sound and Vibration,* Vol. 214, 4, 1998, pp. 639-658.

[7]. Klepka, A., An application of wavelet transform for nonlinearities detection, *Diagnostyka*, 2008, pp. 107-112.

(019)

Theoretical Approaches to Signal Processing for Optimizing Blade Tip Timing Probes Arrangement

M. L. Mekhalfia [1], P. Procházka [1], R. Smid [2] and E. B. Tchawou Tchuisseu [1]
[1] Institute of Thermomechanics of the CAS, Dolejškova, 1402, 182 00, Praha, Czech Republic
[2] Czech Technical University in Prague, Technická 1902/2, 166, 27, Praha, Czech Republic
Tel.: + 420777975067
E-mail: m.mekhalfia@it.cas.cz

Summary: This theoretical study focuses on optimizing the positioning of blade tip timing probes to enhance the analysis of turbomachinery performance. The accurate measurement of blade vibrations is crucial for assessing the health and efficiency of turbines and compressors. By strategically placing the probes at specific locations along the blade tips, we aim to improve data collection precision and reliability. This paper explains the theoretical background of optimized approaches for determining the optimal probe positions, aiming to maximize data accuracy and minimize errors in vibration analysis.

Keywords: Blade tip timing, Error size, Sensor's arrangement, Vibration.

1. Introduction

In turbomachinery, accurate measurement and analysis of blade vibrations are essential for optimal performance and reliability. Blade tip timing probes have become valuable tool for monitoring blade vibrations in rotating machinery. Strategic probe positioning allows engineers to collect precise data on blade deflections and frequencies, facilitating the diagnosis of potential issues and optimization of overall system performance [1].

Optimizing blade tip timing probe placement involves considering factors such as blade geometry, machinery rotational speed, and desired measurement accuracy level. This theoretical study aims to explore these parameters' influence on optimal probe positioning and develop a framework to maximize probe effectiveness in detecting blade vibrations [2].

Previous research [3] has shown that the probes' positioning uncertainty has a moderate effect on blade tip timing system output and is crucial for system calibration. Therefore, evaluating and minimizing uncertainty associated with later stages, which carry higher uncertainty, is imperative.

2. Blade Tip Timing

A typical blade tip timing system consists of several key components:

Sensors: These are typically non-contact optical sensors such as laser or capacitive sensors that are mounted around the rotating blades to detect the time of passage.

Signal conditioning unit: This component processes the signals from the sensors to extract relevant information such as blade tip timing data.

Data acquisition system: collects and stores the blade tip timing data for future analysis.

Post-processing tools: those tools are often **analysis software's, which** are used to analyze the collected data and extract useful insights into the dynamic behavior of the blades and **monitoring and control interface**: This component provides real-time feedback on the performance of the blades and may trigger alarms or shutdown procedures in case of abnormal behavior.

3. Probe Positioning Optimization Method

The following explanation on the optimization technique builds upon our previous exploration provided in [4]. We assume that the vibration y_i has the form:

$$y_i = d + \sum_{k=1}^{m} a_k sin(EO_K(\theta_i + \\ +2\pi n)) + \sum_{k=1}^{m} b_k cos(EO_K(\theta_i + 2\pi n)), \quad (1)$$

where i is the probe number and k is the mode number, m is the highest mode number, EO and a_k is the Engineer order and the amplitude relevant to the mode k, θ is the angular position of the sensor i and n is the number of rotation. The assessment of optimal probe positioning can be facilitated through the consideration of error size and the Root Mean Square Error (RMSE). Specifically, the error size (ξ_z), denoted as the Euclidean norm of the disparity between the estimated value z and the true value z_{true}, is mathematically represented as follows:

$$\xi_z = \|Z - Z_{true}\|_2 \quad (2)$$

Additionally, the expression for $RMSE_z$ is:

$$RMSE_k = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(Z_i - z_{true})^2}, \quad (3)$$

in this scenario, z_i represents the i^{th} estimated coefficient vector, while z_{true} stands for the true coefficient vector, with M representing the total count of random realizations. It's evident that for a singular realization, $RMSE_z$ and $\boldsymbol{\xi}_z$ are identical, thus interdependent. When dealing with multi-mode blade vibration, instead of employing $\boldsymbol{\xi}_z$ or $RMSE_z$, we might prioritize evaluating the Root Mean Square Error (RMSE) of the vibration amplitude for the specific mode of vibration of interest. Let's denote A_k as:

$$A_k = \sqrt{a_k^2 + b_k^2}, \quad (4)$$

the amplitude of the mode k obtained through the estimator or the mathematical model $\hat{A}_k$:

$$\hat{A}_k = \sqrt{\hat{a}_k^2 + \hat{b}_k^2} \quad (5)$$

The $RMSE$ of the mode k is defined as follows:

$$RMSE_k = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(A_{ki} - \hat{A}_k\right)^2}, \quad (6)$$

where A_{ki} represents the i_{th} estimated amplitude of vibration of mode k, and M denotes the total number of estimations. In a single-mode model without a constant offset, it's evident that the Mean Square Error equates to the mean of the error size ξ_z. Therefore, instead of computing the RMSE in this specific scenario, we propose a comprehensive algorithm that facilitates the selection of optimal probe positioning by minimizing the coherence μ of the design matrix. To provide a basis for comparison with existing methods, the algorithm integrates the minimization of error size ξ_z, the minimization of different RMSEs depending on vibration modes, and the minimization of the condition number Cond. Hence, the general steps for determining the best probe positioning are as follows:

- Determine the desired number of probes denoted L;
- Generate subsets, denoted as l, from the pool of L probes, and construct the sensitivity matrix for each configuration;
- Utilize the combined Auto-Regression with Least Squares Method to ascertain the vibration parameters for each set of probe positions;
- For each configuration: Calculate the condition number and coherence of the design matrix and evaluate the error size and the RMSEz/RMSEk of the estimate corresponding to the mode of vibration k, or the average if multiple modes are of interest;
- Choose the probe configuration that yields the most favorable results based on the computed metrics.

4. Study Case

Applying the previous algorithm considering that the blades have simultaneous synchronous and asynchronous vibration. Our Bladed disk is an aero-compressor bladed disk of 29 blades rotating at 83 Hz.

The Blades found to be vibrating at the following frequencies w_i = {332, 482.2, 599.3} (Hz), with the following corresponding amplitude A_j = *{0.3, 0.2, 0.1} (mm)* respectively. We assume that the selection of probes will be based on a previous configuration that consists of L = 18 equally distributed probes. The selection of 3 probes from this configuration enables 1140 possible sets of probes S_i. Table 1 summarize the three best selected based on the minimum occurred error size, condition number and coherence and the RMSE.

Table 1. Different probe configurations selected after the minimization of the error size ξ_z, the condition number and the coherence (Φ) with normal distributed noise.

S_i	ξ	Cond	μ	RMSE
S_1: (4,9,17)	1.344	1.764	0.919	1.04
S_2: (0,13,17)	1.716	1.33	0.796	1.295
S_3: (0,11,18)	1.679	1.759	0.665	1.264
S_4: (4,9,18)	1.345	1.799	0.867	1.01

Fig. 1 illustrates coherence, error size, and condition number relationships. Most probe positions have a condition number below 10, implying limited alterations in the system's solution with input data errors. Notably, no correlation exists between condition number and error size. Thus, a well-conditioned system may still have a large error size, resulting in lower accuracy in coefficient vector prediction.

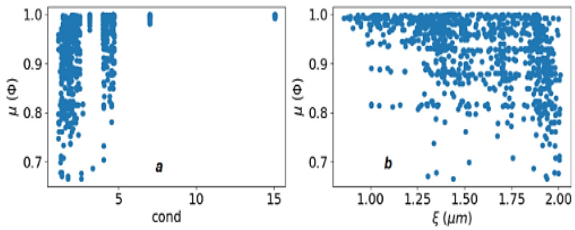


Fig. 1. Coherence evolution across condition number and Error size.

5. Conclusion

This work aimed to investigate the optimal probe position in Blade Tip Timing system. This approach was applied in the case of combined synchronous and asynchronous vibrations in order to find the right configuration based on the error size. Yet considering the selected case of study, we did not find any correlation between the error size and the coherence.

Acknowledgements

Project No. 903141 Experimental and theoretical research of uncertainties of the Blade Tip Timing Method, Institute of Thermomechanics Academy of Sciences of the Czech Republic; Research program Strategy AV21/27 Sustainable energetics, Academy of Sciences of the Czech Republic; Batista (Blade Tip Timing System Validator) – Clean Sky 2-862034, funded by the EU Commission (H2020 CS2).

References

[1]. P. Procházka, F. Vaněk, New methods of noncontact sensing of blade vibrations and deflections in turbomachinery, *IEEE Trans. Instrum. Meas*., Vol. 63, Issue 6, June 2014, pp. 1583-1592.

[2]. E. B. Tchawou Tchuisseu, P. Procházka, M. L. Mekhalfia, et al., New numerical and statistical determination of probes' arrangement in turbo-machinery, *J. Vib. Eng. Technol*., Vol. 11, 2022, pp. 2025-2035.

[3]. P. Russhard, Blade tip timing (BTT) uncertainties, *AIP Conf. Proc.*, Vol. 1740, June 2016, 020003.

[4]. E. B. Tchawou Tchuisseu, P. Procházka, D. Maturkanič, P. Russhard, M. Brabec, Optimizing probes positioning in blade tip timing systems, *Mechanical Systems and Signal Processing*, Vol. 166, 2022, 108441.

(020)

Automated Segmentation of the Left Ventricle in Cardiac CT Angiography Using a 2.5 UNet

Francesca Lo Iacono [1], Juan F.Calderon [1], Gianluca Pontone [2,3] and Valentina D. A. Corino [1,2]
[1] Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy
[2] Department of perioperative Cardiology and Cardiovascular Imaging,
Centro Cardiologico Monzino IRCCS, Milan, Italy
[3] Department of Biomedical, Surgical and Dental Sciences, University of Milan, Milan, Italy
E-mail: francesca.loiacono@polimi.it

Summary: Left ventricle (LV) segmentation in cardiac computed tomography angiography (CCTA) is an important and challenging task for the evaluation of cardiovascular diseases. In this framework, a deep learning approach for LV segmentation can overcome the issues related to its manual delineation, which is time consuming and prone to error. In the current study, we present an automatic method for segmentation of the LV in CCTA scans using the UNet 2.5D. The study includes 85 patients scans (for a total of 6171 images), whose LV segmentation was manually performed by expert radiologists and used as ground truth. The developed model provided a mean Dice score of 89 % on the test set, overcoming the performance obtained with the original and LSTM UNet models. The results achieved showed the potential of UNet 2.5D to provide accurate LV segmentations.

Keywords: Left ventricle, Segmentation, Deep learning, UNet 2.5D, Cardiac computed tomography.

1. Introduction

Cardiac substructure segmentation is a crucial step for cardiovascular disease diagnosis and treatment. Cardiac computed tomography angiography (CCTA) is a non-invasive imaging technique that is performed routinely for disease diagnosis and treatment planning. CCTA is often preferred by clinicians as it provides detailed anatomical information with high signal-to-noise ratio and good spatial resolution [1]. The left ventricle (LV) holds a key role in the study of cardiac function and disease diagnosis. For this reason, delineating LV boundaries represents an important step to investigate significant heart parameters such as the ejection fraction, stroke volume, LV mass, end-systolic volume, and end-diastolic volume [2].

Segmenting the LV represents a challenging task due to its large variations in shape, size, as well as contrast [3]. The manual delineation is time consuming, prone to inter and intra-observer variability and requires the availability of staff and additional resources. For these reasons, a fast and fully automated segmentation algorithm is necessary to improve diagnostic efficiency for the early detection and analysis of cardiovascular risk biomarkers.

Previous studies dealing with LV segmentation exist. In [3] a two-step LV segmentation, based on level-sets deformable contours, was performed in multi-slice CT images. LV internal wall was segmented, and then the external wall was obtained according to the shape of the internal wall using a coarse-to-fine strategy which first detected the LV and then refined the myocardial surface with contour evolution techniques and shape constraint [4].

Recently, convolutional neural networks (CNNs) segmentation-based methods, especially U-net-like models [5], have been widely investigated for cardiac segmentation. In [6] a combination of 3 CNNs was used to independently localize the LV on the axial, coronal, and sagittal plane, creating a bounding box around it. Thereafter, a dedicated CNN was built to identify voxels belonging to the LV. A UNet-based method was also employed in [7], with a 3D deep attention U-Net (DAU-Net), combining an attention U-Net [8] and a deep supervision. In [9] Li et al. proposed an 8-layer residual U-Net with deep supervision whose results were compared with the segmentation output obtained from the original U-Net and FC-DenseNet56 [10].

In the current study we propose a UNet architecture variant called UNet 2.5D to automatically segment the LV. Finally, the performance of this model will be compared with the results obtained from the original and the LSTM-based UNet models.

2. Materials and Methods

2.1. Dataset

The analyzed dataset consists of CCTA scans collected from 85 patients suffering from three different pathologies: amyloidosis (AM), aortic stenosis (AS) and hypertrophic cardiomyopathy (HCM). Each patient scan comprises a mean of 73±17 slices (range 30-115), where the LV was visible. The LV was manually delineated by expert radiologists and used as ground truth. The dataset was splitted into train, validation and test sets containing 75 %, 15 %, and 10 % of the patients, respectively, keeping the original disease stratification.

2.2. CCTA Scan Protocol

CCTA images used in this study were acquired using 256-slices (Revolution CT; GE Healthcare, Milwaukee, WI) or 320-slices wide volume coverage CT scanner (Aquilion ONE VisionTM; Canon Medical Systems Corp., Tokyo, Japan). No premedication with beta-blockers or nitrates was added before CT acquisition. Revolution CT scans were acquired with the following parameters: peak tube voltage of 100 kV, detector collimation of 160 mm using 256 rows by 0.625 mm on Z-axis. Patients received a fixed dose of 50 ml bolus of contrast medium (400 mg of iodine per milliliter, Iomeprol; Bracco, Milan, Italy). Aquilion ONE Vision scans were acquired using a peak tube voltage of 120 kV, detector collimation, 320 detector rows, 1.2-mm section thickness. The study, approved by the Institutional Ethical Committee, conforms to the Declaration of Helsinki, and all participants gave informed consent.

2.3. Image Preprocessing and Augmentation

Preprocessing steps were performed before model training to ensure consistent input dimensions and intensity values.

Since CT images were acquired using different CT scanner, a voxel intensity normalization was performed converting raw CT voxel intensity values to Hounfield Unit (HU) according to the rescale slope and rescale intercept parameters [11] found in the DICOM header. Therefore, windowing was performed: this is a preprocessing technique employed to fine-tune the contrast and brightness of medical images. It involves mapping the original pixel values to a new selected range capturing the characteristic HU values of the tissues of interest. Intensities exceeding this predefined range are displayed as black or white, while intensities within the designated range are mapped to the grayscale spectrum. Performing windowing, window level and window width need to be set: window level represents the center of the new range, while the window width represents the width of the range. The window level and width values commonly used for CT scans of the heart and ventricles, are 40 and 400, respectively, and they provide good contrast between the blood pool, myocardium, and other tissues. Ultimately, pixel intensities were normalized between 0 and 1.

Once preprocessed, data augmentation was performed by applying various transformations, such as rotation, scaling and flipping to the original images. Using data augmentation, the size of the training dataset was artificially increased, improving the generalization capability of the deep learning (DL) model.

2.4. Segmentation Models

Three DL-based segmentation models were implemented and evaluated in this study:

Original UNet: the UNet architecture, proposed by Ronneberger, et al. [5] and commonly used for image segmentation tasks, was slightly modified in this study to improve training stability, and prevent overfitting (see Fig. 1). In particular, a dropout, which consists of randomly dropping out some of the neurons during training, was included in the contracting path, after each convolutional layer, with a probability of 0.5. In addition, L2 regularization was applied to the convolutional layers, with a weight decay of 0.0001, together with a batch normalization which reduced the dependence on the initialization of the weights [12].

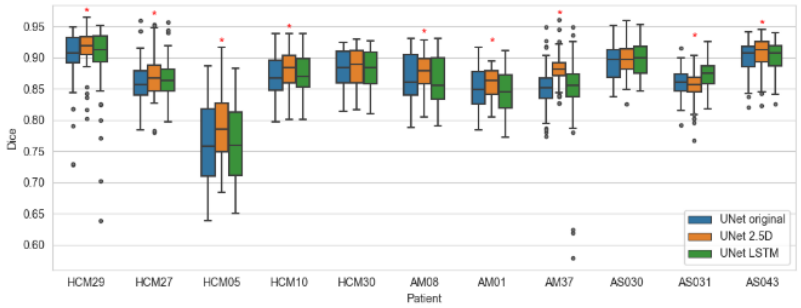


Fig. 1. Dice coefficient of the three models in the validation set. The red asterisk (*) indicates statistical significance (<0.05) comparing the UNet 2.5D, against the other two models, with a Wilcoxon signed-rank test.

Unet 2.5D: a variant of the UNet architecture designed to exploit the temporal dependency between consecutive CT slices. The UNet 2.5D model architecture consists of an encoder and a decoder pathway, similar to the original UNet architecture of Fig. 1. The encoder pathway consists of four down-blocks, where each down-block consists of two convolutional layers, followed by a ReLU activation function and batch normalization. Each down-block also includes a max-pooling layer that down samples the feature maps. The decoder pathway consists of three up-blocks, where each up-block consists of a transposed convolutional layer, followed by two convolutional layers, ReLU activation, and batch normalization. The up blocks also include skip connections that concatenate the corresponding feature

maps from the encoder pathway. Compared to original UNet, the main difference relies on the model input, which consists of 3-channel, instead of a single one, each of them representing a consecutive CT slice. Using this input, the obtained output is a 2D segmentation mask of the middle slice, predicted by considering the spatial information contained in the previous and next slices.

UNet LSTM: UNet architecture integrated with a Long Short-Term Memory (LSTM) level of 256 filters [13], placed after the final convolutional layer in the contraction path. This architecture is useful for modeling temporal dependencies in the input data, such as CT scan.

2.5. Model Evaluation

The performance of the segmentation models was assessed using the Intersection over Union (IoU) and Dice Coefficient, which measure the model ability to accurately segment the LV and are defined as

$$IoU = \frac{|A \cap B|}{|A \cup B|}, \quad (1)$$

$$Dice = \frac{2 \times (A \cap B)}{(A + B)}, \quad (2)$$

where A is the ground truth segmentation and B is the predicted segmentation. The IoU represents the area of overlap between A and B over the union area. The Dice is calculated dividing two times the intersection between A and B by the sum of the two segmentation regions. The models were trained using different loss functions, such as Binary Cross Entropy and Soft Dice Loss, to investigate their impact on the segmentation performance.

3. Results

3.1. Validation Set

Figs. 1 and 2 report the performance, in terms of Dice and IoU, reached with the 3 models considering all the slices for each patient in the validation set. It can be observed that UNet 2.5D achieved the highest Dice and IoU values for all patients. The average Dice was 0.87±0.03 for UNet 2.5D, followed by the LSTM and original UNet (both 0.86±0.04). The average IoU was 0.78±0.05 for UNet 2.5D, followed by the LSTM (0.77±0.06) and original UNet (0.76±0.06).

3.2. Test Set

The model showing the best performance on the validation set, i.e., the UNet 2.5D, was employed to segment the LV in the test set. The segmentation results for these patients are summarized in Table 1. The model achieved high performance in terms of mean Dice coefficient, ranging from 88 % to 91 %, and mean IoU, ranging from 78 % to 84 %.

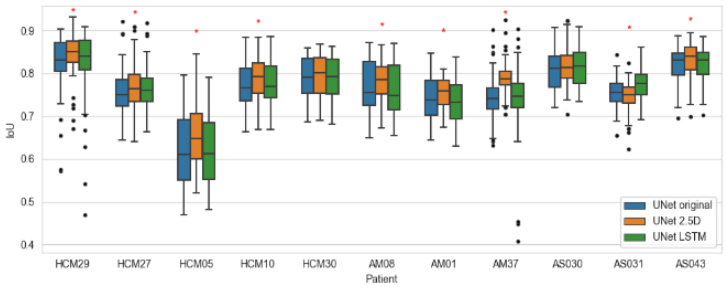


Fig. 2. IoU coefficient of the three models in the validation set. The red asterisk (*) indicates statistical significance (<0.05) comparing the UNet 2.5D, against the other two models, with a Wilcoxon signed-rank test.

Table 1. Metric results for UNet 2.5D on the test set.

Patient	N images	Dice	IoU
HCM13	75	0.869±0.111	0.776±0.149
HCM01	64	0.907±0.121	0.831±0.035
AM29	115	0.900±0.031	0.819±0.050
AS054	54	0.890±0.020	0.802±0.032
AS002	71	0.912±0.012	0.839±0.021
AS021	40	0.885±0.017	0.794±0.027
AS024	75	0.877±0.031	0.782±0.048
Mean	71 ± 22	0.891±0.035	0.806±0.051

Overall, the results demonstrated the effectiveness of the UNet 2.5D model in accurately identifying and segmenting LV of cardiac CT images. Fig. 3 reports an example of the ground truth and predicted masks, showing the promising results provided by the developed model.

4. Discussion

Image segmentation plays a crucial role in the field of medical imaging. In cardiological clinical applications, accurate segmentation of LV is a key step to diagnose and monitor several cardiovascular diseases. CCTA has become an important clinical diagnostic tool due to its non-invasive, short exam time, and low cost.

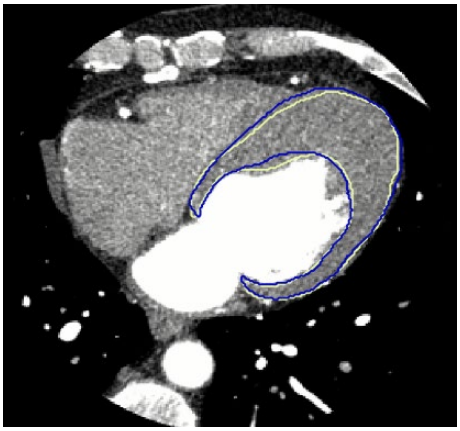


Fig. 3. Example of real (blue line) and predicted (yellow line) mask obtained for patient HCM13 (Dice = 0.869, IoU = 0.776).

In the current study, we proposed and compared an automated DL approach, using UNet 2.5D, to segment the LV in CCTA scans. The model was evaluated and compared with the original and LSTM UNet. The UNet 2.5D model exhibited the best performance on the validation set, both in terms of mean Dice coefficient and mean IoU, overcoming the results achieved with the original UNet. Therefore, the UNet 2.5D model was chosen to be applied on the test set images on which a mean Dice ranging between 0.87 to 0.91 was obtained. It was observed that with respect to the original, and LSTM, UNet models, the UNet 2.5D generated masks with precise boundaries and accurately captured the anatomical characteristics of the LV.

Actually, most of the DL approaches reported in literature for LV segmentation from CCTA, proposed UNet-based methods. Zreik et al. [6], was the first to employ CNNs, combining four of them to first localize the LV and then identify voxels belonging to it: the study reached a Dice coefficient of 0.85 on five scans. In [14] the study tested a fully convolutional network based approach achieving a Dice of 0.88 on 30 scans. Progresses were obtained by Guo et al. [7], employing a DAU-Net with a mean Dice coefficient of 0.916 on 20 scans. Furthermore, in [9] Li et al. computed the ground truth annotations using an interactive semi-supervised algorithm and developed an 8-layer residual U-Net with deep supervision which achieved a mean Dice of 0.92 on 20 patients.

To the best of our knowledge the UNet 2.5D has never been tested for the LV segmentation task. The primary benefit of this model lies in its ability to provide better performance compared to the traditional 2D approach with less computational resources than a 3D approach. The results achieved are promising and comparable to other studies. Some limitations exist: the model was not tested on publicly available datasets, making it difficult to draw comparisons with respect to other developed models. In addition, testing our model on different datasets, acquired with different scans, might confirm the robustness and generalization ability of this model. Finally, the manual annotations on the dataset can vary according to the criteria employed by the radiologist performing this task: this inconsistency may affect the quality of the ground truth labels, and, in turn, the performance of the model.

In conclusion, this study has demonstrated the potential of DL models for accurate medical images segmentation and provided important insight into future directions for research in this field.

References

[1]. J. Larrey-Ruiz, et al., Automatic image-based segmentation of the heart from CT scans, *EURASIP Journal on Image and Video Processing*, Vol. 2014, Issue 1, 2014, 52.

[2]. M. A. Shoaib, et al., An overview of deep learning methods for left ventricle segmentation, *Computational Intelligence and Neuroscience*, Vol. 2023, 2023, e4208231.

[3]. R. Medina, et al., Left ventricle myocardium segmentation in multi-slice computerized tomography, in *Proceedings of the Technical and Scientific Conference of the Andean Council (ANDESCON'16)*, 2016, pp. 1-4.

[4]. L. Zhu et al., A complete system for automatic extraction of left ventricular myocardium from CT images using shape segmentation and contour evolution, *IEEE Transactions on Image Processing*, Vol. 23, Issue 3, 2014, pp. 1340-1351.

[5]. O. Ronneberger, et al., U-Net: Convolutional networks for biomedical image segmentation, in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'15)*, 2015, pp. 234-241.

[6]. M. Zreik, et al., Automatic segmentation of the left ventricle in cardiac CT angiography using convolutional neural network, in *Proceedings of the IEEE 13th International Symposium on Biomedical Imaging (ISBI'16),* Prague, Czech Republic, 2016, pp. 40-43.

[7]. B. Jun Guo, et al., Automated left ventricular myocardium segmentation using 3D deeply supervised attention U-net for coronary computed tomography angiography; CT myocardium segmentation, in *Medical Physics*, Vol. 47, Issue 4, 2020, pp. 1775-1785.

[8]. O. Oktay, et al., Attention U-Net: learning where to look for the pancreas, *arXiv Preprint*, 2018, arXiv:1804.03999.

[9]. C. Li, et al., An 8-layer residual U-Net with deep supervision for segmentation of the left ventricle in cardiac CT angiography, *Computer Methods and Programs in Biomedicine*, Vol. 200, 2021, 105876.

[10]. S. Jegou, et al., The one hundred layers tiramisu: fully convolutional dense nets for semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'17)*, Honolulu (HI), USA, Jul. 2017, pp. 1175-1183.

[11]. U. Schneider, et al., The calibration of CT Hounsfield units for radiotherapy treatment planning, *Physics in Medicine & Biology*, Vol. 4, 1996, 111.

[12]. S. Ioffe, et al., Batch normalization: accelerating deep network training by reducing internal covariate shift, in *Proceedings of the 32nd International Conference on*

International Conference on Machine Learning (ICML'15), Lille, France, 6-11 July 2015, pp. 448-456.

[13]. R. P. K. Poudel, et al., Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation, in Reconstruction, Segmentation, and Analysis of Medical Images (M. A. Zuluaga, K. Bhatia, B. Kainz, M. H. Moghari, D. F. Pace, Eds.), *Springer International Publishing*, 2017, pp. 83-94.

[14]. H. J. Koo, et al., Automated segmentation of left ventricular myocardium on cardiac computed tomography using deep learning, *Korean J. Radiol.*, Vol. 21, Issue 6, 2020, pp. 660-669.

(022)

CARES-UNet: Contour-guided Attention-based RES-UNet for Optic Disc and Optic Cup Segmentation

Tewodros Gizaw, Zhighuan Qin and Habte Lejebo
School of Information and Software Engineering, University of Electronic Science
and Technology of China, Chengdu, 610054, China
E-mail: decmen2008@gmail.com

Summary: For the accurate diagnosis of retinal disorders, especially glaucoma, precise segmentation of the optic disc (OD) and optic cup (OC) is essential. Using state-of-the-art deep learning models for segmentation, our methodology incorporates techniques, namely contour data, to improve the overall performance of the model. Fundus images require extra pseudo-label representation information because of their high contrast, fuzzy edges, and intensity changes. In supervised techniques, the lack of such representation makes segmentation difficult, especially when dealing with fundus pictures that are strongly contrasted and have fuzzy borders. Our suggestion is to use a Contour-Aware Attention-based RESU-Net to overcome these drawbacks. The model improves edge information by utilizing novel contour techniques to handle different pixel intensities. Convolutional Block Attention Module (CBAM) integration sharpens emphasis on key characteristics, making it easier to recognize fuzzy borders and manage intensity and low contrast. Evaluation on REFUGE and DRISHTI-GS datasets demonstrates superior optic disc and optic cup segmentation performance, showcasing the model's potential across diverse clinical scenarios.

Keywords: Contour, Convolutional Base Attention Module (CBAM), Fundus, Glaucoma, Optic cup, Optic disc, Segmentation.

1. Introduction

Glaucoma stands as the third most prevalent cause of irreversible vision loss, following uncorrected refractive errors and cataracts. According to research published in the literature [1], glaucoma is expected to affect 76.0 million people worldwide by 2020 and is expected to rise to 111.8 million by 2040. The fundus images, and other modalities can be used in conjunction with a variety of imaging techniques, including magnetic resonance imaging (MRI) and optical coherence tomography (OCT), as demonstrated by [8], to diagnose glaucoma. The recommended method, however, places a strong emphasis on using fundus pictures to diagnose glaucoma. These images, which were taken using a fundus camera [8], offer a thorough perspective of the eye and serve as the basis for the suggested methodology for diagnosing glaucoma.

Computer aided segmentation techniques have been shown to improve diagnosis speed and accuracy Various traditional segmentation methods, such as region-based and model-based approaches, have been developed for the automated segmentation of the optic disc and optic cup in fundus images indicated [6], OD and OC segmentation are carried out using heuristic-based approaches utilizing.

Manually developed features such as color, gradient, and texture data in the traditional manner which is a heuristic approach. However, because they are the product of artificial feature creation, these attributes range from highly venerated to misguided [8]. Consequently, their stability and representational abilities will determine how well segmentation performs. Deep learning-based techniques particularly U-Net [3], and fully convolutional network (FCN) [8], have been effectively used in this sector in recent years and perform exceptionally well when compared to traditional methods.

In particular, U-Net and other encoder-decoder based models have been widely used for medical image segmentation. The U-Net family of variants, which includes Efficient-UNet, Dense-UNet, Attention-UNet, and RES-UNet, has been extensively used in numerous studies [8].

Due to the complexity of fundus images and the intensity similarity between the surroundings and the optic disc and optic cup regions segmentation becomes challenging making it difficult to achieve accurate results [9].

Unless we specifically focus on the required regions of interest deep learning models employ various strategies such as attention mechanism incorporation, ROI extraction, and localization of the optic disc. Several approaches [10], have shown encouraging results in the segmentation of the optic disc and optic cup from fundus images.

Particularly, for addressing the high redundancy in fundus images, the incorporation of attention mechanisms [11-13], has demonstrated potential in enhancing the performance of CNN-based glaucoma detection [11-13].

Some of the approach follow region of interest extraction (ROI) by cropping the input images and focus on the required regions. Even if this approach is successful has its own limitation due to ROIs are often defined based on specific characteristics or features in an image. If these features are not robust or stable under different conditions, the ROI selection may become unreliable [14]. The other approach they

follow is localization of optic disc before segmentation [2, 3], this approach has limitation regarding retinal images can vary in quality, resolution, and clarity.

Images with low quality or artifacts may pose challenges in accurately localizing the optic disc. In such types of approach, we will face the model generalizability issues for different datasets and the segmentation result for the optic cup is highly rely on the accurate segmentation of optic disc. Unlike the previous approach, we adopt a strategy that involves using a traditional edge-based approach as an additional source of information, without performing preprocessing segmentation tasks to get ROI extraction or OD localization. This approach allows us to concentrate on the necessary regions by implementing powerful image processing algorithms specifically designed for tasks like edge and region extraction.

We propose a hybrid model that combines the edge-based approach with a deep learning model. In the edge-based approach, we have implemented a contour mechanism to guide region extraction which has had a positive impact on our feature representation [9].

Most of the listed approach are follow either edge-based approach or deep learning approach. We try to hybrid the edge-based model with the deep leaning model. Because, edge-based model gives us clear and strong cues for guidance of the required regions they are very powerful in medical image processing [9]. We use a technique of contour for region guidance to feature representation to the best of our understanding, no one has previously used contour aware attention-guided Res-UNet to detect glaucoma in fundus images to segment them accurately and merging both active contour and attention information for glaucoma disorders and artifacts. Contours provide a concise representation of object boundaries, which can aid in image understanding and annotation.

By incorporating contour information, algorithms can infer spatial relationships between objects, localize objects of interest, or generate annotations that facilitate human understanding or automated processing. Incorporating contour information allows for a more comprehensive analysis of image data and improves the accuracy, robustness, and efficiency of various computer vision and image processing tasks [9]. Our work makes the following contributions:

1. We integrate intensity in homogeneity by employing contour methods to address variations in pixel intensities across different regions of the image;
2. We proposed a guided contour module to enhance edge preservation, where the contour information contributes supplementary edge details to the feature map;
3. We propose a hybrid architecture, merging Contour and Attention Gates with a Res-UNet backbone, to prioritize the activation of relevant regions in the feature maps;
4. We thoroughly test our methods on the REFUGE and DRISHTI-GS datasets, outperforming both baseline models and currently available cutting-edge segmentation methods.

2. Proposed Method

Various methods described in the introduction parts has their own limitation while performing the task of optic disc and optic cup segmentation. To alleviate those mentioned problems, we proposed a novel approach contour-aware attention-based RESU-Net. We modify the architecture presented by [15], which takes the advantages of both U-Net and RES-Net model. Before discussing the general architecture first, we present the proposed encoder model compared with the previous approach described in Fig. 1.

$$xl + 1 = f(x1) + xl \quad (1)$$

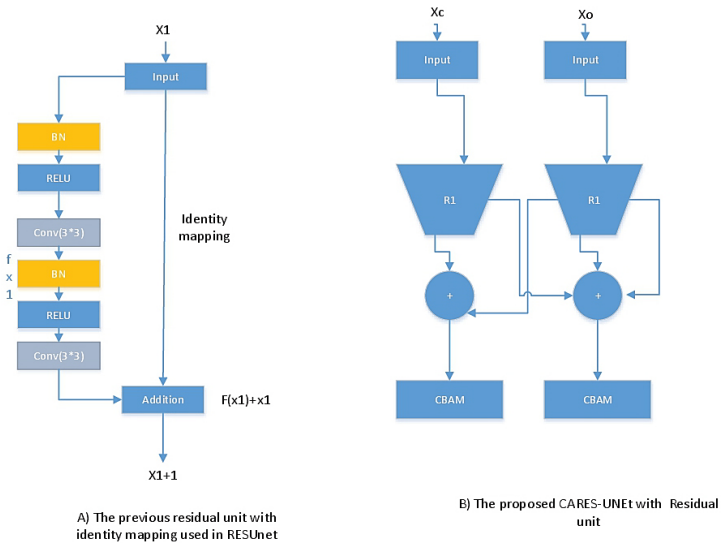


Fig. 1. Comparison Previous residual unit and the proposed CARES-UNet.

In case A) as shown in Fig. 1 the output is xl+1 the previous residual unit with identity mapping. The input given are xl which is the original images and the modified residual block as a function f(x+1).

In case B) as shown in Fig. 1 there is incorporation of contour as (X_c) in addition to the original images contours are given to the residual block before performing the identity mapping.

$$Xc(out) + 1 = f(x1) + Xc + CBAM + \\ + Xo(out) + 1 \quad (2)$$

The contour images output before giving to the next residual block is inputs are the residual blocks as function f(xl+1), the contour images (Xc), the output of original images as described in equation four Xo(out)+1 and CBAM.

As shown in Fig. 2 the proposed model contour-guided attention based RESU-Net model before giving the inputs to the next residual blocks it accepts both original images and contour images then fused with CBAM.

$$xo(out) + 1 = f(xl + 1) + Xo + \\ + Xc(out) + 1 + CBAM \quad (3)$$

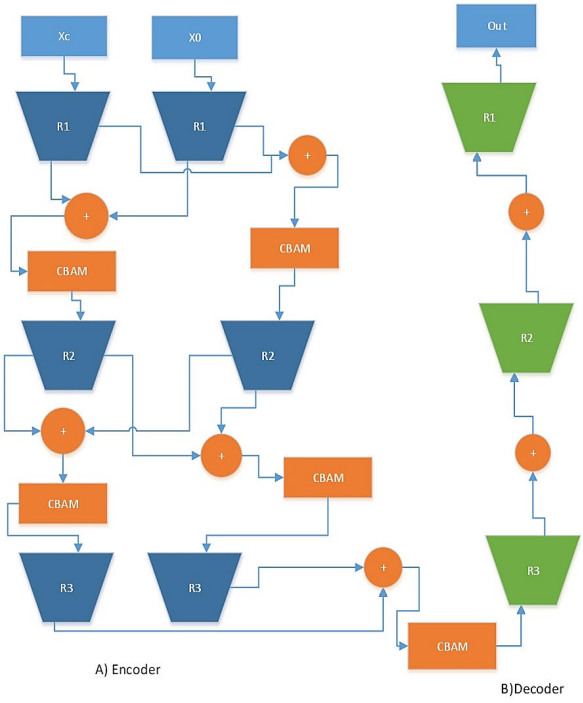


Fig. 2. The Proposed Model.

The original images output before giving to the next residual block is inputs are the residual blocks as function f(xl+1), the original images (Xo), the output of contour as described in equation (3) xc(out), and CBAM. We proceed these steps in each of the encoder blocks to strength our model for extracting better features.

Attention Incorporation

Following the concatenation of original features and contour image features, we applied CBAM [16]. Across these features $zcat$. In addition to the aforementioned components, our study also focuses on incorporating attention mechanisms into the architecture design. Extensive research has been conducted on the significance of attention [16], which not only helps determine areas of focus but also enhances the representation of relevant features. Our objective is to enhance the network's representational power by leveraging attention processes that prioritize vital structure while subduing unnecessary.

To achieve this, channel and spatial axes are two significant dimensions that this module enables us to emphasize. The CBAM module effectively recognizes essential features by applying convolution techniques, which combine cross-channel and spatial information. Each branch of the network may learn what and where to pay attention to in the channel and spatial axes by applying the channel and spatial attention modules ule substantially enhances the information flow inside the network by learning which information to give attention or suppress. The network's ability to acquire and represent significant features is ultimately improved by this improvement in information flow, which raises both the network's overall performance and accuracy. Feature map is given as input to the CBAM it follows a sequential process to refer both channel and spatial attention map.

$$f' = Mc(f')' \otimes' f, f = Ms(f')' \otimes' f' \quad (4)$$

Contour Extraction

Generation of contours Contour-based image segmentation methods work by utilizing edge or contour information to identify and delineate object boundaries [9]. Here, is a general overview of the process: Edge Detection: The first step is to detect edges in the image using edge detection algorithms such as Canny, Sobel, or LoG. These algorithms analyze pixel intensity gradients to identify significant changes in intensity that correspond to object boundaries.

Contour Extraction: Once the edges are detected, contour extraction algorithms are used to obtain continuous curves or boundaries that represent the object contours. This can be achieved through techniques like the Marching Squares algorithm or the Watershed algorithm.

Contour Refinement: Sometimes, the extracted contours might contain noise or inaccuracies. Contour refinement techniques are applied to enhance the extracted boundaries, smooth out irregularities, and increase the overall quality of the contours. This step helps to ensure more accurate segmentation.

Segmentation using Contours: The extracted and refined contours are then utilized as the basis for segmentation. Various segmentation algorithms can be employed, such as region-growing, region-based level set, or active contour models (snakes), to segment the image based on the identified contours.

To extract the contour, we follow the following steps first we used an openCV to read an input image then we resize the image accordingly. After resizing the image, we have converted the image into gray scale after that we add Gaussian Blur then we extract using canny (10,100) get structuring element. Finally, we add the dilation operation to the edged image then we find and draw the contours. Finally, refine contour for both the datasets REFUGE and DRISHTI-GS. For DRISHTI-GS for a total of 168 augmented images we extract contour information. Accordingly, for Refuge dataset we extract a total of 1200 contour images.

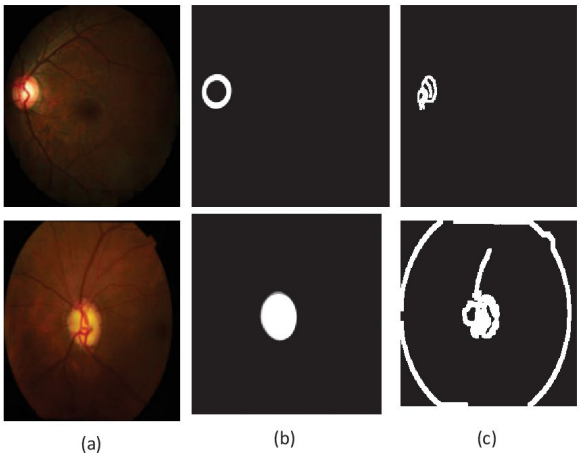


Fig. 3. Extracted Contour information from both datasets. (a) Indicates original images; (b) Indicates the ground truth, and (c) Indicates the extracted contour information.

3. Dataset and Experiment Setting

In this study, two openly accessible datasets REFUGE and DRISHTI-GS where utilized. REFUGE datasets are referenced in [17], a total of 1200 images with the ground truth information's all the images are captured using fundus cameras validated by professionals no biased information's mainly used for glaucoma identification purposed. We have been used 960 images for training and 240 images for validation and testing purpose. In addition, to the originals images the pseudo-label information's by extracting the contour information's is implemented.

The other dataset used in this study is DRISHTI-GS [18], these datasets consist of 50 images with ground truth information. We apply the data augmentation techniques of horizontal flip, vertical flip, and rotate to enlarge our dataset size making them a total of 268 images. Among them we have been used 200 images for training and 68 images are used for validation and testing purpose.

For the implementation purpose the PyTorch 1.13.1 DL library and python 3.8.16 performing language were utilized for the experiment. The machine used for the experiments was an included four NVIDIA GeForce RTX 3080Ti GPUs with 6 GB of RAM. With Cuda version 11.7 of 12th generation of intel(R).

The performance evaluation metrics used throughout the explement is Dice and IOU how we compute it described below as an equation.

$$Dice = 2TP/2TP + FP + FN, \quad (5)$$

$$IOU(a,b) = \frac{(a \cap b)}{a} + (b) - (a \cap b) \quad (6)$$

4. Result and Analysis

We perform extensive experiment and compare with the SOTA model like GDSeg-Net [4], Attention U-Net [2], Deep ResU-Net [15]. Systematically test and evaluate the mentioned model to obtain their predicted equivalent mask results. During the experimentation, the same data split strategy is applied to divide the data into training, validation, and test sets. All evaluated models utilize the same dataset, with consistent data sizes and an equal number of epochs.

The results presented in Fig. 4 showcase the optic disc (OD) segmentation performance using the DRISHTI-GS dataset. In this experiment, we with the same data split strategy, we conducted a comparative analysis of our proposed method against other prominent CNN-based approaches, including Deep RES U-Net [15], Attention U-Net [13], EARDS-Net [6], and GDCSeg-Net [4]. Table 1 showcases the comparative results on DRISHTI-GS datasets for optic disc segmentation. As depicted in Fig. 4, Segmentation results of OD by the proposed model are illustrated in Fig. 4. When utilizing the same input images and labeled images denoted as (a) and (b), respectively, the predicted masks.

Table 1. Performance Result of OD for DRISHTI-GS dataset.

No.	Model	Dice	IOU
1	U-Net [6]	0.9642	0.9319
2	Efficient-UNet [6]	0.9715	0.9447
3	EARDS-Net [6]	0.974	0.949
4	GDCSeg-Net [4]	0.974	0.95
5	Ours	0.982	0.961

Provided by different models are not identical. Some of the compared models yield entirely different predictions for a given input. However, when we examine the last two models, their results are slightly similar. However, our proposed model produces a superior optic disc (OD) prediction, especially for the image type with high contrast depicted in the Fig. 4, due to the incorporation of supportive information such as contour to learn better features. This result is noteworthy when compared with other state-of-the-art (SOTA) models. The comparison highlights that our proposed CARES U-Net demonstrates superior

Performance compared to the aforementioned U-Net architecture based model Specifically, as shown

in Table 1, in addition sample of the result of predicted OD segmentation mask as presented in Fig. 4.

Fig. 5 result show that the joint OD and OC segmentation result of SOTA model and the proposed model. label (a), shows the original image of REFUGE dataset, label (b) shows the ground truth information of the original images, and label(c) shows the predicted segmented result. As shown in the figures specially in highly contrast images the other models are not performing well with this regard our model attains good segmentation result. For the normal images input given the result are very competitive as shown in the Fig. 5. Our proposed model capable to learn region based information using pseudo-label information.

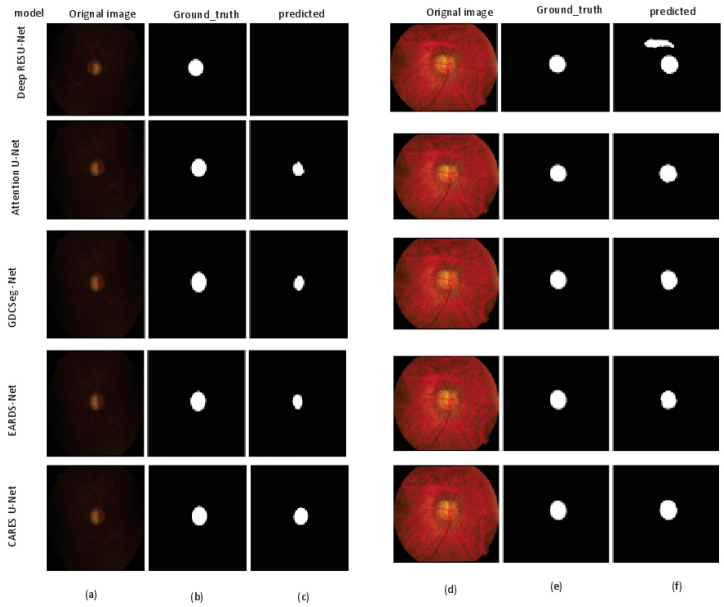


Fig. 4. Result of OD for sample image dataset in DRISHTI-GS.

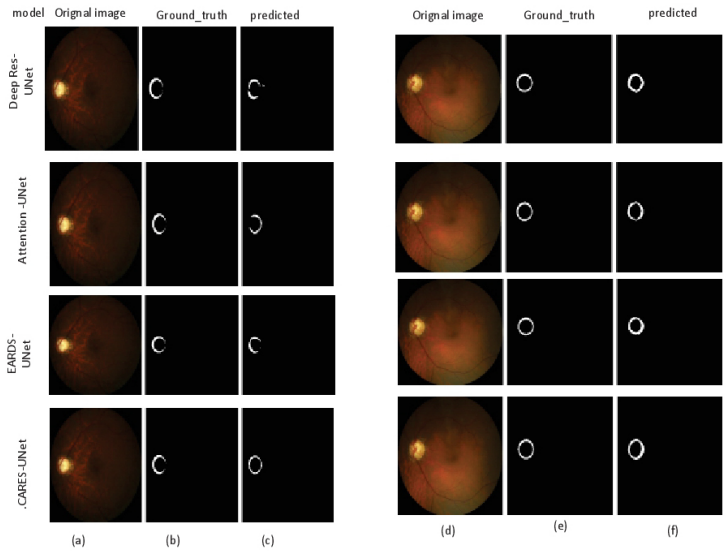


Fig. 5. Result of REFUGE for sample image dataset.

The table result shows the comparison among SOTA in DRISHTI-GS dataset our proposed model yields better result in optic disc segmentation. We also presented in Fig. 4 for more illustration of the segmentation result.

The incorporation of contour information used as a supportive information and each residual block fused with CBAM yields better feature extraction.

5. Discussion

The complexity of fundus images, particularly in high contrast scenarios and with fuzzy borders, poses challenges for accurate segmentation of optic disc and optic cup regions. Our proposed approach leverages pseudo-label information to enhance performance, providing valuable support for region-based and

edge-based information. The results, evaluated using IOU and Dice on both REFUGE and DRSHT-GS datasets, are presented in Table 1 and accompanying figures.

The incorporation of pseudo-label information has a notable impact on performance improvement. As indicated in Table 1, the proposed model achieved an increase of 0.008 % for Dice and 0.011 % for IOU.

As stated in Fig. 5 the proposed model attained accurate segmentation result in REFUGE datasets. The input image are joint optic disc and optic cup regions the outputs also show are joint segmentation result. For such types of images calculating the cup to disc ratio is difficult so, further strategies must be used to get the separate result for both required regions.

Additionally, integrating contour information and CBAM proved to be beneficial, contributing to significant performance improvements. However, these enhancements did not completely solve Challenges associated with fuzzy borders and high-contrast images, as observed in optic cup segmentation. The impact on optic cup segmentation did not result in improved outcomes.

Further, we are doing the ablation studies to clearly show an impact of the corporate modules in the base-line model performance.

In order to do that we have been checked the result of the base line model without adding both contour and CBAM. After doing so we proceed by adding only contour information to check the performance change on the base-line model. Finally, adding both contour and CBAM assessing the impact of the base-line model.

According to the overall abilities studies, the base line model performed better in the optic disc and optic cup segmentation tasks when contour and CBAM were integrated. However, further work needs to be done to raise the performance improvement level in optic cup segmentation. It is recommended to utilize a robust algorithm for extracting contour information.

6. Conclusions

This paper introduces CARES-UNet, a supervised framework for image representation learning in segmentation. Utilizing a contour-aware module, the proposed method obtains region-based representations of labeled fundus images. Additionally, the inclusion of the CBAM module enhances critical region identification for each encoder model, addressing challenges in highly contrasted and fuzzy-bordered images.

Experimental findings show that incorporating contour details significantly improves model performance, especially for images with high contrast and fuzzy borders. The suggested approach is tested on two publicly available fundus image datasets, REFUGE and DRISHTI-GS, showcasing its efficacy in mitigating representation biases and improving the supervised segmentation of the required regions. Future research should focus on investigating the stability of the proposed approach and improving segmentation accuracy by integrating deep learning techniques with clinical data.

Further exploration is recommended for refining optic cup segmentation, exploring alternative modalities, and generalizing across various glaucoma datasets like ACRIMA and RIM-ONE.

Acknowledgements

Under the project Privacy & Security key theory and Technology based on Data Cycle (Grant No. 61520106007), the National Natural Science Foundation of China (NSFC), University of Electronic Science and Technology of China, provided assistance for this work.

References

[1]. F. Guo, W. Li, J. Tang, B. Zou, Z. Fan, Automated glaucoma screening method based on image segmentation and feature extraction, *Medical Biol. Eng. Comput*., Vol. 58, 2020, pp. 2567-2586.

[2]. L. Li, M. Xu, H. Liu, Y. Li, X. Wang, L. Jiang, Z. Wang, X. Fan, N. Wang, A large-scale database and a CNN model for attention-based glaucoma detection, *IEEE Trans. Medical Imaging*, Vol. 39, Issue 2, 2020, pp. 413-424.

[3]. A. Almustofa, A. Handayani, T. Mengko, Optic disc and optic cup segmentation on retinal image based on multimap localization and U-Net convolutional neural network, *Journal of Image and Graphics*, Vol. 10, Issue 3, 2022, pp. 109-115.

[4]. Q. Zhu, X. Chen, Q. Meng, J. Song, G. Luo, M. Wang, F. Shi, Z. Chen, D. Xiang, L. Pan, et al., GDCSeg-Net: general optic disc and cup segmentation network for multi-device fundus 617 images, *Biomedical Optics Express*, Vol. 12, 2021, pp. 6529-6544.

[5]. S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, D. Terzopoulos, Image segmentation using deep learning: A survey, *IEEE Trans. Pattern Anal. Mach. Intell*., Vol. 44, Issue 7, 2022, pp. 3523-3542.

[6]. W. Zhou, J. Ji, Y. Jiang, J. Wang, Q. Qi, Y. Yi, EARDS: EfficientNet and attention-based residual depth-wise separable convolution for joint OD and OC segmentation, *Frontiers in Neuroscience*, Vol. 17, 2023, 1139181.

[7]. P. S. Mangipudi, H. M. Pandey, A. Choudhary, Improved optic disc and cup segmentation in glaucomatic images using deep learning architecture, *Multim. Tools Appl*., Vol. 80, Issue 20, 2021, pp. 30143-30163.

[8]. H. N. Veena, A. Muruganandham, T. S. Kumaran, A novel optic disc and optic cup segmentation technique to diagnose glaucoma using deep learning convolutional neural network over retinal fundus images, *J. King Saud Univ. Comput. Inf. Sci*., Vol. 34, Issue 8, 2022, pp. 6187-6198.

[9]. Y. Li, G. Cao, T. Wang, Q. Cui, B. Wang, A novel local region-based active contour model for image segmentation using Bayes theorem, *Information Sciences*, Volume 506, 2020, pp. 443-456.

[10]. B. Liu, D. Pan, Z. Shuai, H. Song, ECSD-Net: A joint optic disc and cup segmentation and glaucoma classification network based on unsupervised domain adaptation, *Comput. Methods Program Biomed*., Vol. 213, 2022, 106530.

[11]. R. Liu, T. Wang, X. Zhang, X. Zhou, DA-Res2UNet: Explainable blood vessel segmentation from fundus images, *Alexandria Engineering Journal*, Vol. 68, 2023, pp. 539-549.

[12]. H. Lyu, X. Li, J. Zhang, C. Zhou, X. Tang, F. Xu, Y. Yang, Q. Huang, W. Xiang, D. Li, Automated inter-patient arrhythmia classification with dual attention neural network, *Comput. Methods Programs Biomed*., Vol. 236, 2023, 107560.

[13]. T. Shyamalee, D. Meedeniya, Attention u-net for glaucoma identification using fundus image segmentation, in *Proceedings of the International Conference on Decision Aid Sciences and Applications (DASA'22)*, 2022, pp. 6-10.

[14]. A. Zhao, H. Su, C. She, X. Huang, H. Li, H. Qiu, Z. Jiang, G. Huang, Joint optic disc and cup segmentation based on elliptical-like morphological feature and spatial geometry constraint, *Comput. Biol. Medicine*, Vol. 158, 2023, 106796.

[15]. Z. Zhang, Q. Liu, Y. Wang, Road extraction by deep residual U-net, *IEEE Geosci. Remote. Sens. Lett*., Vol. 15, Issue 5, 2018, pp. 749-753.

[16]. S. Woo, J. Park, J. Lee, I. S. Kweon, CBAM: convolutional block attention module, in *Proceedings of the 15th European Conference Computer Vision (ECCV'18)*, Munich, Germany, September 8-14, 2018.

[17]. H. Fu, F. Li, J. I. Orlando, H. Bogunović, X. Sun, J. Liao, Y. Xu, S. Zhang, X. Zhang, REFUGE: Retinal Fundus Glaucoma Challenge, *IEEE Dataport*, 2019.

[18]. J. Sivaswamy, S. R. Krishnadas, G. D. Joshi, M. Jain, A. U. S. Tabish, DRISHTI-GS: Retinal image dataset for Optic Nerve Head (ONH) segmentation, in *Proceedings of the IEEE 11th International Symposium on Biomedical Imaging (ISBI'14)*, April 29 – May 2, 2014, Beijing, Chin, Beijing, China, pp. 53-56.

(026)

A Multi-class Classification for Reproduction of Non-articulatory English Alphabets with Minimal Phonetic Combination Dictionary

Aprameya V. Madhwaraj [1], Ashish A Iyer [1], Mahitha M [1], Palli Padmini [2] and Kaustav Bhowmick [1]
[1] PES University, Bangalore, India
[2] Genpact, Bangalore, India
E-mail: aprameya.madhwaraj@gmail.com

Summary: This paper presents a novel approach to reproduce unsorted English alphabets, emphasizing laryngeal and glottal involvement over traditional articulatory gestures. With 6-11 % of the global population facing speech disorders, impacting natural communication, the study explored LSTMs, Random Forests, and RNN models initially, yielding less than satisfactory results. Subsequently, an Artificial Neural Network (ANN) Multi-Class Classifier demonstrated remarkable accuracy, exceeding 88 %, correctly predicting alphabets in 9 out of 10 instances. The research includes a self-made concise phonetic dictionary with 42 samples, containing phonetic combinations of Non-Articulatory English alphabets, prepared using Fast Fourier Transform. Also has an interactive feature where the model predicts the alphabets given a few features and also provides voice outputs with Google's Text-to-Speech. Despite dataset constraints, the ANN proves efficient in predicting non-articulatory alphabet sounds, showing potential for practical applications, especially in aiding individuals with speech disabilities and addressing a significant global gap in supporting speech disorders.

Keywords: Artificial neural networks, Fourier transform, Google text-to-speech, Machine learning, Phonetics, Speech synthesis.

1. Introduction

The production of speech looks at the interaction of different vocal organs, for example the lips, tongue and teeth, to produce particular sounds. Speech production depends on the combination of phonemes. Unlike articulatory alphabets that involve the movement or contact of speech organs (articulators), non-articulatory alphabets primarily rely on the larynx and glottis, in some cases nasal as well in their production [1]. Phonemes are combined to form syllables, words, and ultimately, complete utterances. Phoneme combinations are the smallest units of sound in a language that can distinguish words from one another. A subset of the English alphabet, whose pronunciation is based on the combination of phonemes altogether. Articulatory Phonetics utilizes tools like real-time MRI (Magnetic Resonance Imaging) to observe vocal tract changes during speech, while more abstract methods such as Ultrasound Tongue Imaging and Palatography provide insights into tongue and palate movements for a comprehensive analysis of speech articulation [2]. The complexity of speech, influenced by factors such as accent, age, and health poses challenges for accuracy. Phonetic research plays a crucial role in refining speech technology, enabling machines to understand and produce nuanced variations in speech sounds [3]. English spellings often fail to consistently depict the diversity of pronunciation, resulting in variations even within the same word due to different dialects.

This highlights the need for a systematic and formal approach and this is where phonetics plays its role. The importance of phonetics, which when held in comparison with abstract sound units is crucial as it provides a more detailed and precise analysis of speech sounds, hence the English alphabet encompassing 26 letters, are typically studied in terms of phonetics for a reliable and undeterred understanding [4]. However, acoustic similarity is also one of the important variables to explain the confusions of speech sounds. Vowels are characterised by sustained voicing and lack of constriction whereas consonants are characterised by vocal tract constriction and aperiodicity [5]. Different acoustic-phonetic measurements such as voiced and unvoiced parameters, articulatory features, vowel offset and onset points, nasalization etc. are the necessities for phoneme recognition, which is the fundamental unit of speech with defined numbers in every language. Voiced speech arises from vocal tract vibration, while unvoiced speech stems from turbulent airflow due to vocal tract constriction [6]. Alphabets like /H/ and /O/, whose pronunciation are least dependent on mouth movements, are solely dependent on airflow through the larynx. Special cases, such as /M/ and /N/, which sound very similar if not pronounced well, are difficult to reproduce accurately without considering nasal involvements. In the case of dental fricatives like /F/, require a combined effort of teeth and lip for production. These cases underscore the significance of special movements and the substantial involvement of the larynx and glottis for accurate speech reproduction.

The emergence of Deep Neural Networks, notably Time-delay Neural Networks (TDNNs), marked a paradigm shift in speech recognition systems due to

their ability to classify speech patterns (such as phonemes) in a time-invariant manner, akin to human speech perception. Leveraging techniques like statistical methods, hybrid approaches, RNNs, K means, neural networks, and vector quantization, classification algorithms yielded significant performance gains with global optimization of ANNs and sequence learning [7]. Precise pronunciation of non-articulatory alphabets depends on the combination of known phonemes to produce the former. ANN model is used to train the dictionary dataset. Since there are not many unsorted alphabet samples, stratified cross validation is performed to further train the model.

The paper explores a unique perspective on English alphabets offering a classification that reveals how the production of precise pronunciation depends on the combination of phonemes. Hence, the present work claims the following achievements:

- A phonetic method for accessing an underrepresented category in English;
- A highly accurate and reliable ML algorithm through Neural Networks, which to the best of our knowledge, has not been used in speech production techniques;
- A light and minimal dictionary of Phonetic Combinations;
- Predictive model with an Interactive User Experience.

The paper concludes with a comprehensive implementation, demonstrating the accurate reproduction of speech sounds, marking a significant stride toward resolving the targeted problem. The overall workflow with the technical details is explained in depth in the methodology (Section 2). Further, this system is tested and measured with performance metrics with visuals and snippets for every step (Section 3). The impact of this solution with its applications are discussed in Section 4.

2. Methodology

This work revolves around identification and careful classification of Non-Articulatory English Alphabets. Following this, led to the development of a concise Phonetic dictionary. Keeping this as a dataset, the ML model powered by Artificial Neural Networks is trained over 100 epochs. Post training, the model is capable of accurately predicting alphabets by accepting user inputs. The identified alphabet is pronounced clearly, leveraging Google's Text-to Speech algorithm. Each process is explained in detail in the following paras. The methodology flow can be found in Fig. 1.

2.1. Speech Sound Production Model

The following sections explains in detail the processes followed in the identification and development of a phonetic model for the underrepresented alphabets in the English language.

2.1.1. Identification of Unsorted English Alphabets

Initial phase of the research was regarding the production of speech sounds, particularly within the realms of phonetics and speech production. Concentration was on gaining a comprehensive understanding of how various alphabets are produced, focusing particularly on the involvement of the larynx and glottis in their generation. A pivotal aspect of this investigation involved categorising certain alphabets – B, F, H, M, N, O and P as non-articulatory, based on the substantial roles played by the airflow within the larynx and glottis in their pronunciation. This classification provided valuable insights into the nuanced relationship between the precise pronunciation of these alphabets and the coordinated movements of the vocal cord folds and the glottal space.

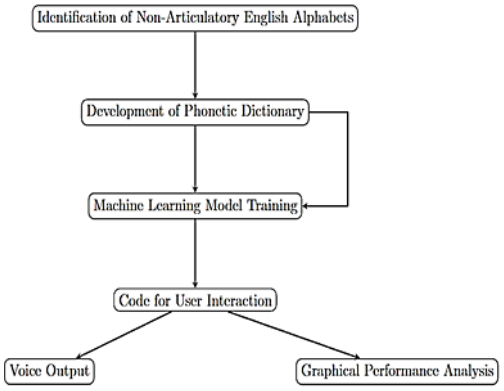


Fig. 1. Methodology Flow.

2.1.2. Categorization of Non-articulatory English Alphabets

After identifying a distinct subset of English alphabets, it is observed that their pronunciation is directly linked to the conservative regulation of oral cavity movements. This unique characteristic sets them apart from their counterparts, which predominantly rely on the larynx for pronunciation.

2.1.3. Development of Phonetic Dictionary

Following a new reliable approach leading to the development of a comprehensive phonetic dictionary encompassing combinations that accurately replicate the sounds of the identified non-articulatory alphabet. Using the concepts of Fast Fourier Transform, a self-made database encompassing non-articulatory alphabets represented by their corresponding known phonetic combinations was created and the same shown in Table 1.

The phonetic dictionary (See Table 1), comprises a concise dataset of 42 samples and serves as the training set for Machine Learning Model, discussed in Section 2.2.

Table 1. Phonetic Dictionary.

F	M	N	O	P	B	H
f	m	n	o	p	b	ha
ph	mm	nn	oo	pp	bb	he
fr	mb	ng	oe	ph	bh	hi
ff	mn	nk	ow	pf	pb	ho
pf	mp	kn	au	pl	mb	hu
lf	hm	gn	aw	pr	nb	
ft						

2.2. Machine Learning Model

LSTMs can be used to model the temporal dependencies in the speech sounds production process. The sequential nature of phonetic data can be effectively captured by LSTM networks, making them suitable for capturing the intricate relationships between different phonetic elements over time. However, in this scenario LSTMs are prone to overfitting due to the limited training data set setting a drawback overall [7].

Random Forests, considered as "black box" models, make it difficult to interpret the decision-making process. Understanding why the model makes a specific prediction can be challenging, which might be crucial especially in this application and they can overfit to outliers if not appropriately tuned [7].

RNNs could be used to model the sequential nature of phonetic data. They have the capability to process input sequences and maintain hidden states that capture temporal information. However, compared to LSTMs, they may struggle with long-term dependencies which is crucial for the precise pronunciation of the combinational phonemes [7].

Therefore, the model utilises Artificial Neural Networks (ANN) with TensorFlow libraries, a sophisticated class of ML models that consists of interconnected nodes organised into layers, including the input layer, hidden layers, and the output layer.

The paper focuses on the application of Artificial Neural Networks (ANN) to a Multi-Class Classification task. The main objective is to leverage the network's capabilities to predict the 7 class labels (F, M, N, O, P, B, H) based on a set of features and hence generate speech output.

2.2.1. ANN Model Design

A sequential model with 3 dense layers is leveraged for this task.

- The First Layer or the Input Layer is a fully connected dense layer with 64 units and uses ReLU (Rectified Linear Unit) Activation function. This function introduces non-linearities to the model, allowing it to learn complex patterns and relationships in the data.
- Following the input layer, there is a dropout layer with a rate of 0.5 which helps prevent overfitting by randomly setting a fraction of input units to zero during training. This step is crucial, considering the minimal database.
- Following the drop-out there is a Second Dense Layer with 32 units with ReLU activation.
- Following the Second Hidden Layer comes the Output Layer which utilises the robust SoftMax function, perfectly suited for multi-class predictions. The number of output neurons is adjusted so as to match the precise number of classes within the dataset. Throughout the training process, the model adjusts weights and biases consecutively, aiming to discern and predict the non-articulatory alphabet based on phonemic combinations.

2.2.2. Data Preparation

Next, the features and labels are defined as arrays and encoded using a LabelEncoder for optimal processing. Furthermore, a StandardScaler is employed to scale features for improved training. These steps ensure that the data is prepared in a desirable format for the neural network.

2.3. Model Evaluation

The performance of Phonetic-ANN model was rigorously evaluated implementing the StratifiedKFold cross-validation with five splits. This step helps train the model with diverse data subsets and gain valuable insights into its generalizability. Using ADAM as the optimizer, the model was trained for 100 epochs following the cross-validation which further refines its performance [7]. The final accuracy in the upwards of 88 % serves as a testament to the model's effectiveness to tackle classification tasks.

2.3.1. Interactive Model Interface

In addition to the core function of prediction, the code allows users to interact with the model by providing new data through an innovative interactive element. Asking few user inputs like 'Vowel/Consonant', 'Word/Phoneme Combination', 'Lip Rounding', 'Voiced/Voiceless', the provided data is scaled and encoded before being fed to the neural network, which enables real-time application and immediate feedback. The predicted class is presented, offering further exploration and feedback. To enhance the code further, integrating the gTTS library to generate text-to-speech audio based on the predicted class label. This enhances accessibility to the users by

allowing them to hear the predicted class label, improving overall clarity and interaction.

2.3.2. Deep Analysis of Model Performance

The model was further analysed deeply to generate a comprehensive classification report and a confusion matrix (See Fig. 6), providing deeper insights into the model's performance across different classes, including identifying potential biases or misclassifications. Also plots a confusion matrix and a ROC curve for visualising the performance of the model, creates a pair plot and a count plot for analysing label distribution of predicted labels for better understanding.

Despite the limited dataset (See Table 1), our primary objective was to achieve high accuracy, emphasising the efficiency and effectiveness of our code. Hence, through rigorous implementation of Artificial Neural Networks (ANN) in a Multi-Class Classification task, this model strategically leveraged the network's capabilities to accurately predict the 7 class labels based on a comprehensive set of features leading to a deep analysis of its performance and accuracy.

3. Results

Understanding relationship between pairs of features in a dataset, particularly for the relationships between multiple variables simultaneously an insightful plot called Pair Plot (See Fig. 2), also known as a pairs plot or scatterplot matrix, was used. Essential for detecting outliers, understanding data distributions, and guiding feature selection and dimensionality reduction during preprocessing and feature understanding. Aids in feature selection, potential transformations and guiding dimensionality reduction techniques. Each scatter plot represents a pair of features, and the points are colour-coded based on the true labels of the data. The diagonal plots show the distribution of individual features.

The subplot in the upper left corner (See Fig. 2), the X-Axis is labelled "Feature 1" and the Y-Axis is labelled "Feature 2". The data points in this subplot show that there is a positive correlation between Feature 1 and Feature 2. This means that as the value of Feature 1 increases, the value of Feature 2 also tends to increase. The subplot in the lower right corner, the X-Axis is labelled "Feature 3" and the Y-Axis is labelled "Feature 4". The data points in this subplot show that there is a weak negative correlation between Feature 3 and Feature 4. This means that as the value of Feature 3 increases, the value of Feature 4 tends to decrease, but the relationship is not very strong.

While comparing different optimizers like Stochastic Gradient Descent (SGD), Root Mean Square Propagation (RMSProp), Adaptive Gradient Algorithm (Adagrad), Adaptive Moment Estimation (Adam), as well as different size of neural architecture, epochs and batch size, the aforesaid architecture (refer Section 2) was found to be the best fit with 100 epochs. However, it was observed that the accuracy did not show substantial development with other combinations of architecture.

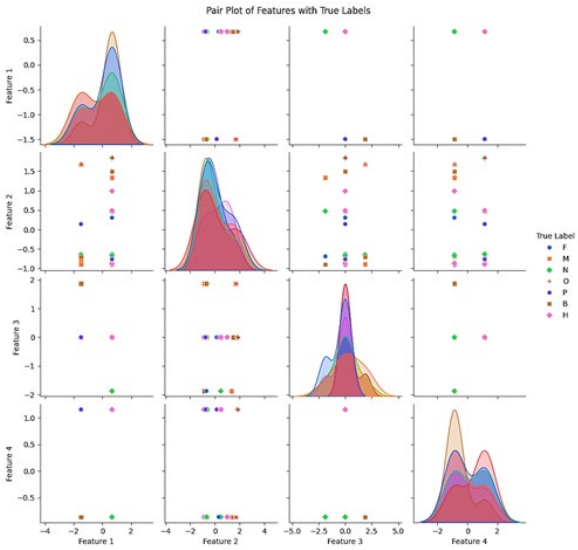


Fig. 2. Pair Plot visualising pairwise relation between Features.

Analysing the previous approaches, LSTMs have 74 % accuracy indicating the model's effectiveness in predicting non-articulatory English alphabets based on phonetic combinations, which is the least. Due to the limited dataset the accuracy of the model is unsatisfactory. RNN model achieved a 78 % accuracy. Random Forest model performed relatively well in making predictions based on the provided dataset featuring an accuracy of 83 % overall. The ANN Multi-class classifier outperforms its predecessors with a mighty accuracy score in the excess of 88 % as seen in Fig. 3.

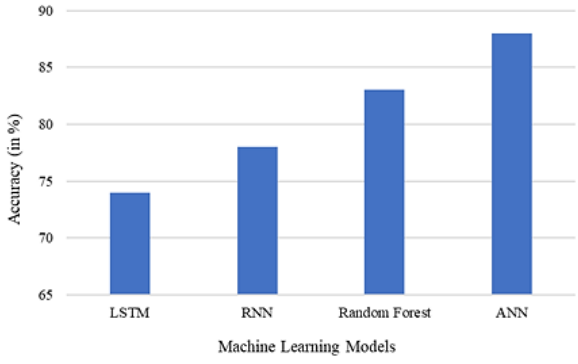


Fig. 3. Bar Graph comparing the Accuracies of various models.

The code being trained over 100 epochs where the lines at the top show that the training process reached

the desired 100 iterations. An epoch is one iteration over the entire training dataset. The lines in the middle show the accuracy and loss scores for each epoch. As the training progressed, the accuracy score increased and the loss score decreased, which indicates that the model was learning to make better predictions. The bottom line shows the final accuracy score, which was 88.1 %.

After training the model on the phonetic dictionary dataset comprising 42 samples of non-articulatory English alphabets, there has been an impressive accuracy of 88.1 %. This indicates that our model has successfully learned to predict the non-articulatory alphabet based on phonetic combinations, showcasing its robustness even with a limited dataset.

Fig. 4 shows the interactive experience, allowing users to type in features of the phoneme combination to display the predicted alphabet. Further, the accurate voice output generated with Google's Text-to-Speech algorithm can be seen in Fig. 4.

The example shown in Fig. 4 was tested for the sound *'Oh'*, the inputs are given as:

1. Vowel (0) or Consonant (1) - 0 (Oh sounds like the vowel O);
2. Word or Phoneme Combination – 158 (15 representing alphabet /O/, 8 representing alphabet /H/);
3. Lip Rounding (0 to 2 with 0 being least) – 2;
4. Voiced (0) or Voiceless (1) – 0.

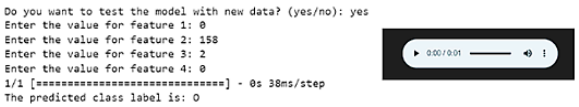


Fig. 4. Interactive UI accepting feature inputs, in turn predicting the alphabet with Voice Output.

The ROC curve shown (See Fig. 5) illustrates the performance of a classifier in a binary classification task. Each line represents the ROC curve for one of the 7 classes in the dataset.

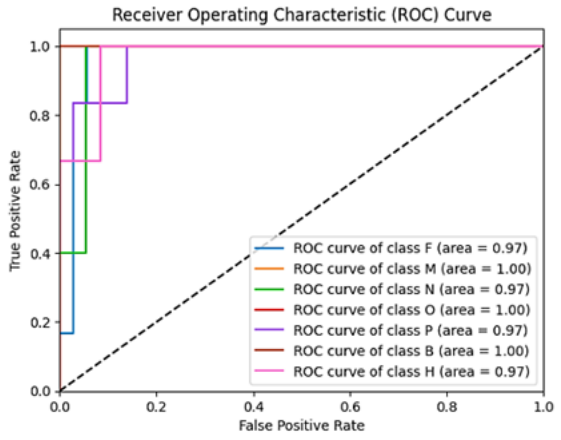


Fig. 5. ROC Curve.

ROC curves for all seven classes have high AUCs implying that the model is performing well at classifying all of the classes. The ROC curve for class 1 has the highest AUC, at 1.00, which means that the model is perfectly classifying all of the cases in class 1. The ROC curve for classes F, N, P and H has the lowest AUC, at 0.97, which is still impressive and is in the close vicinity of 1.

The classification report (See Fig. 6a) provides a detailed assessment of how well the model performs for each class. The report is organised by class, and the metrics are computed for each class separately.

For instance, in class '0' the precision is 0.75, recall is 1.00, and F1-score is 0.82, the support indicates that there are 6 instances of class '0' in the dataset. The accuracy, macro avg, and weighted avg provide overall measures of model performance across all classes. In the Confusion Matrix (See Fig. 6b), each row represents the actual labels of the data points, each column represents the predicted labels and each cell in the matrix indicates the count of instances for a particular combination of true and predicted labels. For example, the cell (0,4) represents instances where the true label is 'B' but the model predicted 'N'.

While there has been research into speech synthesis and phonetic classification, there hasn't been extensive work specifically targeting the reproduction of non-articulatory English alphabets. The proposed approach of using multi-class classification could open up new avenues of research in this area. Neural networks have indeed been widely used in various areas of natural language processing, including speech recognition and synthesis. The effectiveness of this approach would likely depend on the quality of the training data and the architecture of the neural network model [8].

	precision	recall	f1-score	support
0	0.75	1.00	0.86	6
1	1.00	1.00	1.00	7
2	0.62	1.00	0.77	5
3	1.00	1.00	1.00	6
4	1.00	0.67	0.80	6
5	1.00	1.00	1.00	6
6	1.00	0.50	0.67	6
accuracy			0.88	42
macro avg	0.91	0.88	0.87	42
weighted avg	0.92	0.88	0.88	42

(a)

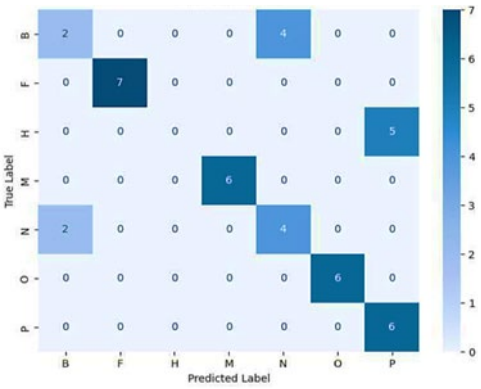


Fig. 6. (a) Classification Report; (b) Confusion Matrix.

The development of a minimal dictionary of phonetic combinations is an interesting approach, especially for non-articulatory alphabets where traditional phonetic rules may not apply. This aspect could be seen as an attempt to streamline the process and reduce complexity in reproducing these alphabets. Incorporating an interactive user experience into the predictive model adds a practical dimension to the research. This could enhance usability and accessibility, making the system more user-friendly and applicable in real-world scenarios. Overall, there are no pre-existing methods to address the challenge of reproducing non-articulatory English alphabets which combines elements of machine learning, linguistics, and user experience design.

4. Conclusions

This research marks a significant advancement in the comprehension and application of phonetics and speech production, particularly focusing on the interplay between laryngeal and oral cavity movements in generating diverse alphabets. The identification of a distinct unsorted English alphabet, development of a novel phonetic dictionary, and implementation of Artificial Neural Networks (ANN) have led to the creation of an effective classification model.

The use of TensorFlow libraries, a thoughtful architecture with ReLU activation, dropout layers, and SoftMax output demonstrates meticulous optimization despite the limited dataset. The inclusion of an interactive element in the code facilitates real-time user interaction and immediate feedback, while the integration of gTTS for Text-to-Speech audio enhances accessibility. Thorough analysis through classification reports, confusion matrices, ROC curves, and label distribution visualisations provides insight into the model's performance. Despite challenges with the dataset, achieving an accuracy of over 88 % underscores the code's robustness in addressing multi-class classification, emphasising its practicality in speech sound production research. Previous approaches with LSTM model (74 % accuracy), RNN model (78 % Accuracy) and, though competitive, Random Forest Model (83 % Accuracy) reinforce the superior performance of our Phonetic-ANN Model.

Therefore, this paper showcases potential for practical use, especially in aiding individuals with speech disabilities, underdeveloped articulators, strokes, hence filling a considerable global gap in supporting those with speech disorders. With the careful amalgamation of technology with phonetics fronted by an interactive User Experience, fosters a deeper understanding and thereby facilitating the acquisition of extensive knowledge.

References

[1]. University of Sheffield, How Phonetics is Studied, https://www.sheffield.ac.uk/linguistics/home/all-about-linguistics/about-website/branches-linguistics/phonetics/how-phonetics-studied

[2]. Pressbooks Pub, Chapter 3, Phonetics, https://pressbooks.pub/morethanwords/chapter/chapter-3-phonetics/.

[3]. P. Padmini, D. Gupta, M. Zakariah, Y. A. Alotaibi, K. Bhowmick, A simple speech production system based on formant estimation of a tongue articulatory system using human tongue orientation, *IEEE Access*, Vol. 9, 2021, pp. 4688-4710.

[4]. G. Gallagher, Total Identity in Co-occurrence Restrictions, *Berkeley Linguistics Society and Linguistic Society of America*, 2008.

[5]. Z. Jachova, L. Ristovska, J. Kovacevic, Phoneme recognition in speech threshold testing with disyllabic words, in *Proceedings of the International Scientific Conference "Improving the Quality of Life of Children and Youth"*, 2022.

[6]. S. Bhatt, S. Bansal, A. Kumar, S. K. Pandey, M. K. Ojha, K. U. Singh, S. Chakraborty, T. Singh, C. Swarup, A comprehensive examination of phoneme recognition in automatic speech recognition systems, *Traitement du Signal*, Vol. 40, Issue 5, 2023, pp. 1997-2008.

[7]. J. Vanek, J. Michalek, J. Psutka, Recurrent DNNs and its ensembles on the TIMIT phone recognition task, *arXiv Preprint*, 2018, arXiv:1806.07186.

[8]. Pangeanic: Neural Networks and How They Work in Natural Language Processing, https://blog.pangeanic.com/neural-networks-and-how-they-work-in-natural-language-processing

(027)

An Image-based Deep Learning Approach for the Automated Detection of Knee Arthroplasty Failure

A. Corti [1], M. Loppini [2], K. Chiappetta [2] and V. D. A. Corino [1,3]
[1] Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milan, Italy
[2] IRCCS Humanitas Research Hospital, Rozzano, Italy
[3] Cardio Tech-Lab, Centro Cardiologico Monzino IRCCS, Milan, Italy
E-mail: anna.corti@polimi.it

Summary: The rising number of total knee arthroplasty (TKA) revisions combined with the inferior outcomes compared to the primary TKA highlights the critical need for early detection of primary TKA failure. The present work proposes an image-based deep learning (DL) model to automatically detect TKA failure from radiographs by applying a transfer learning approach from a previously developed hip prosthesis failure DL model. The dataset comprised 475 radiographs from 120 failed and 105 non-failed TKA patients, and were subdivided in training, validation, and test sets (236, 100 and 139, respectively). Following preprocessing phases, the images were analyzed using a pretrained DenseNet169 and applying transfer learning fine-tuning algorithms. In the test set, an accuracy of 0.83, sensitivity of 0.89, specificity of 0.77, F1 score of 0.84 and area under the curve (AUC) of 0.91 were achieved, demonstrating the potentialities of the developed DL approach in automatically detecting TKA failure from plain radiographs.

Keywords: Total knee arthroplasty, Prosthesis revision, Artificial intelligence, Predictive modeling, Image classification.

1. Introduction

Total knee arthroplasty (TKA) is the most effective orthopedic surgery for patients with advanced knee osteoarthritis (KOA) and presents a 10-year cumulative revision rate ranging from 3.5 % to 6 % [1], [2]. Nowadays, the global number of TKA is massively increasing due to the increased longevity of the population and the higher prevalence of knee arthritis [3], and it is projected to grow by 85 % by 2030. Consequently, an increase in revision TKA is also expected, as demonstrated by a study projecting a growth in revision TKA by 600 % between 2005 and 2030 in USA [4]. Revision TKA procedures are more complex, costly, and associated with decreased implant longevity and suboptimal patient-reported outcomes when compared to primary TKA [5, 6]. The growing number of revisions presents challenges for the health care systems and the early detection of primary TKA failure has become of utmost importance in the orthopedic field.

In this context, machine learning (ML) models have gained prominence in the field of TKA orthopedic surgery, with earlier efforts aimed at predicting various outcomes of TKA, including complications, length of hospital stay, costs, patient satisfaction, functional outcome and revision [7, 8]. As regards TKA revision prediction, recent studies developed ML prediction models for revision TKA based on clinical [9, 10] or on radiographical data [11, 12]. Focusing on the image-based predictive models, both studies aimed to predict implant loosening: the study by Shah et al. [11] considered pre-operative radiographs and achieved an accuracy of 85.8 % on a test set of 138 patients, while the study by Lau et al. [12] considered post-operative radiographs and achieved an accuracy of 96.3 % on a test set of 95 radiographs. However, so far, an image-based deep learning (DL) model for the prediction of TKA failure (not limited to loosening) has not been proposed.

Our research group recently developed an image-based DL predictive model for the automatic identification of hip prosthesis failure from post-operative plain radiographs, achieving an accuracy of 0.97 on the test set [13, 14]. Considering the remarkable performance of our developed DL model, we seek to investigate whether a transfer learning approach from the hip prosthesis failure model can accurately detect primary TKA failure from post-operative plain radiographs. To the best of the authors' knowledge a similar approach has never been investigated so far.

2. Methods

2.1. Patient Dataset

225 Patients included in this study were retrospectively collected from the digital medical records at Humanitas Research Hospital and Ospedale Santa Corona di Pietra Ligure, Italy, between 2000 and 2019. All the radiographic images were provided by the Clinical and Radiographic Arthroplasty Register of Livio Sciutto Foundation Biomedical Research in Orthopedics – ONLUS. The study was approved by the Institutional Ethical Committee of Humanitas Research Hospital (prot. 408/19, approved on June 25, 2019), Italy, and all patients gave their written informed consent. 120 patients with TKA, who underwent total or partial revision due to implant failure in the considered period were included (failed

group). TKA failure encompassed loosening, dislocation, fracture, polyethylene wear and infection. A control group (non-failed group) was randomly collected from patients who underwent TKA in the same period (105 patients). To be eligible for the study, a minimum of one antero-posterior or lateral radiographic view of the implant needed to be available before revision surgery for the failed group and during follow-up time for the non-failed group. When patients of the non-failed group had undergone TKA for both knees, both implants were used for the analysis. Finally, 238 and 237 images were included for the failed and non-failed group, respectively.

2.2. Image Preprocessing

All images underwent the same preprocessing steps previously adopted for the development of the DL model of hip prosthesis failure [14]. Specifically, i) a gamma power transformation was applied to reduce the mist like effect and increase brightness [15]; ii) a sigmoidal function and the contrast-limited adaptive histogram equalization (CLAHE) method were applied to enhance contrast, thus highlighting the prosthesis compared to bone structures [16]; iii) a low pass-filtering operation was performed using a 2-D Gaussian smoothing kernel, eliminating frequencies above the cutoff frequency, which typically represent noise. Finally, the image was resized to a standard input dimension (224×224) and normalized using z-score standardization. Fig. 1 shows an example of initial and preprocessed image.

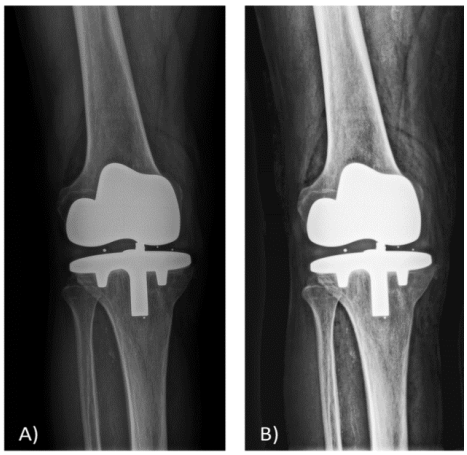


Fig. 1. Image preprocessing. A) Initial image. B) preprocessed image.

2.3. Model Development and Testing

The DL model was developed by applying transfer learning with fine-tuning algorithms from the DL model predicting hip prosthesis failure from plain radiographs [14]. Specifically, the hip prosthesis DL model was developed from the Densenet169 [17] pretrained for ImageNet [18], by replacing the Fully Connected layers of the original structure with a Global Average Pooling, a 128-Dense, a Dropout and 2-Dense layers and by applying a transfer learning fine-tuning approach [14]. To train the TKA failure model, layers of the pre-trained hip prosthesis model were frozen up to the first convolutional layer within the first dense block of the fourth stage.

The data were split into training, validation and test sets using a stratified approach: 20 % of the samples were reserved for model testing, and the remaining data were further divided into an 80-20 split for training and validation, respectively. The number of samples in the training, validation and test sets are reported in Table 1 for both failed and non-failed groups.

During training, the accuracy along epochs was evaluated to assess the network performance. The model performance on the test set was assessed by evaluating the sensitivity, specificity, accuracy, F1 score and AUC.

Table 1. Dataset.

Set	Tot	Failed	Non-failed
Training	236	118	118
Validation	100	50	50
Test	139	70	69

3. Results

Fig. 2 shows the training and validation accuracy as function of epochs, reaching a plateau around epoch 12. The model achieved a training and validation accuracy of 0.99 and 0.88, respectively.

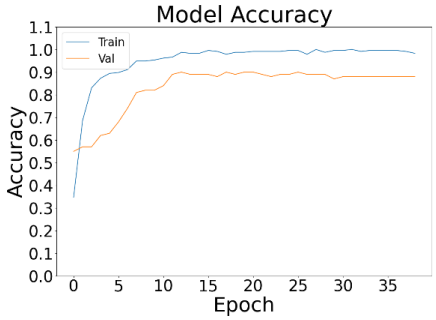


Fig. 2. Trend of training and validation accuracy.

When applied to the test set, the model presented a balanced accuracy of 0.83, sensitivity of 0.89, specificity of 0.77, F1 score of 0.84 and AUC of 0.91 (Figs. 3 and 4). Table 2 details the model performance in the validation and test sets. Sixty-two (over 70) images were correctly classified as failed, with a mean probability of 0.98±0.07 and 53 images (over 69) were correctly classified as non-failed with a mean probability of 0.96±0.10. Fig. 5 details the classification probabilities of the images of the failed and non-failed classes.

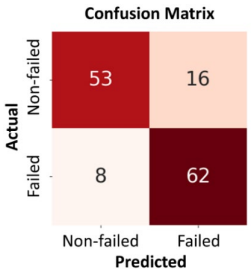


Fig. 3. Confusion matrix on the test set.

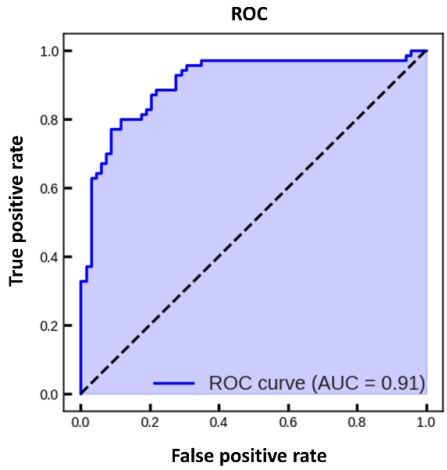


Fig. 4. Receiver Operating Characteristic (ROC) curve on the test set.

Table 2. Model performance.

Set	Validation	Test
Accuracy	0.88	0.83
Sensitivity	0.84	0.89
Specificity	0.92	0.77
F1 score	0.87	0.84
AUC	0.94	0.91

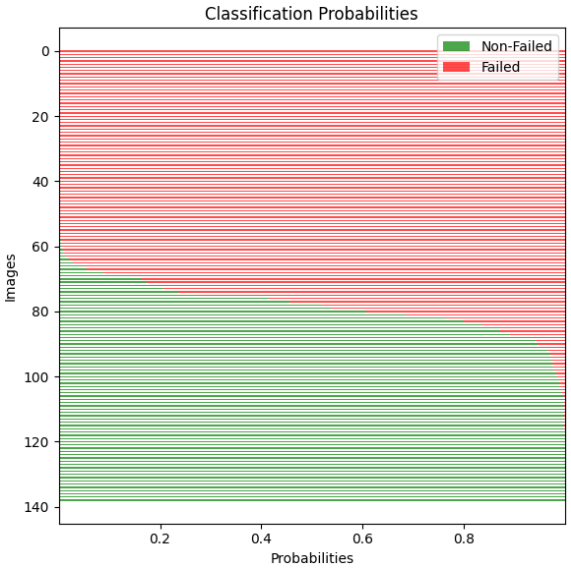


Fig. 5. Classification probabilities of the failed images in red and the non-failed images in green.

4. Conclusions

Nowadays, the global number of primary and revision TKA is continuously increasing and this trend is expected to continue. Revision TKA has a less favorable outcome than primary TKA, therefore the early detection of primary TKA failure may be beneficial. However, the early detection of TKA failure from radiographs can be challenging for clinicians. Moreover, the ever-growing population of TKA patients will represent a burden for the healthcare system with increased follow-up requirements. In this context, ML and DL model for predicting TKA failure have the potential to support the clinician diagnostic activity and alleviate their workload, offering automated TKA failure detection. To the best of the authors' knowledge, only two DL models were proposed to predict TKA failure from plain radiographs. Although their good results, these studies were focused on TKA loosening.

Herein, we developed a novel image-based DL model to predict TKA failure from post-operative radiographs, where "failure" encompasses the requirement for revision, extending beyond mere loosening. Moreover, we successfully demonstrated the effectiveness of a transfer learning fine-tuning approach from a previously developed DL model for hip prosthesis failure. This approach allowed us to achieve satisfying results (in line with studies in the literature) with a relatively small dataset.

In failure prediction for hip prosthesis, clinical variables have proved to be good predictors [19]. Thus, in future works, to improve the diagnostic performance, the image-based DL model will be integrated with patients' clinical information. Finally, to confirm the generalizability of the model, the DL pipeline should be tested on a larger multi-centric cohort of patients.

Acknowledgements

This research received funding from the Ministry of Health (grant number: GR-2018–12367275).

References

[1]. R. Civinini, et al., The survival of total knee arthroplasty: current data from registries on tribology: review article, *The Musculoskeletal Journal of Hospital for Special Surgery*, Vol. 13, Issue 1, 2017, pp. 28-31.

[2]. A. Klug, et al., The projected volume of primary and revision total knee arthroplasty will place an immense burden on future health care systems over the next 30 years, *Knee Surgery, Sports Traumatology, Arthroscopy*, Vol. 29, Issue 10, 2021, pp. 3287-3298.

[3]. M. Sloan, et al., Projected volume of primary total joint arthroplasty in the U.S., 2014 to 2030, *The Journal of Bone And Joint Surgery*, Vol. 100, Issue 17, 2018, pp. 1455-1460.

[4]. S. Kurtz, et al., Projections of primary and revision hip and knee arthroplasty in the United States from 2005 to

2030, *The Journal of Bone and Joint Surgery*, Vol. 89, Issue 4, 2007, pp. 780-785.

[5]. M. Bhandari, et al., Clinical and economic burden of revision knee arthroplasty, *Clinical Medicine Insights. Arthritis and Musculoskeletal Disorders*, Vol. 5, 2012, pp. 89-94.

[6]. M. D. Roman, et al., Outcomes in revision total knee arthroplasty (Review), *Experimental and Therapeutic Medicine*, Vol. 23, Issue 1, 2022, 29.

[7]. L. S. Lee, et al., Artificial intelligence in diagnosis of knee osteoarthritis and prediction of arthroplasty outcomes: a review, *Arthroplasty (London, England)*, Vol. 4, Issue 1, 2022, 16.

[8]. F. Hinterwimmer, et al., Machine learning in knee arthroplasty: specific data are key-a systematic review, *Knee Surgery, Sports Traumatology, Arthroscopy*, Vol. 30, Issue 2, 2022, pp. 376-388.

[9]. A. El-Galaly, et al., Can Machine-learning Algorithms Predict Early Revision TKA in the Danish Knee Arthroplasty Registry?, *Clinical Orthopaedics and Related Research*, Vol. 478, Issue 9, 2020, pp. 2088-2101.

[10]. J. D. Andersen, et al., Development of a multivariable prediction model for early revision of total knee arthroplasty – The effect of including patient-reported outcome measures, *Journal of Orthopaedics*, Vol. 24, 2021, pp. 216-221.

[11]. R. F. Shah, et al., Incremental inputs improve the automated detection of implant loosening using machine-learning algorithms, *The Bone & Joint Journal*, Vol. 102-B, Issue 6, 2020, pp. 101-106.

[12]. L. C. M. Lau, et al., A novel image-based machine learning model with superior accuracy and predictability for knee arthroplasty loosening detection and clinical decision making, *Journal of Orthopaedic Translation*, Vol. 36, 2022, pp. 177-183.

[13]. M. Loppini, et al., Automatic identification of failure in hip replacement: an artificial intelligence approach, *Bioengineering*, Vol. 9, Issue 7, 2022, 288.

[14]. F. Muscato, et al., Combining deep learning and machine learning for the automatic identification of hip prosthesis failure: Development, validation and explainability analysis., *International Journal of Medical Informatics*, Vol. 176, 2023, 105095.

[15]. Y. Ren, et al., Study on construction of a medical x-ray direct digital radiography system and hybrid preprocessing methods, *Computational and Mathematical Methods in Medicine*, Vol. 2014, 2014, 495729.

[16]. K. Zuiderveld, Contrast limited adaptive histogram equalization, in Graphics Gems {IV}, *Academic Press Professional Inc.*, USA, 1994, pp. 474-485.

[17]. G. Huang, et al., Densely Connected Convolutional Networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)*, 2017, pp. 2261-2269.

[18]. O. Russakovsky, et al., ImageNet large scale visual recognition challenge, *International Journal of Computer Vision*, Vol. 115, Issue 3, 2015, pp. 211-252.

[19]. M. Bulloni, et al., AI-based hip prosthesis failure prediction through evolutional radiological indices, *Archives of Orthopaedic and Trauma Surgery*, Vol. 144, Issue 2, 2024, pp. 895-907.

(028)

Deep Learning Based Detection of Concrete Cracks in Critical Underwater Infrastructure

U. Orinaitė, M. Pal, P. Palevicius and M. Ragulskis
Kaunas University of Technology, Department of Mathematical Modelling, Studentu 50-147,
Kaunas LT-51368, Lithuania
Tel.: + 37069822456
E-mail: minvydas.ragulskis@ktu.lt

Summary: A novel approach to automatic concrete crack identification in critical underwater infrastructure is presented in this paper. Optical effects in shallow underwater environment are accurately simulated a used to augment the standard dataset of cracked and intact concrete surfaces. 3D wave geometry based on the JONSWAP spectrum and ray tracing technology is used to render realistic underwater effects. It is demonstrated that a transfer learning approach can be efficiently used to perform reliable and accurate automatic identification of cracks in underwater infrastructure. The overall accuracy of crack identification in shallow underwater environment reaches 99.7 % what can be considered as a reliable approach to automatic routine inspection monitoring of critical infrastructure.

Keywords: Underwater optical effects, Crack detection, Machine learning, Transfer learning, Augmentation.

1. Introduction

Concrete structures are vital components of critical underwater infrastructure requiring durability in harsh environments. Ensuring their integrity is crucial to prevent possible accidents or even larger scale disasters. Routine inspection of such structures is limited by challenges related to the visibility and accessibility, making traditional inspection methods risky and inefficient. Techniques employing visual inspections by divers are costly and time-consuming. Emerging technologies based on automatic underwater drones offer promises but require further developments in terms of reliability, consistency, and accuracy. With the increasing underwater infrastructure, the reliability of crack detection methods becomes an essential factor. This paper addresses the existing gap in the existing technology by proposing efficient crack detection methods specifically tailored for underwater concrete structures. Early detection enables timely repairs, ensuring long-term safety and functionality of critical underwater infrastructure.

2. Dataset Overview and Image Augmentation Techniques

The dataset of 40000 images containing both cracked and intact surfaces from Ozgenel's concrete crack dataset [1] has been used. The neural network has been trained on a mix of "Positive" and "Negative" images, each sized at 227×227 pixels, and its accuracy assessed on a smaller test subset. To simulate underwater conditions, 60 realistic underwater caustic effects using Blender's Cycles were generated. Additionally, each of the 40000 images have been paired with a unique optical water effect mask, achieved through various modifications like rotation and zoom. These masks superimposed onto the concrete surface images using blending techniques to enhance crack visibility, a process applied across the entire dataset [2].

The interaction between light and underwater environments, influenced by seabed topography and streamer depth, impacts optical effects. Utilizing these variables in a wave propagation model enables the prediction of areas where such effects do occur. Wave models, crucial for surf forecasting, can be developed using phase averaging or resolving techniques, with modern iterations finding application in various engineering fields. This study employs a model to generate 3D wave geometry for image augmentation, based on the JONSWAP spectrum [3]. Rendering realistic underwater concrete structure images is essential, achieved through techniques detailed in [4]. A deep learning method is tested using a dataset of 40000 photos, augmented with underwater optical effects. The augmentation process involves diverse modifications and blending techniques to enhance dataset variability [murky problem].

3. Considerations for Shallow Water Optical Effects

Shallow water environments present unique challenges for optical imaging due to factors such as light refraction, scattering, and absorption. These optical effects can distort images, making crack detection in underwater structures challenging. Strategies to mitigate optical distortion include optimizing lighting conditions, using specialized optics, and employing image processing algorithms to correct for distortion. However, even with these strategies, shallow water optical effects can still impact

crack detection accuracy by altering the appearance of cracks and surrounding areas. Understanding and accounting for these effects is crucial for developing robust crack detection systems in underwater environments.

Examining 3D object representation in underwater settings entails understanding techniques for handling structural features, pivotal for integrating with deep learning models. Modeling non-flat surface images, however, already presents a number of important challenges, requiring innovative approaches to accurately represent complex structures [5]. Overcoming these obstacles is crucial for applications like concrete crack detection, where precise modeling of surface structures is essential. This comprehensive approach to analyzing and modeling underwater environments enables advancements in structural analysis and contributes to the broader field of underwater exploration and engineering.

In our study, we utilized image raytracing techniques to simulate the optical surface effects. Optical underwater effects are meticulously crafted and seamlessly blended onto the regenerated crack images, further enhancing their depth and authenticity. Subsequently, the applied shallow water optical effects are augmented to the existing dataset of images, enriching their visual complexity and mimicking real-world underwater environments. Through this comprehensive approach, we aimed to create a highly accurate and immersive representation of underwater structures, facilitating precise crack detection and analysis in challenging aquatic settings (Fig. 1).

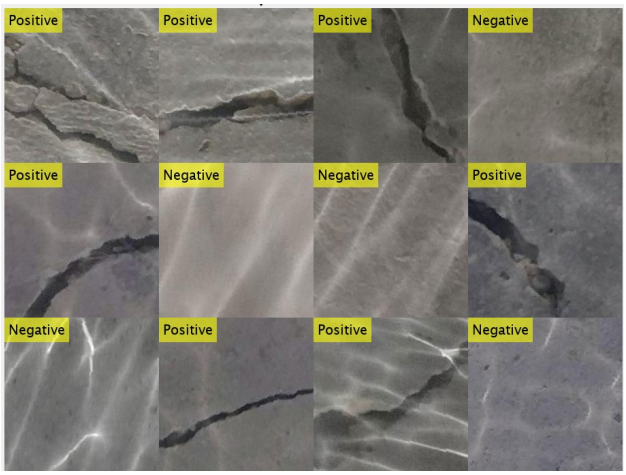


Fig. 1. The Ozgenel's concrete crack dataset augmented by the optical shallow underwater effects.

4. Results and Discussion

The transfer learning approach is adopted for machine learning tasks due to its versatility and time-saving benefits. Transfer learning allows the utilization of pre-trained deep learning networks, like SqueezeNet and AlexNet, eliminating the need to build networks from scratch. Leveraging pre-trained networks expedites the training process for new tasks, enhancing sensitivity in real-world applications within a shorter timeframe. [5, 6], guiding the precise methodology employed in this paper. The confusion matrix with underwater crack images dataset results into the overall accuracy of 99.7 % (Table 1).

Table 1. Confusion matrix.

True Class	Negative	5981	19
	Positive	20	5980
		Negative	Positive
		Predicted class	

References

[1]. Ç. F. Özgenel, Concrete Crack Images for Classification, Version 2, *Mendeley Data*, 2019.

[2]. U. Orinaitė, P. Palevičius, M. Pal, M. Ragulskis, A deep learning-based approach for automatic detection of concrete cracks below the waterline, *Vibroengineering Procedia*, Vol. 44, Aug. 2022, pp. 142-148.

[3]. K. Hasselann, et al., Measurements of wind-wave growth and swell decay, *Ergänzung zur Deut. Hydrogr. Z.*, Vol. 12, 1973, pp. 1-95.

[4]. K. Iwasaki, Y. Dobashi, T. Nishita, Efficient rendering of optical effects within water using graphics hardware, in *Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, 2001, pp. 374-383.

[5]. P. Palevičius, M. Pal, M. Landauskas, U. Orinaitė, I. Timofejeva, M. Ragulskis, Automatic detection of cracks on concrete surfaces in the presence of shadows, *Sensors*, Vol. 22, 2022, 3662.

[6]. U. Orinaitė, V. Karaliūtė, M. Pal, M. Ragulskis, Detecting underwater concrete cracks with machine learning: a clear vision of a murky problem, *Appl. Sci.*, Vol. 13, 2023, 7335.

(031)

Fuzzy Agent-based Simulation for Managing Battery Recharging for a Fleet of Autonomous Industrial Vehicles

J. Grosset [1,2], A.-J. Fougères [2], M. Djoko-Kouam [2,3] and J.-M. Bonnin [1]
[1] IMT Atantique, IRISA, UMR 6074, Rennes, France
[2] ECAM Rennes, Louis de Broglie, Campus de Ker Lann, Bruz, Rennes 35091, France
[3] IETR, UMR CNRS 6164, CentraleSupélec, Rennes, France
E-mail: juliette.grosset@ecam-rennes.fr

Summary: The article presents a multi-agent simulation utilizing fuzzy logic to explore battery recharging management for Autonomous Industrial Vehicles (AIVs). This approach offers adaptability and resilience through a distributed system, accommodating variations in AIV battery capacity. Results highlight the efficacy of adaptive fuzzy multi-agent models in optimizing recharging strategies, enhancing operational efficiency, and curbing energy consumption. Dynamic factors like workload variations and AIV-infrastructure communication are considered in the form of heuristics, emphasizing the significance of flexible, collaborative approaches in autonomous systems. Notably, infrastructure capable of optimizing recharging based on energy tariffs can significantly reduce consumption during peak hours, emphasizing the importance of such strategies in dynamic environments. Overall, the study underscores the potential of incorporating adaptive fuzzy multi-agent models for AIV energy management to drive efficiency and sustainability in industrial operations.

Keywords: Cooperative mobile robots, Recharging battery management, Fuzzy logic, Multi-agent simulation, Airport 4.0.

1. Introduction

Industry 4.0 is coming with a high degree of digitalisation of industrial processes, but also a significant increase in communication and cooperation between the machines that make it up. This is the case with autonomous industrial vehicles (AIVs) and other cooperative mobile robots that are proliferating in factories or airports, and whose intelligence and autonomy are increasing.

The deployment of AIV fleets raises several issues, all of which related to their actual level of autonomy: acceptance by employees, vehicle localization, traffic flow, collision detection, and vehicle perception of changing environments. Simulation allows us to take into account the different constraints and requirements formulated by manufacturers and future users of these AIVs.

Before starting to test AIV traffic scenarios on a large scale in sometimes complex industrial or airport situations, it is essential to simulate these scenarios [1]. One significant benefit of running simulations is that usable results without the need to applying a scaling factor.

The main benefits of simulating AIV operations are extensively presented by Tsolakis et al. [2]: simulation reduces the development time and cost of an AIV, minimises the potential operational risks associated with the AIV, enables the feasibility of different AIVs scenarios to be assessed at a strategic or operational level, provides a rapid understanding of AIV operations (under conditions of limited data availability), and identifies improvements in facility layout configurations hosting AIVs.

The simulation also provides flexibility in terms of deployment and redeployment, and enables us to study the sharing of responsibility between the central server and the robots (local/global balance) for the various operational decisions. Another advantage of simulations is to introduce humans into the scenarios in order to convince people, before the actual deployment of autonomous mobile robots, of the safe nature of the coexistence and possible interactions between these future mobile robots and human operators in industry [3].

Agent-based approaches are often proposed for the simulation of autonomous vehicles [4], including path planning in a large-scale context [5], or optimal task allocation with collision and obstacle avoidance [6].

Our current research focuses on the use of fuzzy agents to manage the levels of imprecision and uncertainty involved in modelling the behaviour of simulated vehicles [7]. Fuzzy set theory is well suited to the processing of uncertain or imprecise information that must lead to decision-making by autonomous agents [8]. The concept of the fuzzy agent can therefore be proposed as a partial implementation of this theory.

Most of the control tasks performed by autonomous mobile robots (*perception, localisation, mapping, path and task planning, navigation and motion control, obstacle avoidance, communication, and energy control* [9]) have been the subject of performance improvement studies using fuzzy logic:

1) Navigation of mobile robots from conceptual, theoretical or application points of view [10], navigation of several mobile robots [11], navigation and control of a mobile robot in an unknown environment in real time [12], and comparison of navigation performance of mobile robots obtained using fuzzy logic or neural networks [13];

2) Obstacle avoidance from conceptual and systemic points of view in an unknown dynamic environment [14];
3) Path planning strategies focusing on obstacle avoidance [15] or global navigation [16];
4) Motion planning [17];
5) Localisation of mobile robots [18];
6) Intelligent management of energy consumption [19].

An agent-based system is fuzzy if its agents have fuzzy behaviours or if the knowledge they use is fuzzy. This means that agents can have: 1) fuzzy knowledge (fuzzy decision rules, fuzzy linguistic variables, and fuzzy linguistic values); 2) fuzzy behaviours (the behaviours adopted by the agents as a result of fuzzy inferences); and 3) fuzzy interactions, organisations or roles [20].

Fuzzy agents can follow the evolution of fuzzy information coming from their environment and from the agents [21]. By interpreting the fuzzy information they receive or perceive, fuzzy agents interact within a multi-agent system; they can also interact in a fuzzy manner. For example, a fuzzy agent can discriminate a fuzzy interaction value to evaluate its degree of affinity (or interest) with another fuzzy agent [22].

2. Fuzzy Agent-based Simulation

The different elements of the fuzzy agent model are as follows [7]: (1) the agent-based fuzzy system; (2) the behaviour of a fuzzy agent, inspired by perceive-decide-act feedback loops [23]; (3-5) the behavioural functions of a fuzzy agent; (6) and the fuzzy interactions between two fuzzy agents.

Table 1. Fuzzy agent model used in our simulations [7, 24].

$\widetilde{M}_{\alpha} = < \widetilde{A}, \widetilde{I}, \widetilde{P}, \widetilde{O} >$ (1)
where A is a set of agents, $A = \{\alpha_1, ..., \alpha_n\}$; $\widetilde{A}$ is a set of fuzzy agents, $\widetilde{A} = \{\widetilde{\alpha}_1, ..., \widetilde{\alpha}_m\}$ with $\widetilde{A} \subseteq A$; $\widetilde{I}$ is a set of fuzzy interactions between fuzzy agents; $\widetilde{P}$ is a set of fuzzy roles filled by fuzzy agents; and $\widetilde{O}$ is a set of fuzzy organisations defined for fuzzy agents (subsets of strongly related fuzzy agents).
$\widetilde{\alpha}_i = \left\langle \Phi_{\Pi(\widetilde{\alpha}_i)}, \Phi_{\Delta(\widetilde{\alpha}_i)}, \Phi_{\Gamma(\widetilde{\alpha}_i)}, K_{\widetilde{\alpha}_i} \right\rangle$ (2)
where, for a fuzzy agent $\widetilde{\alpha}_i$, $\Phi_{\Pi(\widetilde{\alpha}_i)}$ is its observation function, $\Phi_{\Delta(\widetilde{\alpha}_i)}$ its decision-making function, $\Phi_{\Gamma(\widetilde{\alpha}_i)}$ its action function and $K_{\widetilde{\alpha}_i}$ its knowledge base.
$\Phi_{\Pi(\widetilde{\alpha}_i)} : (E_{\widetilde{\alpha}_i} \cup I_{\widetilde{\alpha}_i}) \times \Sigma_{\widetilde{\alpha}_i} \rightarrow \Pi_{\widetilde{\alpha}_i}$ (3)
$\Phi_{\Delta(\widetilde{\alpha}_i)} : \Pi_{\widetilde{\alpha}_i} \times \Sigma_{\widetilde{\alpha}_i} \rightarrow \Delta_{\widetilde{\alpha}_i}$ (4)
$\Phi_{\Gamma(\widetilde{\alpha}_i)} : \Delta_{\widetilde{\alpha}_i} \times \Sigma \rightarrow \Gamma_{\widetilde{\alpha}_i}$ (5)
where, for a fuzzy agent $\widetilde{\alpha}_i$, $E_{\widetilde{\alpha}_i}$ is the set of fuzzy events observed, $I_{\widetilde{\alpha}_i}$ all its fuzzy interactions, $\Sigma_{\widetilde{\alpha}_i}$ all its fuzzy states, $\Pi_{\widetilde{\alpha}_i}$ all its fuzzy perceptions, $\Delta_{\widetilde{\alpha}_i}$ all its fuzzy decisions, $\Gamma_{\widetilde{\alpha}_i}$ all its fuzzy actions, and Σ is the state of the fuzzy multi-agent system $\widetilde{M}_{\alpha}$.
$\widetilde{\iota}_t = < \widetilde{\alpha}_s, \widetilde{\alpha}_r, \widetilde{\gamma}_c >$ (6)
where, for fuzzy interaction $\widetilde{\iota}_t$, $\widetilde{\alpha}_s$ is the fuzzy source agent, $\widetilde{\alpha}_r$ is the destination fuzzy agent, and $\widetilde{\gamma}_c$ is a fuzzy communication act (*inform, diffuse, ask, reply, …*).

3. Case Study: Autonomous Management of Battery Recharging

We present an adaptable fuzzy multi-agent model (Fig. 1) that addresses the challenges of energy management for AIVs. Efficient management of AIVs requires a holistic approach that takes into account several factors, including operational availability, energy consumption [25], collaboration between AIVs and the dynamic infrastructure, and their adaptation to changing conditions. We aim to optimise recharging based on energy costs, as a low workload combined with frequent recharging can increase the overall energy consumption of the system. In addition, poor anticipation can limit system availability.

AIV missions do not follow a uniform distribution in terms of frequency, creating periods of intense activity and others that are quieter. It is therefore essential to link the energy consumption of AIVs to the amount of work carried out and their operational availability.

To avoid an overload of recharging requests due to too many simultaneous requests, the AIVs need to work together by communicating with each other or via the infrastructure. As for automatic recharging, although it solves the problem of the number of charges, it requires space and consumes energy. Even a 2 to 3 % reduction in energy consumption is significant for certain warehouses and airports. For the introduction of fleets of autonomous vehicles in the

industry of the future, it therefore seems necessary to fine-tune the number of recharging points. This sizing can be improved by taking into account the possibilities for communication between the AIVs, which can collectively avoid critical (urgent) recharging.

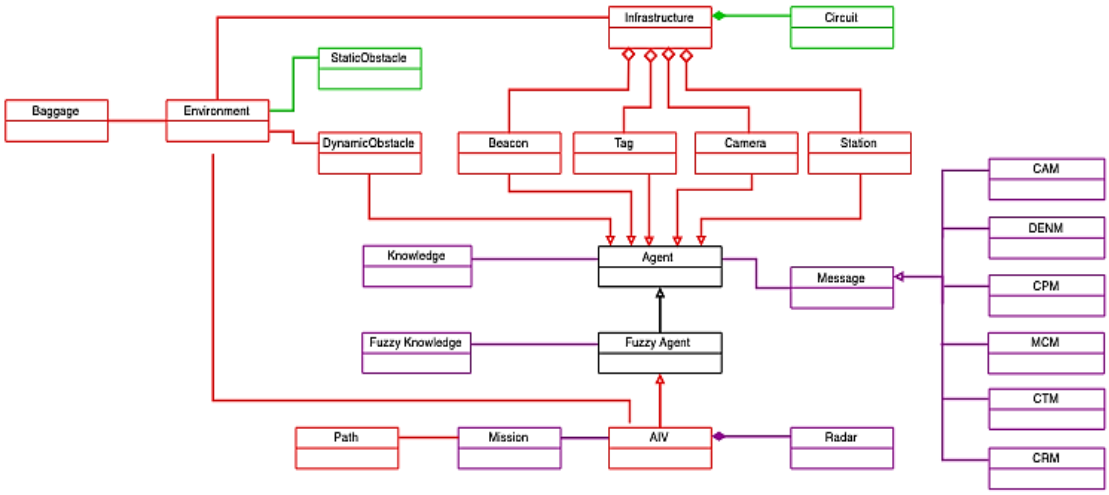


Fig. 1. Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple.

3.1. Description of the Simulation Framework

To test different autonomous management strategies for solving the problem of AIVs recharging batteries, we defined an initial scenario, which we will refer to as the basic scenario (Fig. 2). We made several improvements to this basic scenario and compared the number of missions carried out **(1)**, the number of recharges performed **(2)**, the average time taken to complete a mission in seconds **(3),** and waiting times for recharging in seconds **(4)**. We also varied the charge threshold at which an AIV must recharge its battery. We then introduced a fuzzy inference system to determine the recharge time. We also varied the values of the fuzzy model (fuzzy linguistic values).

3.2. Comparisons between Thresholds and Fuzzy Logic Models

In this section, we delve into a comparative analysis between different thresholds and fuzzy logic models. We propose 3 different scenarios:

- Scenario 1 (or 'Sc1'), which corresponds to a Basic Scenario;
- Scenario 2 (or 'Sc2'), where different threshold values are tested in the context of scenario 1;
- Scenario 3 (or 'Sc3'), where AIVs use a fuzzy logic model for recharge.

We simulated these three scenarios for 1000 baggages (a discussion regarding the scenario results is provided in the following three sections). The temporal results are shown in Table 3. We aim to discern the optimal threshold configurations that maximise mission throughput, minimise recharging frequency, and optimise resource utilisation, thereby improving the overall efficiency of autonomous management strategies for recharging the AIV battery.

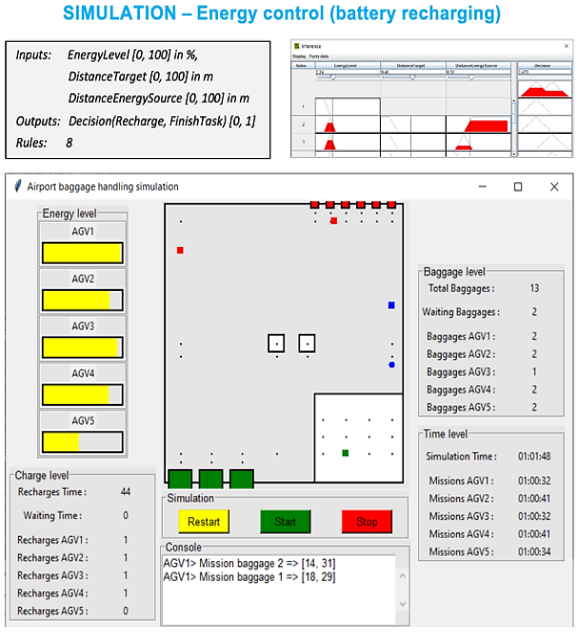


Fig. 2. Simulation Application.

Table 2. Time results for 1000 baggages for Sc1, Sc2 and Sc3.

Scenarios	Sc1	Sc2	Sc3
Number of baggages	1000	1000	1000
Total recharge time (s)	3675	3535	3561
Total simulation time (hour:minutes: seconds)	04:36:46	04:35:11	04:34:58

3.2.a. Basic Scenario

In the "Basic Scenario", AIVs have a single threshold model set at 30 % for recharge. This scenario makes it possible to compare performance in terms of mission processing time (overall and individual time), number of recharges, and waiting time for recharges (access to a free station). The AIVs results for Scenario 1 are shown in Table 3.

Table 3. AIVs results for Scenario 1.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
Thresholds	30	30	30	30	30	
(1)	201	200	199	200	200	1000
(2)	67	67	66	67	67	334
(3)	80	80	80	80	80	80
(4)	0	0	0	11	23	34

3.2.b. Different threshold values

Scenario 2 enables us to compare different threshold values for AIVs recharge. When we compare with thresholds varying between 15 % and 30 %, the overall mission processing time is slightly lower, and the number of recharges and overall recharge time are also lower (297 and 3535, respectively). The performance of AIV1 with the lowest threshold (15 %) is obviously the best, although there is a greater risk of not being able to reach a station due to a lack of charge in the event of an incident!

Table 4. AIV results for Scenario 2.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
Thresholds	15	20	25	30	35	
(1)	202	200	200	199	199	1000
(2)	50	57	57	66	67	295
(3)	79	80	80	77	80	79.2
(4)	31	38	27	2	13	111

3.2.c. Fuzzy Logic Model

In comparison with Scenario 1, where AIVs have a threshold of 30 %, in Scenario 3, AIVs use a fuzzy basic model. The results presented in Table 5 demonstrate an improvement in overall and individual AIV times (79.4 seconds on average instead of 80 seconds) and fewer recharges (285 recharges instead of 334).

Table 5. AIV results for Scenario 3.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
FL model	FL	FL	FL	FL	FL	
(1)	200	200	200	200	200	1000
(2)	57	57	57	57	57	285
(3)	80	80	80	80	77	79.4
(4)	0	19	0	15	0	34

3.3. Increases in Fuzzy Logic Criteria

To improve the results of the previous simulations, we made 3 types of adaptation (heuristics), taking into account more realistic constraints and the possibility of AIVs communicating with each other and with infrastructure elements such as charging points:

1) Adaptation of recharging according to the needs of the AIVs and the availability of the charging points (centralised scenario by supervision and decentralised scenario by communication between the AIVs and the charging points);
2) Adaptation of recharging according to the rate of baggage arrival and the resulting variation in activity (the number of missions to be performed by the AIVs in a unit of time is no longer constant);
3) Adapting the speed of the AIVs according to the rate of baggage arrival (centralised scenario by supervision and decentralised scenario by communication between the AIVs and the charging points).

The objective of this section is to show that specific heuristics allow certain situations to be dealt with fairly finely and increase the collective/overall performance of AIVs. We simulated these three improved scenarios for 1000 baggages. The temporal results are shown in Table 6.

Table 6. Time results and configuration for 1000 baggages for Sc4, Sc5 and Sc6.

Scenarios	**Sc4**	**Sc5**	**Sc6**
Number of baggages	1000	1000	1000
Total recharge time (s)	3528	3574	11807
Total simulation time (hour:minutes:seconds)	03:59:06	03:51:08	02:25:00
Maximum number waiting baggages	486	650	499
Average Baggages Waiting	242	322	266

3.3.a. Adapting Recharging to Demand and the Availability of Charging Points

Scenario 4 simulates the adaptation of charging to demand and the availability of charging points. The AIV results are shown below, in Table 7. The effectiveness of this heuristic is clearly visible, especially for AIV1: 14 fewer recharges than for AIV5, and 15 fewer than for AIV4. The total recharging time is also shorter than for scenarios 1 and 2: 3528 seconds instead of 3675 seconds and 3535 seconds.

3.3.b. Adaptation of Recharging According to the Baggage Arrival Rate

Scenario 5 simulates an adaptation of recharging as a function of the baggage arrival rate and therefore of

the variation in induced activity (the number of tasks to be carried out by the AIVs). Table 8 shows that the adaptation of recharging enables AIVs to complete their missions more quickly than in scenario 4. In fact, they complete 1 mission in 66 seconds on average, compared with 69 seconds for Scenario 4.

Table 7. AIV results for Scenario 4.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
Thresholds	15/15	20/15	25/20	30/20	35/25	
(1)	202	201	200	200	197	1000
(2)	50	56	57	66	65	294
(3)	69	69	69	69	69	69
(4)	117	11	44	14	0	186

Table 8. AIV results for Scenario 5.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
Thresholds	20	20	20	20	20	
(1)	201	200	201	200	200	1000
(2)	57	57	58	57	57	286
(3)	66	66	66	66	66	66
(4)	0	46	0	0	24	70

3.3.c. Adapting the Speed of the AIVs to the Flow of Baggage Arrivals

In scenario 6, we propose to adapt the speed of the AIVs to the flow of baggage arrivals. Compared with scenario 5, the 30 % threshold has been adapted (the 20 % threshold causing too many load faults due to the increase in energy consumption in cases of faster speed). The overall simulation time is much shorter despite a much longer overall reload time, as presented in Table 6. Moreover, Table 9 shows that the throughput is a little better controlled since the baggage waiting time is 266 seconds in this scenario instead of 332 seconds for scenario 5, presented in Table 8.

Table 9. AIV results for Scenario 6.

Indicators	AIV1	AIV2	AIV3	AIV4	AIV5	Global
Thresholds	20	20	20	20	20	20
(1)	199	203	198	203	197	1000
(2)	195	200	194	200	194	983
(3)	41	40	41	40	41	40.6
(4)	330	39	342	20	343	1074

4. Conclusions

We have developed a multi-agent simulation, including fuzzy logic, to test various scenarios of battery recharging management. This approach offers a flexible adaptation to the various aspects of AIV management and facilitates any adjustments required for deployment on the industrial site. The use of a distributed system provides temporary autonomy in the event of failure of the central infrastructure, taking into account the individual differences in the battery capacity of the AIVs.

The simulation results demonstrate that incorporating adaptive fuzzy multi-agent models for AIV energy management can significantly optimize recharging strategies, improve operational efficiency, and mitigate energy consumption, particularly by considering dynamic factors such as workload variation, communication between AIVs and infrastructure elements. In fact, an infrastructure capable of optimising recharging according to energy tariffs is advantageous, particularly with the ability to cut consumption over an hour. These findings will underscore the importance of flexible, collaborative approaches in enhancing the performance of autonomous systems in dynamic environments.

We plan to continue integrating fuzzy models into our AIV simulation agents in order to increase the relevance and effectiveness of their decisions in the management of their energy recharge.

References

[1]. X. Hu, B. P. Zeigler, A simulation-based virtual environment to study cooperative robotic systems, *Integrated Computer-Aided Engineering*, Vol. 12, Issue 4, 2005, pp. 353-367.

[2]. N. Tsolakis, D. Bechtsis, J. S. Srai, Intelligent autonomous vehicles in digital supply chains: From conceptualisation, to simulation modelling, to real-world operations, *Business Process Management J.*, Vol. 25, Issue 3, 2019, pp. 414-437.

[3]. A. Hentout, M. Aouache, A. Maoudj, I. Akli, Human–robot interaction in industrial collaborative robotics: a literature review of the decade 2008-2017, *Advanced Robotics*, Vol. 33, Issue 15-16, 2019, pp. 764-799.

[4]. P. Jing, H. Hu, F. Zhan, Y. Chen, Y. Shi, Agent-based simulation of autonomous vehicles: A systematic literature review, *IEEE Access*, Vol. 8, 2020, pp. 79089-79103.

[5]. N. M. Kou, C. Peng, X. Yan, Z. Yang, et al., Multi-agent path planning with non-constant velocity motion, in *Proceedings of the 18th Int. Conference on Autonomous Agents and MultiAgent Systems*, 2019, pp. 2069-2071.

[6]. J. Grosset, A. Ndao, A.-J. Fougères, M. Djoko-Kouam, C. Couturier, J.-M. Bonnin, A cooperative approach to avoiding obstacles and collisions between autonomous industrial vehicles in a simulation platform, *Integrated Computer-Aided Engineering*, Vol. 30, Issue 1, 2023, pp. 19-40.

[7]. A.-J. Fougères. A modelling approach based on fuzzy agent, *Int. J. of Comp. Science Issues*, Vol. 9, Issue 6, 2013, pp. 19-28.

[8]. A.-J. Fougères, E. Ostrosi, Fuzzy agent-based approach for consensual design synthesis in product configuration, *Integrated Computer-Aided Engineering*, Vol. 20, Issue 3, 2013, pp. 259-274.

[9]. M. De Ryck, M. Versteyhe, F. Debrouwere. Automated guided vehicle systems, state-of-the-art control algorithms and techniques, *J. of Manufacturing Systems*, Vol. 54, 2020, pp. 152-173.

[10]. T. S. Hong, D. Nakhaeinia, B. Karasfi, Application of fuzzy logic in mobile robot navigation, in Fuzzy Logic: Controls, Concepts, Theories and Applications (E. P. Dadios, Ed.), *IntechOpen*, United Kingdom, 2012.

[11]. S. K. Pradhan, D. R. Parhi, et al., Fuzzy logic techniques for navigation of several mobile robots, *Applied Soft Computing*, Vol. 9, Issue 1, 2009, pp. 290-304.

[12]. V. Yerubandi, Y.M. Reddy, M.V. Kumar. Navigation system for an autonomous robot using fuzzy logic, *Int. J. of Scientific and Research Publications*, Vol. 5, Issue 2, 2015, pp. 5-8.

[13]. H. M. Yudha, T. Dewi, N. Hasana, et al., Performance comparison of fuzzy logic and neural network design for mobile robot navigation, in *Proceedings of the Int. Conference on Electrical Eng. and Comp. Sc*., 2019, pp. 79-84.

[14]. A. Meylani, A.S. Handayani, R.S. Carlos, et al., Different types of fuzzy logic in obstacles avoidance of mobile robot, in *Proceedings of the Int. Conference on Electrical. Eng. and Computer Sc*., 2018, pp. 93-100.

[15]. A. Shitsukane, W. Cheriuyot et al., A survey on obstacles avoidance mobile robot in static unknown environment, *Int. J. Comput.*, Vol. 28, Issue, 2018, pp. 160-173.

[16]. B. K. Patle, A. Pandey, D. R. K. Parhi, et al., A review: On path planning strategies for navigation of mobile robot, *Defence Technology*, Vol. 15, Issue 4, 2019, pp. 582-606.

[17]. A. Nasrinahar, J. H. Chuah, Intelligent motion planning of a mobile robot with dynamic obstacle avoidance, *Journal on Vehicle Routing Algorithms*, Vol. 1, Issue 2, 2018, pp. 89-104.

[18]. M. Alakhras, M. Oussalah, M. Hussein, A survey of fuzzy logic in wireless localization, *EURASIP J. on Wireless Com. and Networking*, Vol. 1, 2020, pp. 1-45.

[19]. M. F. R. Lee, A. Nugroho, Intelligent energy management system for mobile robot, *Sustainability*, Vol. 14, Issue 16, 2022, 10056.

[20]. E. Ostrosi, A.-J. Fougères, M. Ferney, Fuzzy agents for product configuration in collaborative and distributed design process, *Applied Soft Computing*, Vol. 8, Issue 12, 2012, pp. 2091-2105.

[21]. N. Ghasem-Aghaee, T. I. Ören, Towards fuzzy agents with dynamic personality for human behavior simulation, in Proceedings of the *Summer Computer Simulation Conference (SCSC'03)*, Montreal, Canada, 2003, pp. 3-10.

[22]. E. Ostrosi, A.-J. Fougères, M. Ferney, et al., A fuzzy configuration multi-agent approach for product family modelling in conceptual design, *Journal of Intelligent Manufacturing*, Vol. 23, Issue 6, 2012, pp. 2565-2586.

[23]. Y. Brun, G. D. M. Serugendo, et al., Engineering self-adaptive systems through feedback loops, in Software Engineering for Self-Adaptive Systems, *Springer*, Berlin, Heidelberg, 2009, pp. 48-70.

[24]. J. Grosset, A.-J. Fougères, M. Djoko-Kouam, J.-M. Bonnin, Multi-agent simulation of autonomous industrial vehicle fleets: towards dynamic task allocation in V2X cooperation mode, *Integrated Computer-Aided Engineering*, 2024, pp. 1-18, Pre-press.

[25]. H. Lasi, P. Fettke, H. G. Kemper, et al., Industry 4.0, *Business & Information Systems Engineering*, Vol. 6, Issue 4, 2014, pp. 239-242.

(032)

Efficient Graph Embedding and Semantic Relationship Reconstruction in the WordNet Lexical Database

Ailin Song [1,2], Mingkun Xu [1,3] and Shuai Zhong [1]
[1] Guangdong Institute of Intelligence Science and Technology, Zhuhai, China
[2] University of Electronic Science and Technology of China, Chengdu, China
[3] Center for Brain-Inspired Computing Research (CBICR), Tsinghua University, Beijing, China
E-mails: {xumingkun, zhongshuai}@gdiist.cn

Summary: WordNet, an expansive English lexical database, intricately structures nouns, verbs, adjectives, and adverbs into synsets interconnected by semantic and part-of-speech relationships. As a fundamental resource in natural language processing (NLP), the dataset necessitates transformation into computationally manageable vectors, commonly referred to as word embeddings. While Graph Convolutional Networks (GCN) demonstrate proficiency in graph embedding, Graph Autoencoders (GAE) assumes a central role in unsupervised graph learning. This study explores the integration of NLP and GNN methodologies to efficiently embed graphs and reconstruct semantic relationships within the WordNet lexical database. Specifically, to address the dataset's inherent complexity, we focus on the WN18RR subset and graph partitioning strategy to efficiently embed subgraphs by employing a multi-graph training approach. Furthermore, we introduce a delicately designed dual-decoding mechanism to enhance the model's capability in reconstructing node features. This work offers a novel solution for leveraging the WordNet dataset in GAE, paving the way for more effective graph embeddings in future research.

Keywords: WordNet, Graph autoencoder, Graph convolutional networks, Subgraph embedding, Semantic relationship reconstruction.

1. Introduction

WorldNet, a comprehensive English lexical database [1], organizes nouns, verbs, adjectives, and adverbs into sets of synonyms. Interconnections between these synonym sets are established through semantic and part-of-speech relationships. Frequently utilized in natural language processing, WorldNet requires a conversion of lexical entries into vector forms, known as word embeddings, to enhance computer comprehension of textual information. Classic word embedding models, such as the bag-of-words model and distributed representations are commonly employed. It is noteworthy that a significant similarity exists between natural language datasets and graph data. Words can be considered as nodes, with relationships forming edges. Similar to natural language data, graph data cannot be directly input into computer models and requires reliance on graph embedding techniques. These embedding algorithms draw inspiration from the field of natural language processing, exemplified by techniques like DeepWalk and node2vec, both rooted in the core ideas of word2vec.

Compared to textual data, graph-structured data is more complex, exhibiting diverse and unstable structural variations with tight relationships among various data nodes. A change in a single node may potentially affect the entire graph structure. Graph representation methods based on random walk strategies primarily focus on the structural features of graph networks, which often neglecting the attribute features of nodes and textual information. These elements play a pivotal role in graph networks, especially in heterogeneous graphs. In recent years, GCN has excelled in graph embedding algorithms. It adeptly integrates the attribute information of nodes and their adjacent neighbors. Knowledge graphs are closely related to graph data and have been widely applied with outstanding results in natural language processing. Researchers often focus on issues such as word embedding representation, link prediction, and knowledge graph completion. We aim to strengthen the relation between natural language processing and graph learning, and explore the potential and possibilities of graph deep learning and natural language processing from another perspective.

Our study uniquely applies the WordNet dataset to GAE, achieving graph data reconstruction. GAE [2] incorporates an encoder-decoder mechanism for graph data reconstruction. The encoder effectively compresses complex data into low-dimensional vector representations, which serves as the latent information output of the model. By distilling crucial information while filtering out irrelevant details, it efficiently preserves the informational essence of the data. Subsequently, the extracted shallow variables are inputted into the decoder, which leverages the classical dot-product decoding technique to derive the reconstructed output of the graph. However, there are two major issues associated with reconstructing the WordNet dataset through GAE. First, the original WordNet dataset is extensive and intricate, cannot directly represent relationships between word nodes, rendering it impractical for direct input into GAE model. To address this challenge, we select WN18RR the subset of Wordnet as the original dataset. It contains 40943 entities and 11 relationships, represented in triple form (head entity, relationship, tail entity), facilitating efficient graph construction.

Given the distinctive nature of WordNet, which structures word relationships through synonym sets, it is inherent that the comprehensive dataset comprises interconnected or discrete subgraphs. Despite its reduced size, WN18RR remains complex, with multiple subgraphs. Consequently, we fragment the dataset into subgraph tasks and employ multi-graph training for GAE, which achieves embeddings for subgraphs, and ultimately representing the entire dataset. This strategy enhances the model training efficiency and aids in capturing both local and global information within the graph structure. Second, we observe that the original GAE model focuses solely on graph structure reconstruction, neglecting node feature reconstruction. The feature vectors of nodes serve as a pivotal element in graphs, encapsulating rich information pertaining to both the individual node and its intricate relationships with other nodes. By reinstating the original dimensionality of these feature vectors, we enable a more thorough reconstruction of the graph's structural and semantic nuances. We introduce a dual decoding mechanism, utilizing GCN decoding to achieve node feature reconstruction.

In conclusion, GAE has demonstrated exceptional performance in graph data reconstruction, with a concise model design and strong plasticity. However, there are currently few studies that integrate WordNet data into GAE to achieve knowledge graph reconstruction. This paper proposes a reliable method that successfully combines WordNet data with GAE, yielding satisfactory experimental results. This research not only broadens the application scope of GAE but also provides new ideas and methods for knowledge graph reconstruction. We propose a multi-graph training strategy by selecting appropriate subsets, splitting subgraph tasks, and employing multi-graph training, our study provides a robust solution for the application of the original WordNet dataset in the GAE model. The introduction of a dual decoding strategy further lays the groundwork for efficient graph embedding representation.

2. Related Works

WordNet holds extensive application value in the field of natural language processing, particularly playing a crucial role in knowledge graph representation learning, link prediction, and knowledge graph completion. Among early traditional methods, algorithms based on translational distances such as TransE [3] and TransR [4], as well as tensor decomposition models like DisMult [5] and ComplEx [6], treat the knowledge graph as a three-dimensional adjacency matrix, effectively capturing the complex interactions between entities and relations. With the advancement of deep learning techniques, the integration of neural networks with knowledge graphs has also made significant progress. For instance, models like ConvE [7] and ConKB [8] have further improved the performance of knowledge graph representation learning by introducing deep learning methods such as convolutional neural networks. In these studies, WordNet, as an abundant and structured vocabulary resource, provides indispensable dataset support for the construction and representation learning of knowledge graphs.

In the task of graph reconstruction, obtaining the embedded representation of graph data is a crucial step. This paper employs GCN to reconstruct the WordNet dataset, and the effectiveness of WordNet in GCN has received widespread attention. Schlichtkrull [9] et al. pioneered the utilization of the GCN framework to construct relational networks for knowledge graphs, namely the R-GCN model. In the encoding process, R-GCN constructs a corresponding relational transformation matrix for each relation to perform transformation operations on neighbor entity nodes connected by the relation, enabling better modeling of relational information in the knowledge graph. The WordGCN method constructs a word graph and utilizes graph convolutional networks to perform convolutional operations on the word graph. By aggregating information from neighboring nodes, WordGCN can update the representation of each node, effectively capturing contextual relationships among words. Additionally, the SemGCN [11] proposed by Shikhar Vashishthet al. not only considers the co-occurrence relationships between words but also introduces rich semantic relationships such as hyponymy and synonymy to further enrich the representation of word embeddings. This approach can capture semantic connections among words more comprehensively, improving the quality of word embeddings.

In the realm of graph data reconstruction, GAE has demonstrated extensive application potential. For instance, Shirui Pan [12] and their team proposed a unique adversarial graph embedding framework for graph data. This framework ingeniously encodes the topological structure and node content of the graph into a compact representation, enabling the training of a decoder to accurately reconstruct the graph structure. Furthermore, Zhenyu Hou [13] et al. introduced the GraphMAE model, which applies a masking process to a portion of the nodes in the graph during training. Subsequently, the masked graph is fed into an encoder to obtain the embedded representation of the nodes. Additionally, Hongyuan Zhang [14] et al. proposed a theoretical analysis method based on graph autoencoders and relaxed k-means, known as the EGAE model. This model effectively learns the embedded representation of graphs by designing specific encoder and decoder structures, providing novel insights for the analysis and processing of graph data. Moreover, there is the VGAE [2] model, which is based on variational encoders. This model integrates variational Bayesian methods with autoencoders, further enriching the technical methods for graph data reconstruction. GAE is essentially the application of AE to graph data, a technique that has also gained significant traction in the natural language processing. For instance, Chao Wei et al. introduced a manifold-regularized approach, specifically Short Text

Embedding Autoencoders (STE-AEs) [15]. This method aims to incorporate semantic information from neighboring texts into the regularization training of Autoencoders (AEs), enabling the extraction of discriminative low-dimensional embeddings for short texts. By integrating semantic manifolds within the AE framework, this approach enhances the representational power of the model, effectively capturing subtle yet crucial textual nuances.

Furthermore, there exist numerous other models in the field of graph reconstruction, such as GraphRNN. This model ingeniously transforms the problem of graph generation into a sequence generation problem, enabling effective modeling of graph structures. Additionally, Deepak Nathani [16] and their team proposed an attention-based feature embedding method that can accurately capture the entity and relationship features within any given entity neighborhood, providing robust support for the deep analysis and processing of graph data.

3. Methods

This approach extends the GAE model by introducing a node feature reconstruction network and incorporating the WN18RR dataset. The objective is to achieve the reconstruction of both word-graph relationships and word-node features, involving two primary aspects:

Data Preprocessing

In prior research, datasets like Cora were utilized for training GAE models [5], stored in the form of adjacency matrices accompanied by corresponding feature vectors. However, the WN18RR dataset is text-based, organized in triplets (head entity, relationship, tail entity), lacking complete feature vector files compatible with GAE. Hence, preprocessing of the WN18RR dataset is essential to construct the adjacency matrix and feature vectors. Given the immense size of the WN18RR dataset and the relationships among synonym sets are fragmented and complex, it is impractical to adopt the straightforward methods employed in the handling of datasets like Cora. Typically, researchers unified the training, validation, and test sets into a single graph, randomly partitioned the valid edges, and reconstructed the complete graph. However, processing the WN18RR dataset presents two challenges: (1) Data Structure Issue: As a collection of synonyms, WN18RR exhibits graph relationships predominantly in the form of independent subgraphs, owing to its complex and extensive nature. A comprehensive graph training approach is unsuitable. (2) Feature Vector Construction Issue: The original WN18RR dataset lacks feature vectors, necessitating their manual construction.

To address these challenges, we propose a multi-graph training strategy, splitting the dataset into independent subgraphs facilitating the reconstruction of individual subgraphs and, consequently, the entire dataset relationship (as illustrated in Fig. 1). We meticulously organize the subgraph list data in a triplet format, systematically extracting triplet relationship head nodes from the comprehensive dataset. Utilizing each selected node as the focal point, we comprehensively identify all its first-order relationships, thereby constructing distinct subgraphs. Subsequently, entities and edges in each subgraph are converted into index form respectively to build the adjacency matrix. For feature vector construction, considering the graph convolution model's ability to learn graph features, we adopt a random initialization approach, constructing feature vectors following a 0-1 normal distribution. Notably, the dimensionality of feature vectors can be adjusted according to requirements, significantly impacting the model's training effectiveness. Properly balancing the dimensionality enhances the model's expressive power and generalization capabilities.

This thorough data preprocessing lays the foundation for effective model training on the WN18RR dataset, addressing its unique challenges and paving the way for advanced graph-based semantic analysis.

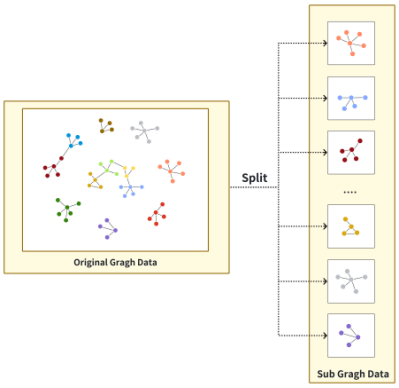


Fig. 1. Graph data preprocessing process. Splits a complete single graph into individual subgraphs.

Reconstruction Modeling

Based on the GAE model, we introduce a node feature vector reconstruction network as shown in Fig. 2. GAE is primarily encompassing encoding and decoding stages for graph reconstruction tasks. During the encoding phase, we leverage a two-layer GCN with sequential convolutional operations to meticulously capture the intricate relationships among nodes within the graph. Subsequently, these relationships are transformed into low-dimensional representations of the nodes, yielding the latent variables Z as output. This representation framework efficiently encodes the inherent structural and semantic information of the graph, serving as a crucial input for the subsequent decoding phase. The mathematical representation of the model is shown in Equation (1), where $\overline{A}$ represents the normalized adjacency matrix as shown in equation 2, D is the diagonal matrix with values corresponding to the sum of each row of the adjacency matrix, where $\tilde{A} = A + I$.

$$Z = f(\overline{A}, X) = \overline{A} f_{ReLu}(\overline{A} X W^{0}_{a\times b}) W^{1}_{b\times c}, \quad (1)$$

$$\overline{A} = D^{-\frac{1}{2}} \tilde{A} D^{-\frac{1}{2}} \quad (2)$$

Graph reconstruction decoding obtains the output of the graph reconstruction result by doing the dot product of the inverse of the hidden variable with itself as shown in equation (3). In the context of graph embeddings, for any two connected nodes, we strive to achieve a sufficiently large dot product between their respective embedded vectors, thereby enabling this dot product to effectively represent a "1" in the corresponding adjacency matrix entry. Conversely, for any two disconnected nodes, we aim for a dot product that is sufficiently small, allowing it to accurately represent a "0" in the adjacency matrix entry. This approach ensures that the graph embeddings effectively capture the connectivity patterns within the graph.

$$A' = Z \cdot Z^{T} \quad (3)$$

3. Result

The results of multi-graph training, as shown in Fig. 3, we construct node feature vectors with dimensions of 50 and 100, respectively, and the training outcomes vary under diverse parameter settings. We observed that the performance metric of loss value and accuracy improve with an increase in feature dimension. Simultaneously, on the basis of selecting a feature dimension of 100, we added a SoftMax layer to the encoder. Although this modification did not surpass the original model in terms of loss value performance, it exhibited optimal performance in terms of training accuracy and testing performance. To further investigate the impact of the SoftMax layer, we compared the performance of the test set when adding a SoftMax layer and when not adding a SoftMax layer under the condition of a feature dimension of 100. The results showed that the model with the SoftMax layer exhibited more stable performance, achieving a maximum accuracy of 80.7 % on the test set. Fig. 4 demonstrates an example of the reconstructed subgraph, where the number of completely generated subgraphs in the test set exceeded 63.3 %. This result firmly establishes that the multi-subgraph training mode using the WordNet dataset employed in this paper exhibits excellent graph reconstruction effects on the simplified GAE model.

Feature vector reconstruction decoding restores the dimension of the original feature vector by the inverse operation of the encoded part of the GCN as shown in equation (4).

$$X' = f(\overline{A}, Z) = \overline{A} f_{ReLu}(\overline{A} Z W^{0}_{c\times b}) W^{1}_{b\times a} \quad (4)$$

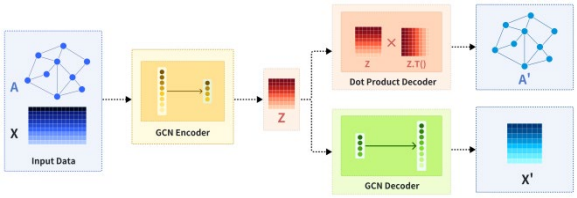


Fig. 2. Dual decoding GAE model. The model takes the adjacency matrix and feature vectors of the graph data as inputs and, after a GCN encoder, outputs the potential representation of the nodes Z. The dual decoder is divided into graph structure decoding (dot product decoder) and feature vector decoding (GCN decoder). Finally, model outputs the reconstructed adjacency matrix and feature vectors.

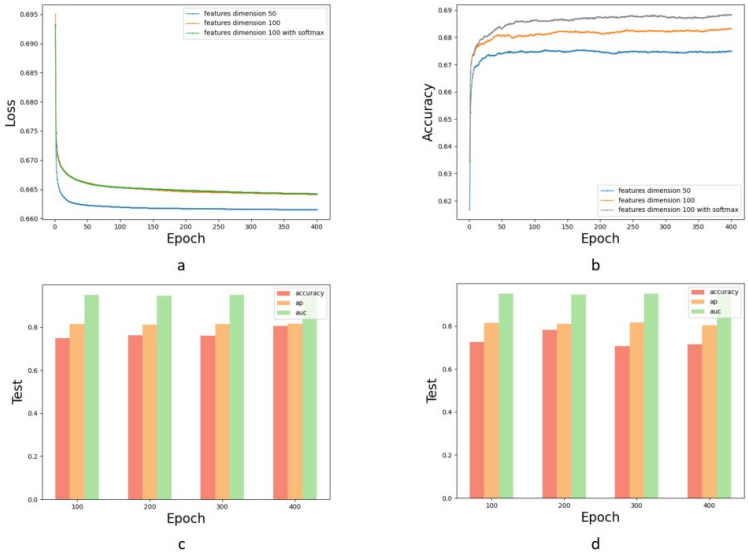


Fig. 3. The multi-plot training results. a. The changes of loss value during the training process. b. The accuracy during the training process. c. When the feature dimension of the data is set to 100, the subsequent evaluation of the test set reveals specific values for test accuracy, average precision (AP), and area under the curve (AUC). d. When the dataset's feature dimension is set to 100, and a SoftMax layer is incorporated into the model, the subsequent evaluation of the test set reveals specific values for test accuracy, AP, and AUC.

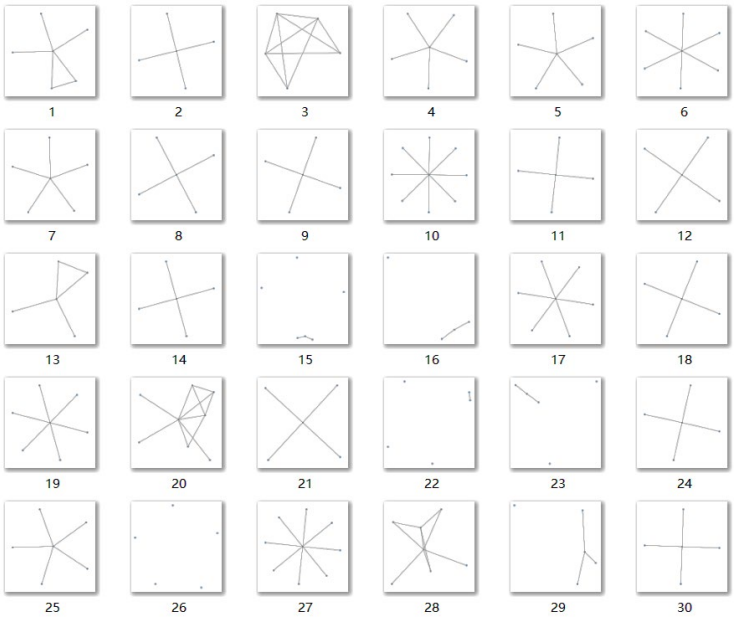


Fig. 4. Subgraph reconstruction case. The reconstructed subgraphs presented by this model are exhibited, with each subgraph prototype adopting a star-shaped form.

4. Conclusion

This study introduces, for the first time, the application of the WN18RR dataset in graph autoencoders for graph reconstruction. We propose a method to split the dataset into subgraphs for multi-graph training reconstruction, concurrently addressing node feature reconstruction and graph reconstruction. This approach generally yields satisfactory reconstruction structures. The multi-graph training mode proves to be effective in enhancing model training efficiency, demonstrating superior reconstruction results compared to whole-graph training. Extensive evidence suggests that the multi-graph training mode significantly enhances the efficiency of model training. In contrast to full-graph training, this approach not only adeptly captures the intricate structural and feature information embedded within large-scale knowledge graphs, but also boosts training efficiency and yields superior reconstruction outcomes. In future work, we plan to incorporate GCN in the data processing phase. By leveraging GCN to learn the effective representations of feature vectors. This enhancement contributes to a better understanding of the intricate structure and relationships within the original graph.

Acknowledgements

This work was supported by the Key-Area Research and Development Program of Guangdong Province (Grants No. 2021B0909060002) and, National Natural Science Foundation of China (Grants No.62204140).

References

[1]. C. Fellbaum, WordNet: An Electronic Lexical Database, *MIT Press*, 1998.

[2]. T. N. Kipf, M. Welling, Variational graph auto-encoders, *arXiv Preprint*, 016, arXiv:1611.07308.

[3]. A. Bordes, N. Usunier, A. Garcia-Duran, et al., Translating embeddings for modeling multi-relational data, in Advances in Neural Information Processing Systems, Vol. 26, *Curran Associates, Inc*., 2013.

[4]. Y. Lin, Z. Liu, M. Sun, et al., Learning entity and relation embeddings for knowledge graph completion, in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 29, 2015.

[5]. B. Yang, W.-t. Yih, X. He, et al., Embedding entities and relations for learning and inference in knowledge bases, *arXiv Preprint*, 2014, arXiv:1412.6575.

[6]. T. Trouillon, J. Welbl, S. Riedel, et al., Complex embeddings for simple link prediction, in *Proceedings of Machine Learning Research (PMLR'16)*, 2016, pp. 2071-2080.

[7]. T. Dettmers, P. Minervini, P. Stenetorp, et al., Convolutional 2D knowledge graph embeddings, in *Proceedings of the AAAI Conference on Artificial Intelligence,* Vol. 32, 2018.

[8]. D. Q. Nguyen, T. D. Nguyen, et al., A novel embedding model for knowledge base completion based on convolutional neural network, *arXiv Preprint*, 2017, arXiv:1712.02121.

[9]. M. Schlichtkrull, T. N. Kipf, P. Bloem, et al., Modeling relational data with graph convolutional networks, in *Proceedings of the 15th International Conference Extended Semantic Web Conference (ESWC'18)*, Heraklion (Crete), Greece, June 3-7, 2018, pp. 593-607.

[10]. S. Vashishth, P. Yadav, M. Bhandari, et al., Graph convolutional networks based word embeddings, *arXiv Preprint*, 2018, arXiv:1809.04283.

[11]. S. Vashishth, M. Bhandari, P. Yadav, et al., Incorporating syntactic and semantic information in word embeddings using graph convolutional networks, *arXiv Preprint*, 2018, arXiv:1809.04283.

[12]. S. Pan, R. Hu, G. Long, et al., Adversarially regularized graph autoencoder for graph embedding, *arXiv Preprint*, 2018, arXiv:1802.04407.

[13]. Z. Hou, X. Liu, Y. Cen, et al.. Graphmae: Self-supervised masked graph autoencoders, in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 594-604.

[14]. H. Zhang, P. Li, R. Zhang, et al., Embedding graph auto-encoder for graph clustering, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 34, Issue 11, 2023, pp. 9352-9362.

[15]. C. Wei, L. Zhu, J. Shi, Short text embedding autoencoders with attention-based neighborhood preservation, *IEEE Access*, Vol. 8, 2022, pp. 223156-223171.

[16]. D. Nathan, J. Chauhan, C. Sharma, et al., Learning attention-based embeddings for relation prediction in knowledge graphs, *arXiv Preprint*, 2019, arXiv:1906.01195.

(033)

Generation of Synthetic EEG Signals for Testing Dynamic Brain Connectivity Estimation Methods

Z. Šverko [1], S. Vlahinić [1], N. Stojković [1] and P. Rogelj [2]
[1] University of Rijeka, Faculty of Engineering, Department of Automation and Electronics,
Vukovarska 58, 51000 Rijeka, Croatia
[2] University of Primorska, Faculty of Mathematics, Natural Sciences and Information Technologies,
Glagoljaška 8, 6000 Koper, Slovenia
Tel.: +385 51 505720
E-mails: zoran.sverko@riteh.uniri.hr, sasa.vlahinic@riteh.uniri.hr, nino.stojkovic@riteh.uniri.hr,
peter.rogelj@upr.si

Summary: In this study, a method for generating synthetic signals of electroencephalographic (EEG) brain activity is presented. The generated signals are suitable for analysis and the development of methods for estimating direct functional connectivity such as Granger causality (GC) from the perspective of unveiling their dynamic capabilities. To make the synthetic signals as realistic as possible, they are generated from real EEG signals and based on a regression model obtained from them. A signal pair with a known causality relationship is generated from two input signals with high directional functional connectivity estimated by GC. Their autoregressive model is used to mix their unrelated parts according to the desired dynamic functional connectivity with values of GC in a range from 0 to the GC of source signals. As the source of generated signals are real signals and their model of mutual dependences, the generated signals have all realistic properties, including the realistic signal dependencies such as delays, and frequency relationships.

Keywords: Electroencephalography, Granger causality, Synthetic signals, Dynamic connectivity.

1. Introduction

The human brain comprises neurons connected by synapses. These neurons are structured across various spatial regions and engaged in functional interactions across diverse time frames [1]. In practice, there are several methods of measuring brain activity, and in this work, we used signals recorded by the electroencephalography (*EEG*) method, which provides us with good temporal resolution [2]. Brain connectivity analysis encompasses two main categories: structural and functional. Structural connectivity [3] analysis involves tracking the direction of fibers among brain regions (suitable measurement methods are magnetic resonance imaging *MRI* or diffusion tensor imaging *DTI*). Functional connectivity analysis examines the information exchange between brain regions or within a single region and can be categorized into undirected [4-8] (gauging the level of connectivity) and directed (assessing strength and direction) measures. Our focus in this study lies on directed connectivity measures. Several methods such as Granger Causality analysis (*GC*) [9-13], Phase Slope Index (*PSI*) [14, 15], Transfer Entropy (*TE*) [16], Partial Directed Coherence (*PDC*) [17], and others are used for directed connectivity analysis of *EEG* data. The most commonly used method is *GC*, and it is used for static and dynamic directed functional connectivity analysis. For the dynamic analysis, *GC* computes the connectivity based on autoregressive models using a temporal window. The size of this window limits the capabilities of describing connectivity dynamics. Assessment of the dynamic capabilities is especially difficult because of the lack of the ground truth for real signals. To enable evaluation of connectivity methods for assessing dynamic connectivity, the capability of measuring changes of connectivity in time, we propose a method for generating synthetic signals.

Upon reviewing the existing literature where researchers employ *GC*, we did not find a way to compute synthetic signals with realistic properties and predefined the connectivity changes in time, which would enable us to test existing and develop improved methods for dynamic analysis of functional connectivity. Building upon the aforementioned, this paper presents the method of generating synthetic signals.

2. Methods

In this section, we review the *GC* method, and demonstrate the proposed procedure for generating realistic synthetic signals suitable for testing the dynamics of connectivity estimation methods.

2.1. Granger Causality

Granger causality (GC) is a statistical technique used to assess the causal relationship between two time series. It operates on the principle that if a variable, denoted as S (source or Granger-causes), influences another variable T (target), then past values of S should contain information that assists in predicting future

values of *T*, beyond what is already predicted by past values of *T* alone. Briefly, Granger causality prediction determines whether past values of one variable enhance the prediction of another variable.

When calculating the prediction of the next sample in an observed variable using only the past values from that variable, a univariate autoregressive model is used. Univariate autoregressive model is defined as:

$$T(n) = \omega_1 T(n-1) + \omega_2 T(n-2) + \cdots + \omega_M T(n-M) + e_T(n), \quad (1)$$

which can be shortened as:

$$T(n) = \sum_{i=1}^{M} \omega_{Ti} T(n-i) + e_T(n), \quad (2)$$

where ω_{Ti} are the coefficients of the autoregressive model for *T(n)*, *M* is the order of regression, and e_T is autoregression error.

Bivariate autoregressive models extend this concept to analyze the causal relationship between two variables and are defined as:

$$\begin{aligned} T(n) = {} & \omega_{T1} T(n-1) + \omega_{T2} T(n-2) + \cdots \\ & + \omega_{TM} T(n-M) + \omega_{S1} S(n-1) + \\ & + \omega_{S2} S(n-2) + \cdots + \omega_{SM} S(n-M) + \\ & + e_{TS}(n), \end{aligned} \quad (3)$$

which can be written as:

$$T(n) = \sum_{i=1}^{M} \omega_{Ti} T(n-i) + \sum_{i=1}^{M} \omega_{Si} S(n-i) + e_{TS}(n), \quad (4)$$

where e_{TS} is multivariate regression error. Model coefficients ω are calculated by minimizing errors for provided signals of *N* samples, where $N > M$. *GC* is then defined as:

$$GC = \log \frac{Var[e_T(n)]}{Var[e_{TS}(n)]} \quad (5)$$

In directed functional connectivity analysis, Granger causality prediction serves as a powerful tool for inferring directional influences between different brain regions based on their time-series data. By applying the principles of Granger causality, it can be assessed whether the past activity of one (source) brain region contains predictive information about the future activity of the other (target) region, beyond what can be predicted by the past activity of that brain region alone. Through the application of Granger causality prediction, insights into static the dynamic interactions and causal relationships within complex systems can be gained, aiding in the understanding of predictive relationships and decision-making processes.

2.2. Generation of Synthetic Signals

Knowing the regression models, we can use them for mixing real signals into synthetic ones. Here, we can control the contribution of source signals by introducing an additional mixing parameter, which can arbitrarily change in time and, thus define the of connectivity dynamics.

First, we have to define a method for generating signals with the maximal connectivity equal to the connectivity of source signals, from independent source signals. Let us take two real source signals *T* and *S* and build their regression models to estimate their univariate and bivariate regression errors e_T and e_{TS}, as well as their connectivity GC_{TS}. We can then take some subsections of signals *T* and *S*, with the same number of samples *N′* but at different time. Let us name them *T′* and *S′*. They are expected to be unrelated and have low *GC*. Using these sequences, we can generate a new sequence R_1, which would resemble the signal *T′* but also be related to *S′*, using the regression model:

$$R_1(n) = \sum_{i=1}^{M} \omega_{Ti} T'(n-i) + \sum_{i=1}^{M} \omega_{Si} S'(n-i) + e'_{TS}(n) \quad (6)$$

The inclusion of the regression error e'_{TS} from the same subsection as signal *T'* makes the generated sequence realistic from the perspective of predictability. The obtained signal R_1 is expected to have $GC = GC_{R1S}$ in relation to *S* similar to GS_{TS}.

The limitation of this generative model is in inability to dynamically control the true connectivity. For getting a signal unrelated to *S*, one could use a univariate regression model, however, the obtained result equals the target signal *T'*:

$$R_2(n) = \sum_{i=1}^{M} \omega_i T'(n-i) + e'_T(n) = T'(n). \quad (7)$$

Having the realistic signals that resemble high and low connectivity, i.e., R_1 and R_2, we can dynamically control the connectivity by mixing them:

$$\begin{aligned} R'(n) = {} & K(n) \cdot R_1(n) + \\ & + (1 - K(n)) \cdot R_2(n) = \\ & = (1 - K(n)) \cdot T'(n) + \\ & + K(n) \cdot (\sum_{i=1}^{M} \omega_{Ti} T'(n-i) + \\ & + \sum_{i=1}^{M} \omega_{Si} S'(n-i) + e'_{TS}(n)) \end{aligned} \quad (8)$$

Here, *K(n)* is a temporal connectivity parameter, defining the connectivity between *R'* and *S* such that for *K = 0* should ideally lead to *GC = 0*, while *K = 1* to $GC \approx GC_{TS}$.

As such, the pair of signals {*R', S'*} should enable us to test methods of dynamic connectivity in further research.

3. Results

We demonstrate the proposed signal generation method on an EEG Motor Movement/Imagery Dataset [18]. We used the S001R01 recording of baseline with eyes open. The data was recorded at a sample rate of 160 Hz with a duration of 61 seconds. The *GC* connectivity matrix was calculated (Fig. 1). High *GC* connectivity of 0.92 was observed between electrodes *FC1* (*T*) and *F1* (*S*). Model coefficients $\omega_{(.)}$ (of bivariate regressive model of order 19) which are needed for signal reconstruction, were calculated. Subsections of signals *T* and *S,* that is, time intervals recorded by electrode *FC1* from 0 to 14.4 seconds (*T'*) and by electrode *F1* from 28.8 to 43.2 seconds (*S'*) were selected. The *GC* of these unrelated parts of signals is 0.008.

The temporal connectivity parameter *K* is defined as 1 in the first interval (0 to 4.8 seconds) and in the third interval (9.61 to 14.4 seconds), while set to 0 in the second interval (4.81 to 9.6 seconds). The synthetic signal was generated using the proposed method (8) and order 19. The resulting synthetic signal *R*' and the source signal (*S*') were used to perform Functional Connectivity analysis using *GC* (Fig. 2). For the three defined intervals, the obtained *GC* values were 0.77, 0.032 and 0.77 respectively. The *GC* value for the second interval is low as expected, while values for the 1st and 3rd interval are high. However, the *GC* for these intervals is still lower than the reference value that equals the GC of the original signals.

To illustrate the usability of the signal generation method for analysis of dynamic connectivity estimation methods, we applied the dynamic *GC* [19-21] using sliding window. Here G*C* is estimated for the generated signal that resembles the modified *FC1* signal with respect to the source signal *F1*. Fig. 3a and 3b shows the results obtained using a window size of 2 seconds and 400 ms, the order is 19.

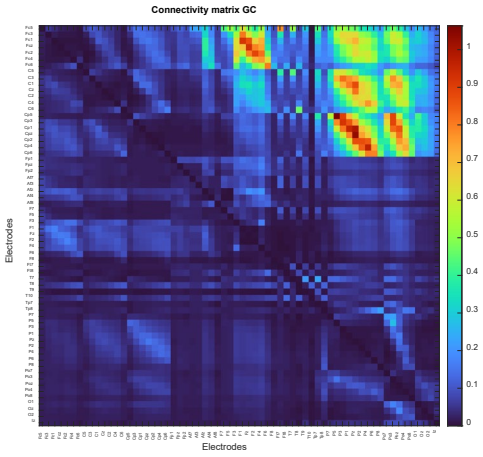


Fig. 1. Connectivity matrix GC for subject S001R01 [18], the baseline eyes open experiment run for order 19.

We can see that the estimated connectivity changes in time, where, as expected, estimated values deviate from the true connectivity, mostly at the transitions. When using lower window size, the reliability of estimated models as part of computing GC is low, leading to high volatility and low reliability. This shows the limited ability of the GC measure to find the time of connectivity changes and the need for further research in this direction.

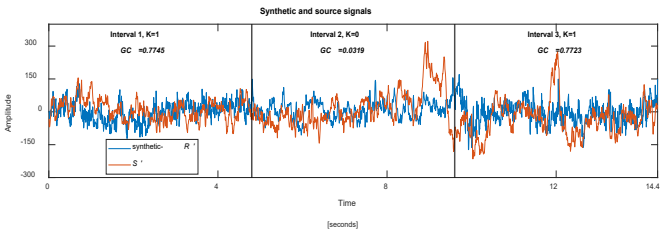


Fig. 2. Synthetically generated signal and true GC values (in intervals). The model order used was 19.

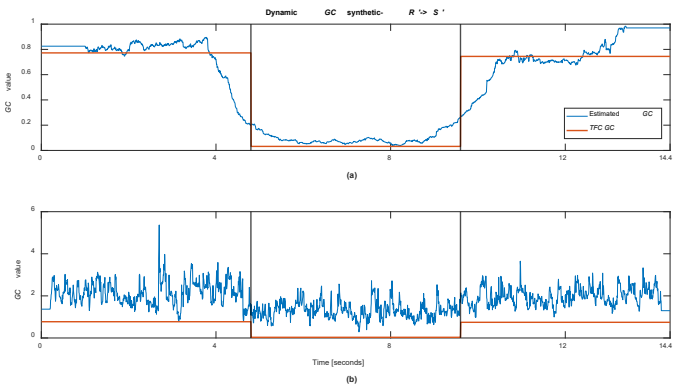


Fig. 3. Dynamic GC values estimated using sliding window analysis and a window size of 2 seconds (a) and 400 ms (b).

4. Conclusions

This work presents the procedure for generating a synthetic signal suitable for analysis of dynamic properties of directional connectivity methods. The signal is generated by reconstruction from real signals based on regression model coefficients. Using the proposed method, we have the capability to manage the influence of source signals by integrating an additional temporal connectivity parameter *K*, which can dynamically vary in time, thus determining the dynamics of true connectivity. This method assures that the generated synthetic signals are highly realistic. When testing the dynamic connectivity methods, the connectivity shall be observed between the generated signal *R*' and the source signal *S*'. Results shall be compared with the weighting signal *K*. We have demonstrated this for dynamic GC estimation using a sliding window. Here the problem is in the selection of the window size. Large windows are expected to result in poor temporal resolution of estimated dynamic connectivity, while the use of narrow windows could harm the accuracy of auto-regressive models leading to high variability in estimated dynamic connectivity. Finding the optimal window size or comparing different methods requires the knowledge of true connectivity changes, which are only known when the signals are generated synthetically. Here it is important that generated signals have realistic properties which are obtained from real signals. Our proposed method fulfills both requirements. Therefore, it can be concluded that the proposed method for generating synthetic test signals is suitable for testing and development of dynamic connectivity methods.

Acknowledgements

This work has been supported by the University of Rijeka under the project number UNIRI-ISKUSNI-TEHNIC-23-31.

Z. Šverko would like to thank the ERASMUS+ organization for the mobility scholarship, number of project: 2023-1-HR01-KA131-HED-000113440, during which this work was created.

References

[1]. A. Fornito, A. Zalesky, E. Bullmore, Fundamentals of Brain Network Analysis, 1st Ed., *Academic Press*, London, 2016.

[2]. S. Sanei, J. A. Chambers, EEG Signal Processing, *John Wiley & Sons Ltd*, 2007.

[3]. M. A. Koch, D. G. Norris, M. Hund-Georgiadis, An investigation of functional and anatomical connectivity using magnetic resonance imaging, *Neuroimage*, Vol. 16, Issue 1, 2002, pp. 241-250.

[4]. C. J. Stam, G. Nolte, A. Daffertshofer, Phase lag index: Assessment of functional connectivity from multi channel EEG and MEG with diminished bias from common sources, *Hum. Brain Mapp.*, Vol. 28, Issue 11, 2007, pp. 1178-1193.

[5]. J.-P. Lachaux, E. Rodriguez, J. Martinerie, F. J. Varela, Measuring phase synchrony in brain signals, *Hum. Brain Mapp.*, Vol. 8, Issue 4, 1999, pp. 194-208.

[6]. M. Vinck, R. Oostenveld, M. Van Wingerden, F. Battaglia, C. M. Pennartz, An improved index of phase-synchronization for electrophysiological data in the presence of volume-conduction, noise and sample-size bias, *Neuroimage*, Vol. 55, Issue 4, 2011, pp. 1548-1565.

[7]. Z. Šverko, M. Vrankić, S. Vlahinić, P. Rogelj, Complex Pearson correlation coefficient for EEG connectivity analysis, *MDPI Sensors*, Vol. 22, Issue 4, 2022, 1477.

[8]. Z. Šverko, M. Vrankić, S. Vlahinić, P. Rogelj, Dynamic connectivity analysis using adaptive window size, *MDPI Sensors*, Vol. 22, Issue 14, 2022, 5162.

[9]. K. Friston, R. Moran, A. K Seth, Analysing connectivity with Granger causality and dynamic causal modelling, *Curr. Opin. Neurobiol.*, Vol. 23, Issue 2, 2013, pp. 172-178.

[10]. T. Uchida, K. Fujiwara, T. Inoue, Y. Maruta, M. Kano, M. Suzuki, Analysis of VNS effect on EEG connectivity with granger causality and graph theory, in *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC'18)*, 2018, pp. 861-864.

[11]. A. K. Seth, A. B. Barrett, L. Barnett, Granger causality analysis in neuroscience and neuroimaging, *J. Neurosci.*, Vol. 35, Issue 8, 2015, pp. 3293-3297.

[12]. S. K. Loo, A. Cho, T. S. Hale, J. McGough, J. McCracken, S. L. Smalley, Characterization of the theta to beta ratio in ADHD: identifying potential sources of heterogeneity, *J. Atten. Disord.*, Vol. 17, Issue 5, 2013, pp. 384-392.

[13]. V. Youssofzadeh, G. Prasad, M. Naeem, K. Wong-Lin, Temporal information of directed causal connectivity in multi-trial ERP data using partial Granger causality, *Neuroinformatics*, Vol. 14, 2016, pp. 99-120.

[14]. A. Basti, V. Pizzella, F. Chella, G. L. Romani, G. Nolte, L. Marzetti, Disclosing large-scale directed functional connections in MEG with the multivariate phase slope index, *Neuroimage*, Vol. 175, 2018, pp. 161-175.

[15]. A. Al-Ezzi, N. Yahya, N. Kamel, I. Faye, K. Alsaih, E. Gunaseli, Social Anxiety Disorder Evaluation using Effective Connectivity Measures: EEG Phase Slope Index Study, in *Proceedings of the Conference on Biomedical Engineering and Sciences (IECBES'20)*, 2020, pp. 120-125.

[16]. M. H. I. Shovon, N. Nandagopal, R. Vijayalakshmi, J. T. Du, B. Cocks, Directed connectivity analysis of functional brain networks during cognitive activity using transfer entropy, *Neural Process. Lett.*, Vol. 45, 2016, pp. 807-824.

[17]. G. Varotto, E. Visani, L. Canafoglia, S. Franceschetti, G. Avanzini, F. Panzica, Enhanced frontocentral EEG connectivity in photosensitive generalized epilepsies: A partial directed coherence study, *Epilepsia*, Vol. 53, Issue 2, 2012, pp. 359-367.

[18]. G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, J. R. Wolpaw, BCI2000: a general-purpose brain-computer interface (BCI) system, *IEEE Trans. Biomed. Eng.*, Vol. 51, Issue 6, 2004, pp. 1034-1043.

[19]. M. Winterhalder, *et al.*, Comparison of linear signal processing techniques to infer directed interactions in

multivariate neural systems, *Signal Processing*, Vol. 85, Issue 11, 2005, pp. 2137-2160.

[20]. C. Wilke, L. Ding, B. He, Estimation of time-varying connectivity patterns through the use of an adaptive directed transfer function, *IEEE Trans Biomed Eng.*, Vol. 55, Issue 11, 2008, pp. 1-21.

[21]. C. Yi, *et al.*, Constructing time-varying directed EEG network by multivariate nonparametric dynamical granger causality, *IEEE Trans. Neural Syst. Rehabil. Eng.*, Vol. 30, 2022, pp. 1412-1421.

(034)

Effective Connectivity for Brain Network Identification in Parkinson's Disease

Z. Fang [1], L. Albera [1], J. Duprez [1], J. F. Houvenaghel [1], H. Shu [2], Y. Kang [3] and R. Le Bouquin Jeannès [1]

[1] Univ Rennes, INSERM, LTSI - UMR 1099, F-35000 Rennes, France
[2] Southeast University, Laboratory of Image Science and Technology, 210096 Nanjing, China
[3] Shenzhen Technology University, College of Health and Environmental Engineering, 528118 Shenzhen, China
E-mails: laurent.albera@univ-rennes.fr and kangyan@sztu.edu.cn

Summary: Parkinson's Disease (PD) has an undeniable influence on patients, frequently resulting in a decline in motor function and impaired cognitive control. Brain connectivity is a revealing measure of cerebral mechanisms. Most studies in this field focus on the analysis of functional connectivity. The originality of our contribution lies in the characterization of effective connectivity in PD patients during the performance of a specific cognitive task. To do this, it is crucial to identify high-dimensional multivariate autoregressive models. We propose here a low-cost solution based on LASSO-type regression, for which the model order and regularization parameter are estimated automatically. Effective connectivity reveals an increased trend towards impulsive action selection and suggests degeneration of visual and information processing functions in PD patients.

Keywords: Parkinson's disease, HR-EEG, MVAR model, eBIC-LASSO, Genetic algorithm, Effective connectivity.

1. Introduction

Parkinson's Disease (PD) is a neural degenerative disease. Patients with PD suffer from deleterious effects on motor function such as rigidity and bradykinesia [1] and cognitive decline symptoms also affect the patient's life extremely severely [2]. One of the main cognitive difficulties in people with PD is an impaired ability to adapt effectively and quickly to changes in the environment, which was specifically classified as Cognitive Action Control (CAC) change. The Simon task is a conflict task which is widely used to evaluate CAC performance [3]. Numerous studies confirmed that cognitive function is associated with communication in brain regions and that changes in these brain networks are associated with neurological disorders [4]. Studying the interactions between brain regions during CAC will help clarify how neuro-degenerative diseases such as Parkinson's disease alter cognitive function [5].

In previous studies, connectivity using static fMRI was used to assess disease [6, 7]. However, ElectroEncephaloGraphy (EEG), which has excellent temporal resolution, is more suitable for assessing PD to evaluate changes in brain function during short time cognitive tasks. To investigate certain changes in brain regions and to map the neural network in PD patients, previous studies have focused on brain changes in functional connectivity patterns [8]. As a consequence, to investigate flow changes in brain regions and to map the neural network in PD patients, we have considered Effective Connectivity (EC) [9], which refers to causal interactions between brain regions.

Effective connectivity is generally implemented with the MultiVariate AutoRegressive (MVAR) model which is proved to be efficient and flexible in neural time series prediction [10]. Least Absolute Shrinkage Selection Operator (LASSO) regression was selected to identify the MVAR model with sparsity [11], since the brain is not fully active during a given activity [12]. Since the Genetic Algorithm (GA) [13] has an excellent capability of searching global extrema, it was employed to estimate the order of the MVAR model and the penalty parameter of the LASSO regression by minimizing an information criterion. We also conducted an analysis of cognitive degeneration through CAC performance change in PD patients, and alternation of EC network in brain.

2. Materials and Method

2.1. Data Acquisition

Ten Healthy Control (HC) subjects (5 males, 5 females) aged between 45 and 70 years (mean = 61.7, sd = 7.3) with 13.5 averaged education years (sd = 3.6) and ten PD patients (5 males, 5 females) aged between 45 and 73 years (mean = 60.4, sd = 7.3) with 12.5 averaged education years (sd = 3.2) were enrolled in the study. HC subjects and PD patients did not significantly differ in age, gender or education.

All participants underwent a neuropsychological assessment, which showed that they did not have severe cognitive deficits. In addition, all participants were free from moderate or severe psychiatric symptoms and did not have any present or past neurological pathology (other than PD for patients).

Participants were asked to perform a color version Simon task, which is a conflict task, widely used to

evaluate CAC performance for both congruent and incongruent tasks. High Resolution (HR) EEG signals of 2 s length and sampled at 1000 Hz were recorded using 256 channels. A set of 12000 trials of congruent and incongruent tasks was considered. This protocol was approved by a national ethics committee (CPP ID-RCB: 2019-A00608-49; approval number: 19.03.08.63626). This study was conducted in accordance with the declaration of Helsinki.

2.2. Method

We performed pre-processing on EEG signals in Brainstorm toolbox [14]. Firstly, an offset removal was applied. Secondly, a notch filter at 50 Hz and a FIR band pass filter from 1 to 100 Hz were applied. Thirdly, bad channels selected under visual inspection were removed or replaced using interpolation. Next, eye blinks and muscle artifacts were removed by Independent Component Analysis (ICA). Then, the signals were attributed to epochs relative to the stimulus onset from -700 ms to 1200 ms. Eventually, poor trials with excessive remaining noise were excluded.

EC is generally computed from cortical electrical activity. The latter can be derived by solving the EEG inverse problem. In the present study, we used the well-known weighted Minimum Norm Estimate (wMNE) algorithm. Next, the brain was parcellized into 148 regions of interest using Destrieux atlas [15]. Thus, a MultiVariate AutoRegressive (MVAR) model was fitted to these 148-dimensional signals.

The EEG signal can be considered as one realization of a p-order and N-sample MVAR sequence $\{\boldsymbol{x}(n)\}$:

$$\boldsymbol{x}(n) \;=\; \sum_{\ell=1}^{p} \boldsymbol{A}_\ell \, \boldsymbol{x}(n-\ell) + \boldsymbol{\varepsilon}(n), \qquad (1)$$

where $\{\boldsymbol{\varepsilon}(n)\}$ is an M-dimensional white Gaussian noise sequence with zero mean and covariance matrix $\sigma^2\boldsymbol{I}$ and where $\boldsymbol{A}_\ell$ is the ℓ-th submatrix of the $(M \times pM)$ coefficient matrix $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_p]$. We assume in the sequel that the p matrices $\boldsymbol{A}_\ell$ are linearly independent. The matrix $\boldsymbol{A}$ can be estimated using the Least Squares method [16] by minimizing the following cost function based on the Frobenius norm:

$$f(A) \;=\; \|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{B}\|_F^2, \qquad (2)$$

with $\boldsymbol{X} \;=\; [\boldsymbol{x}(p+1), \boldsymbol{x}(p+2), \ldots, \boldsymbol{x}(N)]$ and

$$\boldsymbol{B} \;=\; \begin{bmatrix} \boldsymbol{x}(p) & \boldsymbol{x}(p+1) & \cdots & \boldsymbol{x}(N-1) \\ \boldsymbol{x}(p-1) & \boldsymbol{x}(p) & \cdots & \boldsymbol{x}(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{x}(1) & \boldsymbol{x}(2) & \cdots & \boldsymbol{x}(N-p) \end{bmatrix}, \qquad (3)$$

notice that we have $N \;\geq\; pM$. Then, the LS solution is given by:

$$\boldsymbol{A} \;=\; \boldsymbol{X}\boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1} \qquad (4)$$

In the case of high-dimensional data, the MVAR model identification requires regularized solutions such as the LASSO regression [12], which promotes sparsity, since LS is proved to fail [17]:

$$g(\boldsymbol{A}) \;=\; \tfrac{1}{2}\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{B}\|_F^2 + \lambda\|\boldsymbol{A}\|_1, \qquad (5)$$

where λ is a hyperparameter balancing between the data fitting term and the penalty term. The Alternating Direction Method of Multiplier (ADMM) algorithm [18] was used to minimize g based on the following Lagrangian function:

$$\begin{aligned} \mathcal{L}(\boldsymbol{A}) \;=\; & \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{B}\|_F^2 + \lambda\|\boldsymbol{A}\|_1 + \\ & + \frac{\rho}{2}\|\boldsymbol{C} - \boldsymbol{A}\|_F^2 + < \boldsymbol{V}, \boldsymbol{A} - \boldsymbol{C} > \end{aligned} \qquad (6)$$

where $\boldsymbol{C}$ matrix is an additional variable related to $\boldsymbol{A}$ by the equality constraint $\boldsymbol{A} = \boldsymbol{C}$, introduced in splitting methods such as ADMM to facilitate the minimization. ρ was set to 1 and where $\boldsymbol{C}$, $\boldsymbol{A}$ and $\boldsymbol{V}$ were initialized as zero matrices of size $(M \times pM)$.

Regarding the update rule of $\boldsymbol{A}$, it is derived by cancelling the gradient of $\mathcal{L}$ with respect to $\boldsymbol{A}$, which leads to:

$$\begin{aligned} & \boldsymbol{A}^{(k+1)} \;= \\ & = \left(\boldsymbol{A}\boldsymbol{B}^T + \rho\boldsymbol{Z}^{(k)} - \boldsymbol{V}^{(k)}\right)(\boldsymbol{B}\boldsymbol{B}^T + \rho\boldsymbol{I})^{-1} \end{aligned} \qquad (7)$$

Consequently, the update rule of $\boldsymbol{C}$ is given by:

$$\boldsymbol{C}^{(k+1)} \;=\; S_{\lambda/\rho}\left(\boldsymbol{A}^{(k+1)} + \boldsymbol{V}^{(k)}/\rho\right), \qquad (8)$$

where S is the soft-thresholding operator:

$$(S_\eta(\boldsymbol{A}))_{i_1,i_2} \;=\; \begin{cases} A_{i_1,i_2} - \eta \text{ if } A_{i_1,i_2} > \eta \\ 0 \text{ if } \left|A_{i_1,i_2}\right| \leq \eta \\ A_{i_1,i_2} + \eta \text{ if } A_{i_1,i_2} < -\eta \end{cases}, \qquad (9)$$

with A_{i_1,i_2} the (i_1, i_2)-th component of matrix $\boldsymbol{A}$. The update rule of the multiplier is given using a gradient ascent:

$$\boldsymbol{V}^{(k+1)} \;=\; \boldsymbol{V}^{(k)} + \rho\left(\boldsymbol{A}^{(k+1)} - \boldsymbol{C}^{(k+1)}\right) \qquad (10)$$

In order to select the true MVAR model order and an appropriate penalty parameter for LASSO, the extended Bayesian Information Criterion ($eBIC$) has been minimized [19]:

$$eBIC(\lambda, p) \;=\; ln(\|\boldsymbol{X} - \boldsymbol{A}\boldsymbol{B}\|_F^2) + C_T df, \qquad (11)$$

where df is the number of non-zero values in the sparse matrix $\boldsymbol{A}$ obtained from LASSO optimization and $C_T \;=\; ln(N) + 2\gamma ln(Mp)$. We calculated $eBIC$ with $\gamma \;=\; 0.5$. Classically, the $eBIC$ cost function is minimized by scanning candidates in a given grid which could lead to expensive computational time for high dimensional MVAR models.

To overcome this drawback, in this contribution, we propose to minimize the cost function Eq. (11) by means of a Genetic Algorithm (GA).

The order $p \in \{1, \ldots, 25\}$ of the MVAR model and the penalty parameter $\lambda \in [0.1; 100]$ of LASSO were chosen as parents. To ensure that as many combinations as possible were considered, the crossover probability, the mutation probability and the generation gap were set to 80 %, 5 % and 50 %, respectively. Based on the $eBIC$ values of each iteration, fitness values were assigned in equal proportions. We adopted the Roulette Wheel Selection approach to select the retained parental genes. The $eBIC$ values were recorded for each generation. The genetic algorithm terminates when the same optimal solution is obtained in 30 consecutive iterations or when 100 generations are reached.

To analyze effective connectivity, we refer to the Partial Directed Coherence (PDC) [20] computed in pairs across the 148 regions and averaged over the beta band ([12-25] Hz) relevant in PD, where motor units oscillate strongly [21]. This measure is computed from the Fourier transform of the MVAR coefficients:

$$\widehat{\boldsymbol{A}}(f) = \sum_{\ell=1}^{p} \boldsymbol{A}_{\ell}\, exp(-j2\pi f \ell \Delta t), \qquad (12)$$

where Δt is the sampling period. The PDC causality measure is defined as:

$$PDC_{i_1,i_2}(f) = \frac{\hat{A}_{i_1,i_2}(f)}{\sqrt{\sum_{k=1}^{M} |\hat{A}_{\ell,i_2}(f)|^2}} \qquad (13)$$

The workflow of the proposed approach is depicted in Fig. 1.

3. Results and Discussion

Partial directed coherence was computed using 270 trials per task (congruent *vs* incongruent) per participant after rejecting bad trials. Given the (148×148) PDC matrix corresponding to the Destrieux anatomical parcellation, we applied a 0.3 threshold to keep the most significant flow directions in the brain. We were interested in comparing connectivity in PD patients and HC subjects.

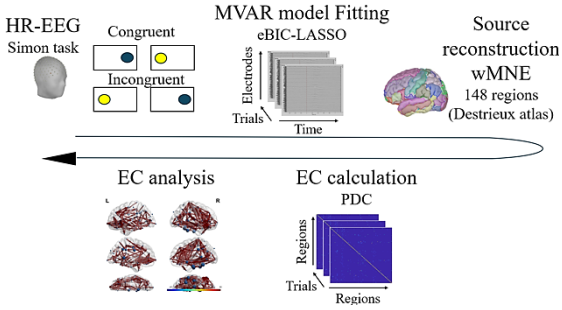


Fig. 1. Workflow of the proposed approach.

Compared to HC subjects, the connectivity of PD patients decreased from transverse frontopolar gyri and sulci to front-marginal gyrus and sulcus. Conversely, we observed an increase in connectivity in PD patients from the middle-posterior part of the cingulate gyrus and sulcus to the posterior-dorsal part of the cingulate gyrus in both congruent and incongruent conditions. The connectivity (i) from the superior segment to the anterior segment of the circular sulcus of the insula, (ii) from the lateral aspect to the planum polar of the superior temporal gyrus and (iii) from the superior frontal gyrus to the middle-anterior part of the cingulate gyrus and sulcus was particularly high in the incongruent condition, while it decreased in the congruent one. It is worth noting that at the subject level, the connectivity changed significantly: contrarily to the congruent condition (14 %), the number of disappearing and emerging flow directions in the incongruent condition was high (37.5 %) when passing from HC subjects to PD patients (see Fig. 2). The number of connectivity changes in the incongruent task was almost twice that of the congruent task. The connections that disappeared were found mainly in the temporal lobe and occipital lobe, while the connections that appeared were found mainly in the prefrontal and temporal lobes.

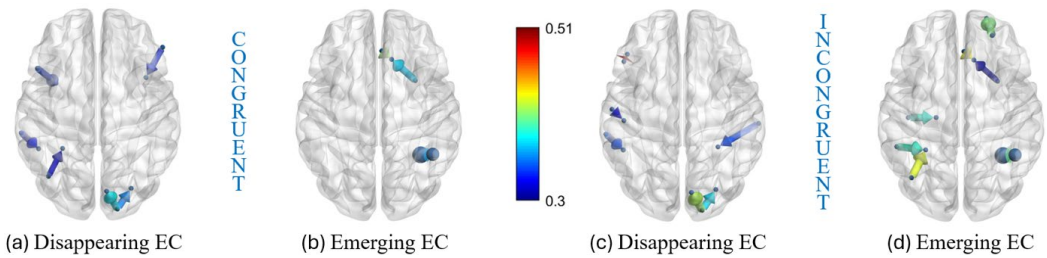


Fig. 2. Changes in effective connectivity when passing from HC subjects to PD patients in both tasks.

In this study, we aimed to explore how the networks change in PD patients in comparison with HC subjects during CAC. Clearly, the various EC modifications in specific regions illustrate some electrophysiological changes in PD patients. Our results showed that PD patients had increased EC in incongruent condition, which could suggest that they had to engage certain brain regions more importantly

than HC subjects. The changes in EC in occipital lobe and temporal lobe are noticeable in both conditions, which could be associated with the impairment of visual cognition in PD patients and information processing [22, 23]. Furthermore, the changes in frontal lobe might be associated with the change of CAC, observed in PD patients, according to [8]. However, this study didn't involve enough patients, which prevents us from investigating connectivity/behavioral performance and thus from elaborating firm conclusions.

4. Conclusion

This paper proposed a robust low-cost method based on a LASSO-type regression to compute simultaneously the order and the coefficients of high-dimensional MVAR models. The MVAR model order and regularization parameter were estimated automatically by minimizing an information criterion by means of a genetic algorithm. Such an approach allowed us to compute efficiently brain effective connectivity. However, improvements could be considered. Physiological hypotheses could be introduced to better target the connections of interest. Since genetic algorithms are time consuming, we plan to estimate the MVAR order simultaneously with the coefficients through a regularized approach. In this study which used the Simon task we showed for the first-time strong changes of EC in PD patients compared to HC subjects. In the future, it will be necessary to investigate more patients in order to assess if changes in EC explain the behavioral changes associated with PD.

References

[1]. M. T. Hayes, Parkinson's disease and parkinsonism, *The American Journal of Medicine,* Vol. 132, Issue 7, 2019, pp. 802-807.

[2]. R. A. Lawson, *et al.*, Cognitive decline and quality of life in incident Parkinson's disease: the role of attention, *Parkinsonism & Related Disorders,* Vol. 27, 2016, pp. 47-53.

[3]. J. R. Simon, A. P. Rudell, Auditory SR compatibility: the effect of an irrelevant cue on information processing, *Journal of Applied Psychology,* Vol. 51, Issue 3, 1967, 300.

[4]. D. S. Bassett, O. Sporns, Network neuroscience, *Nature Neuroscience,* Vol. 20, Issue 3, 2017, pp. 353-364.

[5]. K. R. Ridderinkhof, Activation and suppression in conflict tasks: Empirical clarification through distributional analyses, in Mechanisms in Perception and Action, *Oxford University Press*, 2002, pp. 494-519.

[6]. P. Fries, Rhythms for cognition: communication through coherence, *Neuron,* Vol. 88, Issue 1, 2015, pp. 220-235.

[7]. M. P. Van Den Heuvel, H. E. H. Pol, Exploring the brain network: a review on resting-state fMRI functional connectivity, *European Neuropsychopharmacology,* Vol. 20, Issue 8, 2010, pp. 519-534.

[8]. J. Duprez, *et al.*, Spatio-temporal dynamics of large-scale electrophysiological networks during cognitive action control in healthy controls and Parkinson's disease patients, *NeuroImage,* Vol. 258, 2022, 119331.

[9]. K. J. Friston, Functional, effective connectivity: a review, *Brain Connectivity,* Vol. 1, Issue 1, 2011, pp. 13-36.

[10]. A. Schlögl, G. Supp, Analyzing event-related EEG data with multivariate autoregressive parameters, in Progress in Brain Research (C. Neuper, W. Klimesch, Eds.), Vol. 159, *Elsevier*, 2006, pp. 135-147.

[11]. R. Tibshirani, Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society Series B: Statistical Methodology,* Vol. 58, Issue 1, 1996, pp. 267-288.

[12]. D. C. Mocanu, E. Mocanu, P. Stone, P. H. Nguyen, M. Gibescu, A. Liotta, Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science, *Nature Communications,* Vol. 9, Issue 1, 2018, 2383.

[13]. J. H. Holland, Genetic algorithms, *Scientific American,* Vol. 267, Issue 1, 1992, pp. 66-73.

[14]. F. Tadel, S. Baillet, J. C. Mosher, D. Pantazis, R. M. Leahy, Brainstorm: a user-friendly application for MEG/EEG analysis, *Computational Intelligence and Neuroscience*, 2011*,* Vol. 2011, 879716.

[15]. C. Destrieux, B. Fischl, A. Dale, E. Halgren, Automatic parcellation of human cortical gyriv and sulci using standard anatomical nomenclature, *NeuroImage,* Vol. 53, Issue 1, 2010, pp. 1-15.

[16]. R. A. Roberts, C. T. Mullis, Digital Signal Processing, *Addison-Wesley Longman Publishing Co., Inc*., 1987.

[17]. Y. Wang, C.-M. Ting, H. Ombao, Modeling effective connectivity in high-dimensional cortical source signals, *IEEE Journal of Selected Topics in Signal Processing,* Vol. 10, Issue 7, 2016, pp. 1315-1325.

[18]. S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, *Now Foundations and Trends*, 2011.

[19]. J. Chen, Z. Chen, Extended Bayesian information criteria for model selection with large model spaces, *Biometrika,* Vol. 95, Issue 3, 2008, pp. 759-771.

[20]. L. A. Baccalá, K. Sameshima, Partial directed coherence: a new concept in neural structure determination, *Biological cybernetics,* Vol. 84, Issue 6, 2001, pp. 463-474.

[21]. T. I. Panagiotaropoulos, V. Kapoor, N. K. Logothetis, Desynchronization and rebound of beta oscillations during conscious, unconscious local neuronal processing in the macaque lateral prefrontal cortex, *Frontiers in Psychology,* Vol. 4, 2013, 603.

[22]. W. R. Martin, M. Wieler, M. Gee, R. Camicioli, Temporal lobe changes in early, untreated Parkinson's disease, *Mov. Disord.,* Vol. 24, Issue 13, Oct. 15, 2009, pp. 1949-1954.

[23]. R. S. Weil, A. E. Schrag, J. D. Warren, S. J. Crutch, A. J. Lees, H. R. Morris, Visual dysfunction in Parkinson's disease, *Brain,* Vol. 139, Issue 11, 2016, pp. 2827-2843.

(036)

An Empirical Evaluation of Sliding Windows on Siren Detection Task Using Spiking Neural Networks

S. Kshirasagar [1,2], A. Guntoro [1] and C. Mayr [2]
[1] Robert Bosch GmbH, Robert-Bosch-Campus 1, 71272, Renningen, Germany
[2] TU Dresden, Dresden, Germany
E-mail: Shreya.Kshirasagar@de.bosch.com

Summary: Anomaly acoustic cues like siren sounds, when undetected, could lead to road safety issues like collisions or accidents. Auditory perception systems are resource bound when deployed on power constrained sensory edge devices. Spiking neural networks (SNN) premise brain-like computing with high energy-efficiency. This work presents a quantitative analysis of the variation of sliding window on the performance of acoustic anomaly detection task for siren sounds. We perform FFT based pre-processing and employ Mel-spectrogram features fed as input to the recurrent spiking neural network. SNN model in this work comprises of leaky-integrate-and-fire (LIF) neurons in the hidden layer and a single readout with leaky integrator cell. The non-trivial motivation of this research is to understand the effect of encoding behavior of spiking neurons with sliding windows. We conduct experiments with different window sizes, and the overlapping ratio within the windows. We present our results for performance measures like accuracy and onset latency to provide an insight on the choice of optimal window.

Keywords: Spiking neural networks, Acoustic perception, Anomaly detection, Siren sounds, Sliding window.

1. Introduction

Spiking neural networks closely mimic the sparse and asynchronous biological information processing. The models in SNNs are naturally operated in terms of time and are based on the principles of brain-inspired computing. Temporal perception of complex auditory scenes like speech signals is processed within the range of tens of hundreds of milliseconds (ms) [1], contributing to our investigation on temporal processing of siren audio sequences with the naturally adept, spiking neural networks.

Well known approaches look at the problem of siren sound detection using deep learning [2-6]. Authors in [6], extensively explore 2D CNN to detect siren signals based on the spectrum that is generated by combining multiple windowed FFT to generate image used as an input to the network, this subsequently leads to higher computational effort. SNNs on the other hand, exhibit temporal processing directly in their neurons, this motivates us to exploit time series tasks efficiently. Furthermore, with the advent of neuromorphic hardware [7, 8], these small-scale networks with sparsity introduced through the optimal choice of time constants could potentially save orders of magnitude of energy as shown in [7].

This motivates us to take a closer look at the pre-processing stage and the way of optimizing siren detection task. This work attempts to understand the intricacies of variation of sliding windows on the performance of temporal detection of siren sounds in order to trade-off between the accuracy and onset latency of prediction. We intend to model spiking neurons for real-time applications by inferring the impact of windowing in terms of encoding information. Through this empirical study, we aim to investigate the optimal window size with and without overlapping windows and encapsulate its impact on the task performance. We try to address the lack of understanding of the relation between neuronal decays and sliding windows. The remainder of this paper is organized as follows: Section 2 presents literature review on temporal detection of siren sounds and sliding windows for employed in different tasks in detail. We present the method employed using sliding windows and SNN training approach in Section 3. We detail the experimental setup for empirical study and subsequently present the results in Section 4. Finally, in Section 5 and 6 we present an outlook of the work, discussion and conclusion.

2. Related Work

Deep learning models namely – DNN (deep neural networks), CNN (convolutional neural networks), LSTM (long short-term memory) and hybrid CNN-LSTM are employed to solve human activity recognition (HAR) in [9]. Authors study the effect of sliding windows for preprocessing time-series data using four models and show improvement in accuracy, latency, and processing costs. Furthermore, authors in [10] provide an extensive characterization of windowing technique. They show the impact of diverse window sizes for HAR. Other interesting techniques like adaptive sliding windows are studied in [11] for assisted living application. Authors in [12], explore pose pattern recognition for sensors and extend the study to evaluate the impact on sliding windows. Their study is in alignment with the prior research that shows introduction of overlapping windows increases the accuracy of pattern recognition. Overall, through literature study we garner results on the significance of the choice of optimal window size and sliding

windows. To the best of authors knowledge, there is scarce research on sliding window variation for siren detection task using SNNs. Through this empirical study, we attempt to understand the impact of sliding windows on the performance for siren detection task using SNNs. We evaluate the relation between spiking neurons and the sliding windows in terms of accuracy and onset latency.

Neuroscientific study has shown that leak channels exist in various synaptic transmissions in visual cortex [13] and in sodium ion leak channels [14]. On neuromorphic datasets NMNIST and SHD, for different spiking neurons, leakages are studied for spatio-temporal pattern recognition in [15]. Authors in [15] explore the impact of synaptic and membrane time constants for three different spiking neuron models on pattern recognition and conclude the significance of neuronal leakage for both temporal features and explicit presence of recurrent connections. Authors in [16] have shown the importance of leak for LIF neurons in terms of robustness to noise by acting as high frequency filter. In parallel, authors also comment on the statistical relationship of sparsity introduced through leaky models and hardware efficiency through synaptic operations. Here, we aim to understand if there is a relation between sliding windows and neuronal decays. This work attempts to partially answer this question by conducting an empirical evaluation of spiking neuron time constants in recurrent SNNs for acoustic anomaly detection.

3. Method

We use artificial siren sequences generated from the publicly available siren dataset [17] to train our models. The artificial audio sequences are sampled at a sampling frequency of 48 kHz. We employ a small FFT window to minimize the hardware effort. Taking into account that our signal of interest, i.e., siren sounds, have a fundamental frequency between 400 Hz to 600 Hz, we start with a window size of 4096 which corresponds to 85.33 ms and we reduce window size further below to 2048, …, 512 for this empirical evaluation.

Windowing is applied on the audio sequences of 30 s before FFT calculation as shown in Fig. 1. Feature extraction is carried out using Mel spectrogram. The input to the hidden layer of SNN is fixed to 64 Mel channels. The SNN has a topology of 64-100-1 with recurrent connections in the hidden layer. We keep constant parameters (structural/topological) throughout the experiments for homogeneity.

3.1. Data

Dataset in [17] is comprised of siren sounds and road noise. The dataset consists of different types of sirens sounds namely wail, yelp, hi-lo. We modify the publicly available dataset to perform temporal predictions using artificially generated audio sequences. More specifically, we utilize single channel siren and traffic noise recordings from the dataset presented in [17] and split the samples of each class with an 80/20 ratio into train and validation samples. All samples are resampled to a shared sample rate of 48 kHz. Based on the noise samples, a continuous sequence is generated. To each of these sequences of 30 s duration, a single siren sound is added at a random time with random length. To ensure accurate measurement of onset latency and network stability we enforce the first 1 s of the artificial sequences to exclude siren signals.

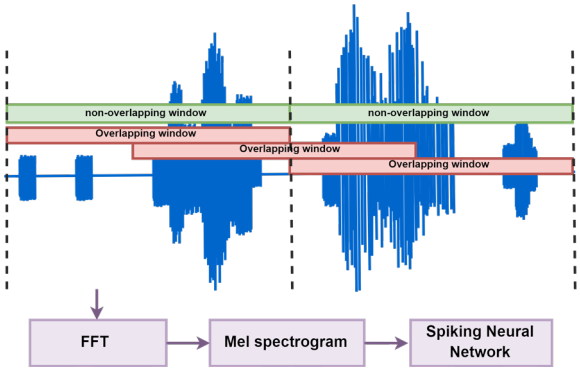


Fig. 1. Block diagram highlights the sliding windows on acoustic anomaly sequences. Different window sizes are provided in time slices as an input to FFT; 64 Mel features are extracted to inject as current input to the recurrent SNN.

3.2. Feature Extraction

We use windowing technique to deconstruct temporal features into spatial features to analyze different frequencies. Hann window is used for smoothening of edges in FFT calculations. In this work, since our focus is on understanding the performance of windowing for siren detection task, we use sliding windows with and without overlapping windows. We use log-scaled Mel spectrograms as input features to our SNN model. The window length and hop length are varied in the order of power of 2 to obtain optimal design choices for better performance. For the Mel transformation, we impose a lower frequency limit of 50 Hz to cover the noise signals so that the network could easily differentiate between noise characteristics. The upper limit is set according to respective window size. For example, w = 512 signals extracted are only within frequency 0-513 Hz. We consider a total of 64 Mel channels to constrain the feature range. Afterwards, the features are converted to dB scale and min-max normalization is applied to each time slice.

3.3. Network Architecture

SNN is designed as in [18] shown in Fig 2 comprises of a hidden spiking layer and a single

readout for predicting siren or not. The information processing in the hidden layer of the proposed model is in terms of spikes. LIF neurons are analogous to the biological neuronal processing. When the input stimulus crosses the threshold voltage, neuronal firing occurs. We aim to understand the effect of sliding windows and the neuronal processing on network predictions. Therefore, we conduct experiments with various synaptic (τ_{mem}) and membrane (τ_{syn}) time constants of the neurons to evaluate the impact of sliding windows.

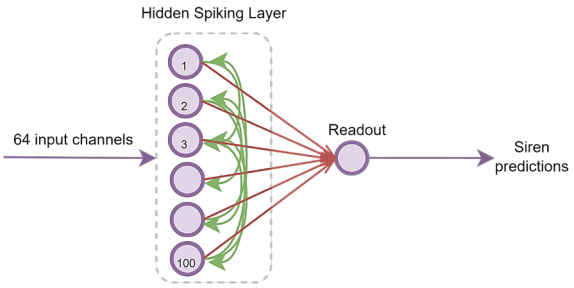


Fig. 2. Overview of the recurrent SNN used in this work is highlighted. The features extracted are given as 64 input channels to the SNN. Spiking architecture comprises of 100 hidden neurons with recurrent connections and a single readout neuron for siren predictions.

The parameters set for the evaluation of siren detection task are described in Table 1. Surrogate gradient based method is used to approximate the derivative of the LIF recurrent cell [19]. We employ a Leaky-integrator (LI) cell as a readout neuron, which has continuous-valued output. We train SNNs used in this work on Norse [20], an extension of PyTorch [21].

Table 1. Description of network structure and LIF neuron parameters.

Network structure	64-100-1
Threshold voltage	1 V
Reset potential	0 mV
Membrane time constant	2 ms
Synaptic time constant	2 ms
Refractory period	0 ms

The differential equations and dynamics of current-based LIF neuron is extensively discussed and presented in equations (1), (2) and (3) in [19]. Neurons have a membrane potential that decays with a membrane time constant (τ_{mem}). Synaptic currents follow specific temporal dynamics. The exponentially decaying current triggered by pre-synaptic input leads to the second dynamics of LIF neurons. This exponential decay of synapses is termed as synaptic time constant (τ_{syn}). The specific dynamics of CUrrent BAsed (CUBA)- LIF neurons for exponential decay of synaptic currents and membrane potential are presented in [15] in a set of equations (3) and (4).

4. Experiments and Results

4.1. Experimental Setup

We employ surrogate gradient method [19] to train our recurrent SNN model. In this work the model is trained for 100 epochs, with a batch size of 16 on Nvidia V100. Adamax optimizer is applied with a learning rate of 1×10^{-3}.

1st experiment setup: We set constant parameters for LIF neurons in the hidden layer of the SNN. Threshold voltage of neuron is set to 1 V, both the time constants (membrane and synaptic) are set to 2 ms. We design experiments with variation in window length ($w = 2^X$, i.e. 4096, 2048, ..., 512) and hop lengths (h). We choose three setups for our evaluation, h = w (no overlap), h = 0.5w (50 % overlap) and h = 0.25w (75 % overlap).

2nd experiment setup: We perform experiments to understand the correlation between sliding windows and neuronal processing speed. Membrane (τ_{mem}) and synaptic time (τ_{syn}) constants are a variable parameter with different window sizes and hop sizes to obtain results for accuracy and onset latency for the siren prediction task. We further investigate the impact of windowing without any overlaps for different time constants to obtain an optimal window size.

4.2. Effect of Overlapping Windows

The focus of this work is to demonstrate the effect of sliding windows with and without overlap on accuracy and onset latency for siren predictions.

As we expected and evident from Fig 3(a), having overlapping windows for the same window size helps to improve the training accuracy. However, we need to keep in mind that we need to feed the input more frequently to the network, e.g., for h = 0.5w, network needs to process input twice faster. The increase in performance for overlaps within sliding windows is due to granularity of information which increases the focus on signal of interest. A slight drop in accuracy with smaller windows, and for their respective overlapping hop lengths is observed. This effect is observed in the window with sample size of 512 with hop length h = 0.5w, it corresponds to a frequency range of 255 Hz. Siren sounds have a typical characteristic frequency of 400-600 Hz. The covered frequencies are below the signal of interest for window sizes below 256.

We investigate the time to detect siren sounds using the onset of events by predicting the neuronal state change. This gives us an intuition of how the fine temporal resolution of the spiking neurons influences the latency. Hence, we design experiments to vary the sliding window with overlapping and fixed windows and measure the time to first event, given an audio sequence is being processed within the ground truth (label). We observe the latency values of validation samples in the last epoch and average over batch size.

As observed in Fig 3(b), the introduction of overlapping windows has a modest influence on latencies within each window. It is our understanding that the hop length introduces faster processing, through reduction in latency. Based on processing time alone, we expect 2× reduction on latency h = 0.5w, 4× for h = 0.25w. A latency improvement of 5× is achieved for window size of 4096, this is explainable from the neuronal sensitivity to detect siren sound events in windows with increased information granularity.

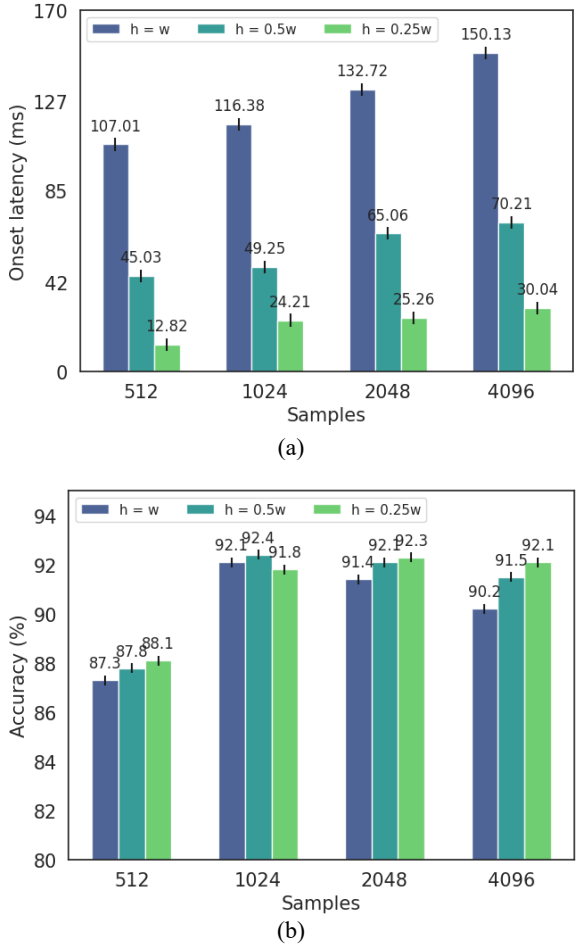


Fig. 3. Experimental results for sliding window variation on the siren detection task with neuron time constants as 2 ms (a) Introduction of hop length improves the prediction accuracy, (b) Onset latency reduces for overlapping windows.

4.3. Relation Between Sliding Windows and Time Constants

We explore the relation between neuronal time constants of charge based LIF and their impact on the performance with sliding windows with overlaps. To garner results in this direction, we perform experiments with variation in overlap ratios for a window size of 4096 with different τ_{mem} and τ_{syn} ranging from 1ms to 100 ms. From Fig 4(a), a clear trend is observed that the accuracy is best performing for neuronal time constants ranging between 1-2 ms. Higher the time constants, the neurons decay at a much slower rate and this effect leads to slight degradation in accuracy for overlapping windows. Nearly 5 % accuracy drop occurs for h = 0.25w for higher time constants due to the reason that fast responses or high sensitivity of neurons with slower membrane decay leads to missing of the crucial information within overlapping windows. Whereas for smaller time constants, this is reflected as a benefit in terms of modest accuracy deviation. With the introduction of overlapping windows (h = 0.5w, h = 0.25w), the speed improves 2.4× steadily as seen in Fig 4(b). Another interesting trend is seen in the onset latency. It is indicative of a linear trend for latency and time constants. Smaller time constants lead to higher accuracy and nearly 4× processing speed with the addition of overlapping windows for a fixed window size of 4096.

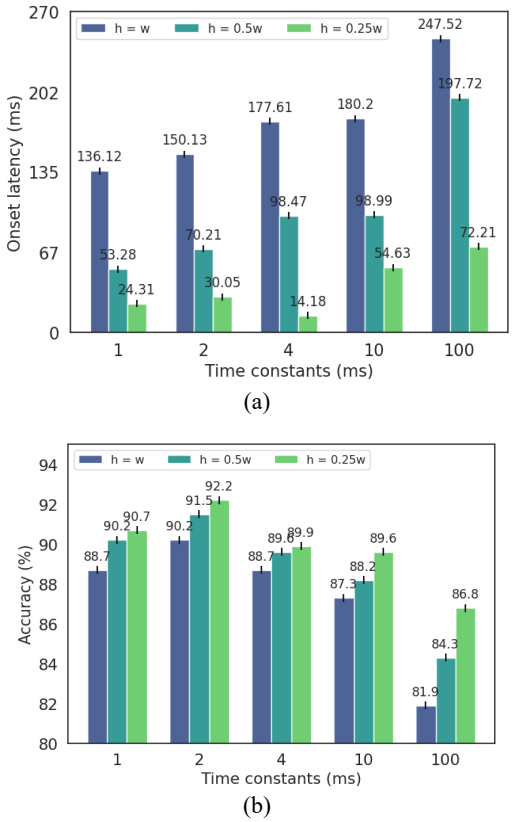


Fig. 4. Time constants (τ_{mem}) and (τ_{syn}) variation for siren detection task with w = 4096 and overlapping windows. (a) Introduction of hop length improves the prediction accuracy for different smaller time constants (b) Onset latency reduces for smaller time constants and overlapping windows.

Secondly, we vary time constants with a variable window size of 4096, ..,512 and without overlaps to analyze the correlation between neuronal decays for windowing. The accuracy is higher for window sizes ranging between 85 ms – 50 ms and it worsens with smaller windows. From Table 2, it is evident that for

smaller constants with an optimal window 85.33 ms – 42.66 ms, best performance is achieved.

4.4. Feature Resolution for Sliding Windows

We demonstrate the feature resolution using different sliding windows on log Mel spectrogram as depicted in Fig. 5. From the experimental setup detailed in Section 4.1, the window and hop sizes are varied to understand the behavior of the network. LIF parameters and time constants are set according to Table 1. Whereas the panels represent the window sizes with different overlapping windows to provide an overview on the Mel spectrogram features and corresponding siren predictions.

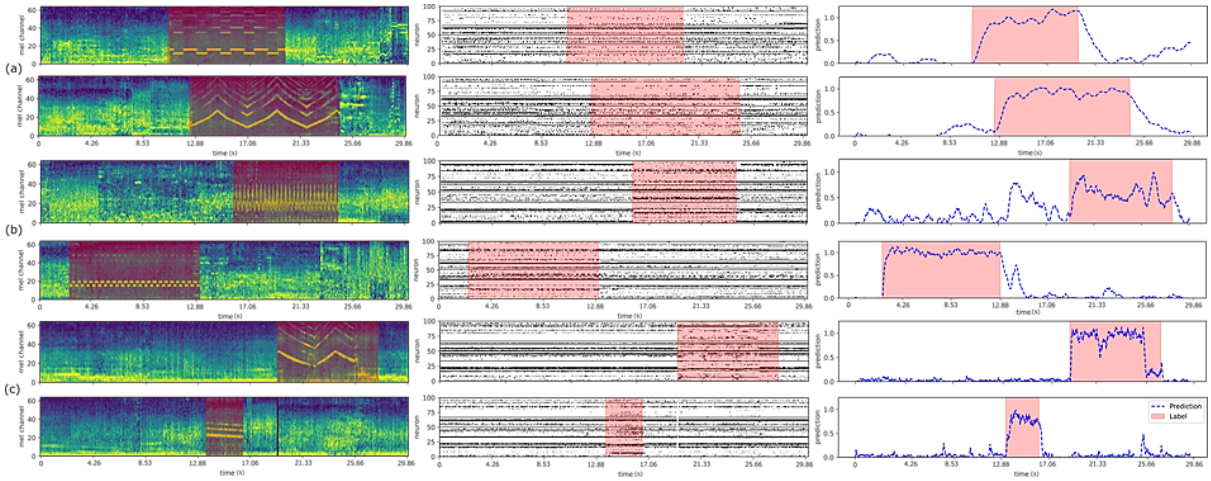


Fig. 5. Demonstration of feature resolution on Mel spectrogram for sliding window variation on the siren detection task (presence of siren is highlighted in red). (a) First two rows: w = 4096 (h = w); (b) Next two rows: w = 2048 (h = 0.5w); (c) Last two rows: w = 1024 (h = 0.25w). The panels in each row showcase stages of information processing in the network pipeline.

Table 2. Accuracy values for different time constants $\tau = \tau_{mem} = \tau_{syn}$ across variation in window sizes with no overlap. Smaller time constants and larger windows tend to achieve accurate siren predictions.

H = W	τ = 1 **ms**	τ = **2 ms**	τ = **4 ms**	τ = **10 ms**	τ = **100 ms**
4096	87.1 %	90.9 %	86.4 %	85.6 %	80.8 %
2048	89.2 %	88.9 %	88.1 %	87.4 %	83.2 %
1024	90.0 %	88.7 %	89.4 %	86.0 %	81.7 %
512	88.2 %	87.3 %	86.1 %	86.0 %	82.6 %

With w = 4096 and h = w, in the first row the features of interest are in lower frequency range due to higher energy concentration. With lower window sizes and smaller overlaps (w = 2048, h = 0.5w) the prediction strength gets stronger due to finer resolution in data points. It is interesting to note the increased spike activity in the hidden layer of SNN. For w = 1024 (h = 0.25w), we see a noticeable difference in feature resolution due to increased granularity with distinguishable noise and siren sounds.

Overlap ratios lead to finer resolution and this is reflected in terms of spike activity which is observed to slightly increase. However, the focus of this work is not to examine the spike activity with different feature resolution.

5. Discussion

We trained recurrent SNN in different experimental setup to detect siren sounds on modified public dataset in [17] to firstly understand the impact of sliding windows on the task performance and secondly, to provide a basis for the correlation between sliding windows and the leaky behavior of LIF neurons. Our results are indicative of a performance boost in terms of accuracy using sliding windows with overlap. This is due to high data points within the overlaps and less probability to miss information. Interestingly, higher overlap ratio for hopping windows in smaller window regimes show best performance with 92.4±0.8 % accuracy closely matching the performance with larger window sizes. We particularly noted slight drop in accuracy with smaller windows below the signal of our interest (siren sounds). Therefore, we conclude that sliding windows with overlapping windows translate into improved task performance. Collectively, our results appear consistent with the body of literature on impact of sliding windows for various detection tasks.

Research works show the significance of time constants of spiking neuron models to help improvise

the performance of spatio-temporal pattern recognition task [15]. Neurons tend to show the same behavior that leakages exist in neuronal models as underpinned by biology [13, 14].

The relation between the leaky behavior of spiking neurons in terms of exponential decay as time constants and that of the sliding windows is understood step-by-step through experiments in this work. At a higher level of abstraction, we understand the variation of sliding windows and the impact on onset latency and accuracy. Due to the faster response time that occurs with smaller membrane and synaptic time constants, for overlapping windows the obtained accuracy is the highest and thus leads to faster processing speed. It is noteworthy, as the time constants fall in the range of audio sequences this adversely impacts the accuracy by nearly 10 % for overlap ratios of 75 %. We also note that for smaller time constants and smaller windows, accuracy and onset latency are 5.5 % higher and 2.4× lower respectively in contrast to higher time constants.

6. Conclusion

In this work, we trained recurrent SNNs for an auditory anomaly detection task. We conducted an empirical study on the impact of sliding windows on accuracy and onset latency. Through our preliminary results, we can conclude that the trade-off between accuracy and latency leads to an optimal window of 30ms to 80ms for constant neuron parameters. We further performed experiments with time constants as a variable to reflect on the relation between spiking neurons and sliding windows. We observe for 4× processing speed and with smaller time constants the accuracy is higher.

Acknowledgements

This work received funding from European Union's Horizon Europe research and innovation programme under the grant agreement No. 101070374.

References

[1]. M. D. Mauk, D. V. Buonomano, The neural basis of temporal processing, *Annual Review of Neuroscience*, 2004, Vol. 27, Issue 1, pp. 307-340.

[2]. B. Fazenda, H. Atmoko, F. Gu, L. Guan, A. Ball, Acoustic based safety emergency vehicle detection for intelligent transport systems, in *Proceedings of the ICROS-SICE International Joint Conference,* 2009, pp. 4250-4255.

[3]. D. Carmel, A. Yeshurun, Y. Moshe, Detection of alarm sounds in noisy environments, in *Proceedings of the 25th European Signal Processing Conference (EUSIPCO'17),* 2017, pp. 1839-1843.

[4]. M. Cantarini, A. Brocanelli, L. Gabrielli, S. Squartini, Acoustic features for deep learning-based models for emergency siren detection: an evaluation study, in *Proceedings of the 12th International Symposium on Image and Signal Processing and Analysis (ISPA'21)*, 2021, pp. 47-53.

[5]. L. Marchegiani, P. Newman, Listening for sirens: locating and classifying acoustic alarms in city scenes, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, Issue 10, 2022, pp. 17087-17096.

[6]. V.-T. Tran, W.-H. Tsai, Acoustic-based emergency vehicle detection using convolutional neural networks, *IEEE Access*, Vol. 8, 2020, pp. 75702-75713.

[7]. M. Davies, et al., Loihi: A neuromorphic manycore processor with on-chip learning, *IEEE Micro*, 2018, Vol. 38, Issue 1, pp. 82-9.

[8]. P. A. Merolla, et al., A million spiking-neuron integrated circuit with a scalable communication network and interface, *Science,* 2014, Vol. 345, pp. 668-673.

[9]. M. Jaén-Vargas, K. M. Reyes Leiva, F. Fernandes, et al., Effects of sliding window variation in the performance of acceleration-based human activity recognition using deep learning models, *Peer J. Computer Science*, Vol. 8, 2022, e1052.

[10]. O. Banos, J. M. Galvez, M. Damas, H. Pomares, I. Rojas, Window size impact in human activity recognition, *Sensors*, Vol. 14, Issue 4, 2014, pp. 6474-6499.

[11]. C. Ma, W. Li, J. Cao, J. Du, Q. Li, R. Gravina, Adaptive sliding window based activity recognition for assisted livings, *Information Fusion*, Vol. 53*,* 2020, pp. 55-65.

[12]. G. Wang, Q. Li, L. Wang, W. Wang, M. Wu, T. Liu, Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors, *Sensors*, Vol. 18, Issue 6*,* 2018, 1965.

[13]. T. P. Snutch, A. Monteil, The sodium "leak" has finally been plugged, *Neuron*, Vol. 54, Issue 4, 2007, pp. 505-507.

[14]. D. Ren, Sodium leak channels in neuronal excitability and rhythmic behaviors, *Neuron*, 2011, Vol. 72, Issue 6, pp. 899-911.

[15]. M. S. Bouanane, D. Cherifi, E. Chicca, L. Khacef, Impact of spiking neurons leakages and network recurrences on event-based spatio-temporal pattern recognition, *Frontiers in Neuroscience*, Vol. 17*,* 2023, 1244675.

[16]. S. S. Chowdhury, C. Lee, K. Roy, Towards understanding the effect of leak in spiking neural networks, *Neurocomputing*, Vol. 454, 2021, pp. 83-94.

[17]. M. Asif, et al., Large-scale audio dataset for emergency vehicle sirens and road noises, *Sci Data*, Vol. 9, 2022, 599.

[18]. S. Kshirasagar, B. Cramer, A. Guntoro, C. Mayr, Auditory anomaly detection using recurrent spiking neural networks, in *Proceedings of the International Conference on Artificial Intelligence Circuits and Systems (AICAS'24),* 2024.

[19]. E. O. Neftci, H. Mostafa, F. Zenke, Surrogate gradient learning in spiking neural networks: bringing the power of gradient-based optimization to spiking neural networks, *IEEE Signal Processing Magazine*, Vol. 36, 2019, pp. 51–63.

[20]. C. Pehle, J. E. Pedersen, Norse: A library To Do Deep Learning with Spiking Neural Networks, *GitHub*, 2019.

[21]. A. Paszke, et al., PyTorch: An Imperative Style, High-Performance Deep Learning Library, *GitHub*, 2019.

(039)

GPU-accelerated Inference Benchmarking for Boosting Models

Jérémie Farret, Roghayeh Soleymani and Nitish Kumar Pilla
Mind in a Box Inc., 3575 St Laurent Boulevard, Suite 200 Montreal, Quebec H2X 2T6, Canada
Tel.: +1-833-636-2269
E-mail: jeremie@mindinabox.ai

Summary: Gradient boosting trees such as lightGBM and XGBoost are widely used in real-world applications of machine learning when dealing with Tabular data. In addition, the data processing and feature engineering modules are computation-heavy components in real-time inference applications of machine learning. This paper examines the performance of various data processing libraries (Pandas, NumPy, Polars, Dask, cuDF, Dask-cuDF) and machine learning frameworks (LightGBM, FIL, ONNX) within a multi-processor and GPU-accelerated environment. It includes a comprehensive benchmarking analysis, focusing on execution time and memory usage during inference tasks. Conducted on the Mind in a Box platform using datasets of varying sizes, the study specifically assesses the efficacy of these tools in data processing and GPU-accelerated inference. Significant findings reveal cuDF and Dask-cuDF effectiveness in handling large datasets and FIL's enhanced performance in machine learning inference, particularly with LightGBM and XGBoost models. This research provides essential insights for choosing the most suitable tools for GPU-accelerated data processing and inference.

Keywords: GPU-accelerated data processing, Machine learning frameworks, Benchmarking study, Large dataset handling, Inference efficiency, cuDF optimization, FIL performance, Mind in a Box.

1. Introduction

In the realm of data science and machine learning, the integration of General-Purpose Graphics Processing Units (GPUs) has marked a paradigm shift, notably enhancing data processing and analysis. This paper embarks on a comprehensive benchmarking study of key Python data processing libraries (Pandas, NumPy, cuDF, Polars, Dask, Dask-cuDF) [1-3] and machine learning frameworks (LightGBM, XGBoost, FIL, ONNX) within a GPU-accelerated environment [4].

For data processing libraries, Pandas stands out for its user-friendly interface but often struggles when handling large datasets. NumPy, while excelling in numerical computations, falls short in advanced data manipulation capabilities. CuDF, optimized for GPU usage, significantly enhances performance but faces compatibility issues with Pandas. Dask, on the other hand, enables processing of large datasets efficiently, though its integration, especially with Dask-CuDF, can be complex to set up. Polars, another emerging library, offers high-performance data processing with a focus on speed and efficiency, but its ecosystem is not as mature as Pandas. Dask excels in parallel computing, allowing for scalable data processing, but its complexity and overhead can be challenging for simpler tasks Regarding machine learning frameworks, LightGBM is efficient for large datasets but less effective for small ones.

XGBoost is versatile but resource intensive. FIL accelerates inference in tree-based models but is model-type specific. ONNX offers model interoperability but adds complexity in management. The study provides insights into selecting appropriate tools for specific tasks in GPU-accelerated research settings.

Our evaluation, conducted on the Mind in a Box Catalyst™, meticulously measures performance across diverse data scales, focusing on execution time, memory usage, and scalability. This research not only serves as a practical guide for professionals dealing with large-scale data and complex model inferences but also lays the groundwork for future advancements in optimizing data processing workflows and machine learning model deployment leveraging the GPU in the era of big data.

2. Experiments Setup and Methodology

Our experiment was conducted in a Python-based data science lab environment (Kubeflow), utilizing JupyterLab and Jupyter Notebook. This environment is to be further described in relation to the experimental protocol in the full publication. The implementation process involved two key steps:

In the first step, a variety of standard data operations were executed to assess the performance metrics of each library. These operations included counting, calculating means and standard deviations, summing, product calculations, and both arithmetic and lagging operations. In the arithmetic operation, the process involved dividing one set of numerical values by another within a NumPy array.

In the second step, we benchmarked the model prediction performance of LightGBM, XGBoost, FIL (Forest Inference Library), and ONNX. Our approach included setting up a GPU-optimized environment using dask_cuda. LocalCUDACluster and dask.distributed.Client, and processing data with dask_cudf for efficiency. The methodology encompassed training LightGBM and XGBoost models via their respective Dask interfaces, with

performance metrics like time and memory usage monitored through a custom profile memory and time function. We also utilized the Forest Inference Library (FIL) for rapid GPU inference and converted models to ONNX format for interoperability assessment. This streamlined process enabled a comprehensive comparison of each framework's efficiency and speed in a GPU-accelerated context.

Building on this foundational approach, a granular performance analysis is performed by documenting execution times and memory usage for both LightGBM and XGBoost across different computational environments: GPU, CPU, FIL, and ONNX. This detailed examination shows the distinct advantages of leveraging FIL for GPU inference, reflecting its superior memory efficiency and processing speed. Specifically, our results illustrate that FIL significantly enhances the performance of LightGBM and XGBoost, making it a sophisticated tool for applications demanding high-speed data processing and minimal memory footprint advantaging from GPU acceleration power when available.

Our benchmarking focused on execution time, memory consumption, and GPU memory usage – crucial factors that influence real-time processing, indicate resource efficiency, and gauge each tool's effectiveness in leveraging GPU resources. By performing multiple runs per operation for each library and framework, we ensured the consistency and reliability of our results, laying a solid foundation for data-driven decision-making in selecting the optimal tools for data processing and machine learning tasks in GPU-accelerated environments.

3. Performance Analysis and Results

In our evaluation of Pandas, NumPy, cuDF, Polars, Dask and Dask-cuDF across datasets of 100000; 2 million; and 15 million records, we observed distinct performance patterns. For smaller datasets, cuDF demonstrated high efficiency in execution time and memory usage, while Pandas and NumPy were faster but less memory efficient [4, 5].

Referencing the visual analytics provided in Fig. 1 and Table 1, we can enhance our understanding of the performance dynamics among the different data processing libraries and machine learning frameworks.

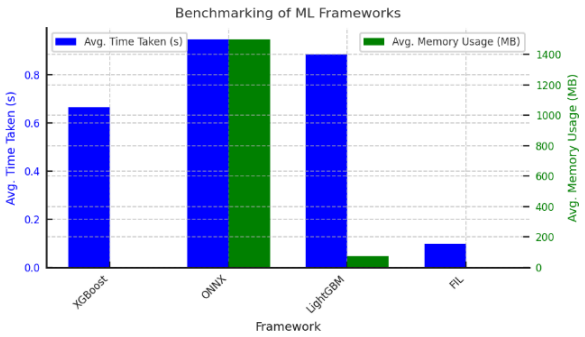


Fig. 1. Benchmarking of ML frameworks.

Table 1. Data Processing Libraries Performance.

	100 k			2 M			15 M		
Library	**Time(S)**	**Memory (MB)**	**GPU Memory (MB)**	**Time(S)**	**Memory (MB)**	**GPU Memory (MB)**	**Time(S)**	**Memory (MB)**	**GPU Memory (MB)**
cuDF	0.124	18.30	**0.20**	**0.161**	0.00	**3.20**	**0.187**	0.00	**23.40**
Dask	0.517	0.64	N/A	2.461	20.95	N/A	10.225	146.33	N/A
Dask-cuDF	1.483	3.57	0.0	0.509	**13.71**	25.6	**0.969**	**63.59**	68.0
NumPy	**0.094**	7.09	N/A	0.313	352.42	N/A	4.071	1875.37	N/A
Pandas	**0.105**	**0.63**	N/A	0.392	58.01	N/A	3.759	217.63	N/A
Polars	0.152	15.65	N/A	0.859	213.60	N/A	4.524	1495.70	N/A

The graphical representation in Fig. 1, which benchmarks ML frameworks including XGBoost, ONNX, LightGBM, and FIL, enables an immediate visual comparison of execution times and memory usage. Notably, it becomes evident that FIL, coupled with XGBoost, offers a compelling combination of speed and memory efficiency, as observed in the minimal execution times and reduced memory footprint [3, 6]. Meanwhile, Table 1 provides a meticulous breakdown of performance across different dataset sizes for Pandas, NumPy, cuDF, Polars, Dask, and Dask-cuDF, mirroring the insights drawn from scholarly discussions on scalable dataframes and the increasing necessity for distributed in-memory dataframes as discussed by DeLisi et al. [1] and Chen et al. [4].

Polars displayed notable performance, particularly in terms of execution time, but its memory usage was higher compared to cuDF and NumPy. At larger scales, cuDF maintained its efficiency, but NumPy showed limitations in handling larger datasets [5]. Dask-cuDF, though slower for smaller datasets, proved to be more effective for large-scale processing, suggesting its suitability for distributed computing scenarios. Our analysis of machine learning frameworks in a GPU-accelerated environment included ONNX with XGBoost and LightGBM, and the FIL with these models. ONNX showed higher execution times and memory usage, indicating a trade-off for its deployment flexibility. GPU versions of LightGBM and XGBoost outperformed their CPU counterparts, with XGBoost on GPU being particularly

efficient. FIL, especially with XGBoost, stood out for its low execution times and minimal memory usage, highlighting its optimization for GPU-accelerated environments.

For optimal data processing performance, cuDF is recommended for both small and large datasets due to its efficient balance of execution time and memory usage [4]. Dask-cuDF is suitable for distributed computing scenarios, especially with large datasets exceeding 30 million data points. In GPU-accelerated environments, FIL with XGBoost is recommended for tasks requiring rapid execution and minimal memory usage. ONNX offers deployment flexibility across platforms, although it exhibits higher execution times and memory usage. Prefer GPU versions of LightGBMand XGBoost for enhanced performance, with XGBoost on GPU being particularly efficient.

4. Conclusions

The conclusions drawn from our comprehensive analysis serve to highlight the pivotal role that the right choice of data processing libraries and machine learning frameworks plays in the realm of GPU-accelerated computing environments. Through meticulous benchmarking, cuDF has emerged as a standout performer, adeptly managing both modest and voluminous datasets with remarkable efficiency. Its counterpart, Dask-cuDF, excels when tasked with the demands of sprawling, distributed datasets. This study further illuminates the limitations of Pandas and NumPy when faced with scaling challenges, a detail particularly relevant for tasks bound by dataset size. The Fast Inference Library (FIL), when employed with XGBoost, is distinguished by its optimal use of GPU resources, highlighting an enviable efficiency that is critical for performance-intensive applications. While ONNX offers deployment flexibility, it does so at the expense of efficiency, suggesting that its use case should be carefully considered in resource-sensitive environments. Significantly, our research corroborates the superior performance of GPU-based versions of LightGBM and XGBoost over their CPU-based alternatives, with XGBoost on GPU displaying particularly remarkable enhancements. This distinction is not only a testament to the advancements in GPU technology but also serves as a critical consideration for practitioners aiming to leverage the full power of modern computing architectures. In sum, the judicious selection of these tools, informed by our findings and grounded in the extensive body of literature, is essential for maximizing computational efficiency and achieving the desired outcomes in data-intensive tasks and real-time applications of these machine learning models.

References

[1]. M. R. DeLisi, Scalable dataframes: design and implementation of a distributed in-memory dataframe, *NSF Public Access*, 2020, pp. 3-6.

[2]. M. Rocklin, Dask: parallel computation with blocked algorithms, in *Proceedings of the 14th Python in Science Conference*, 2015, pp. 130-136.

[3]. D. B. Kirk, W. W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach, *Morgan Kaufmann*, 2012.

[4]. Y. Chen, et al., CUDF: A GPU DataFrame library, *IEEE Transactions on Computers*, Vol. 68, Issue 12, 2019, pp. 1787-1797.

[5]. W. McKinney, Data structures for statistical computing in Python, in *Proceedings of* the *9th Python in Science Conference*, 2010, pp. 51-56.

[6]. A. Syberfeldt, T. Ekblom, A comparative evaluation of the GPU vs. the CPU for parallelization of evolutionary algorithms through multiple independent runs, *International Journal of Computer Science & Information Technology (IJCSIT)*, Vol. 9, Issue 3, 2017, pp. 1-14.

(040)

Assessing Chronic Wound Area Measurement with Machine Learning Techniques in a Single Center, Non-randomized Controlled Clinical Trial

Lorena Casanova Lozano, Ramon Reig Bolaño, Sergi Grau Carrión and David Reifs Jiménez
University of Vic, Central University of Catalonia, Vic, Spain
E-mail: lorena.casanova@uvic.cat

Summary: The measurement of chronic wound size has clinical relevance for the evaluation of lesion evolution. Therefore, an objective, accurate and simple measurement is important for the optimal management of patients. Two methods based on Machine Learning (ML) algorithms have been developed to enable the automatic calculation of the area of skin lesions using a mobile device. One of the methods consists of using an external measurement reference based on an adhesive as a calibrator, and the other one uses a reference given by a capture of the three-dimensional space. This procedure is intended to be an objective, accurate and simple solution to this clinical need in the field of skin lesion management. This article describes the validation process of the new methods for measuring chronic wound area using imaging and ML algorithms. A clinical trial has been carried out in a hospital center in a controlled and non-randomized manner. The results demonstrate the efficacy of our method compared to traditional methods.

Keywords: Chronic wound, Machine learning, Area measurement, Images, Intraclass correlation index.

1. Introduction

1.1. Background

A wound appears when the skin tissue breaks down. This begins a process of regeneration of the damaged skin known as cicatrization, which can extend from hours to years, or may not occur at all. A wound is classified as a chronic wound when this repair time is very long or does not follow an orderly evolution. In contrast, acute wounds heal gradually, in a manner appropriate to the size and type of wound, usually within a short period of time [1]. The most frequently treated lesions are vascular ulcers (venous and arterial), diabetic foot ulcers and pressure ulcers [2].

In Europe, an estimated 1.5 to 2 million people suffer from acute or chronic wounds [3]. These types of skin injuries are treated in hospitals or in community settings, such as primary health centers or in private homes with the visit of a nurse. Wounds pose an urgent clinical challenge due to the great impact they have on both the patient and the healthcare system. On the one hand, patients' quality of life is markedly affected as a consequence of the physical, cognitive and social effects of skin injuries and their treatment [3]. On the other hand, skin injuries have a large impact on healthcare costs due to their high prevalence, recurrence and diversity [4], the time spent by nurses and other healthcare professionals, and the healthcare costs resulting from frequent dressing changes and potential wound complications [3]. In addition, wounds, influenced by factors such as population aging, diabetes, obesity, or bacterial resistance to antibiotics (persistence of infections), are expected to remain a major clinical, social, and economic concern in the coming years [5].

1.2. State of the Art

The measurement of the wound size is part of the phases of the management of subjects with skin lesions, from the initial assessment and classification of the wound as well as the selection of therapeutic strategy, to the evaluation of the evolution of the wound [6]. According to [7, 8], skin injury measurement has clinical relevance to know the healing status. Furthermore, measuring the surface area of a skin lesion is part of the recommendations of national and international clinical guidelines for wound care and management [9, 10]. The framework for wound assessment in clinical practise is the so-called "Wound Assessment Triangle" recommended by the World Union of Wound Healing Societies [11]. The Assessment Triangle is based on three aspects: the wound bed, the wound edge, and the perilesional skin; where the wound bed comprises, among other clinical signs, the measurement of wound size. Then, wound size is part of the parameters that should be recorded during wound assessment, whether using the "Wound Assessment Triangle" framework or other validated wound evolution assessment scales, such as the PUSH 3.0 (Pressure Ulcer Scale for Healing) or the RESVECH 2.0 (Expected Results of the Assessment and Evolution of Chronic Wound Healing Index) [12]. Therefore, an objective, accurate and simple measurement of wounds is important for the optimal management of subjects with this type of pathology.

Then, the measurement of chronic wound size has clinical relevance for the evaluation of lesion evolution. Therefore, an objective, accurate and simple measurement is important for the optimal management of patients. Today, the vast majority of skin lesion measurement methods in use have limitations; ruler measurement overestimates the area with a lack of precision in the measurement of non-rectangular or irregular lesions, planimetry with transparent acetate is an invasive technique and adobe photoshop planimetry is sensitive to illumination and size of the skin lesion. However, new methods using portable systems such as mobile phones and 3D technology can be used to make such measurements. This demonstrates how modern medical imaging combined with advanced telecommunications can provide better health care and diagnosis, improve patient handling, reduce health service costs, and save lives effectively everywhere in the world [13].

2. Objectives

The main objective is to compare the concordance, repeatability and reproducibility of the results of the calculation of the area of skin lesions between three measurements methods: rule measurement, digital planimetry and the developed algorithms. The secondary objective is to evaluate the usability for the calculation of the area using a mobile device with the software-implemented algorithms.

In order to accomplish this, two methods based on Machine Learning (ML) algorithms have been developed to enable the automatic calculation of the area of skin lesions using a mobile device. One of the methods consists of using an external measurement reference based on an adhesive as a calibrator, and the other one uses a reference given by a capture of the three-dimensional space. This procedure is intended to be an objective, accurate and simple solution to this clinical need in the field of skin lesion management.

3. Methods

3.1. Design, Settings and Participants

A prospective, single-center, non-randomized, pre-marketing clinical investigation has been conducted with one arm of subjects to collect skin lesion area for comparison of the results obtained between three methods of measurement. There has been no follow-up of patients within the clinical investigation. After informed consent, total of 67 wound images were obtained from 41 patients (>= 18 years) between November 22, 2022 and February 10, 2023. For inclusion in the study, skin lesions must not be neoplastic, tumorous or precancerous, nor present cutaneous carcinoma or other skin lesions of confirmed malignancy or with excessive exudate that may hide part of the wound and its contour. Also, the calibrator was required to be visible in picture and the wound must be measured with a 15 cm ruler.

3.2. Sample Collection

The study requires a single visit to the center by the subject. The visit consisted of preparing the subject by placing him/her in a suitable and comfortable position, covering the sensitive areas with a cloth and cleaning the skin lesion with physiological saline solution. The room was conditioned with adequate lighting. Since the position of the light is a key factor in the quality of the image, in order to eliminate shadows on the area to be photographed, the image was not taken under a direct light source. The flash has been avoided and a frontal focus of the light has been achieved.

Once the patient has given informed consent, the measurements of the wound area were performed by different trained nurses using the application installed on two different iPhone device models and on an iPad, resulting in three images (photo A, photo B and photo C) to check, apart from the agreement of the area measurement result with the software, the reproducibility and repeatability. First, the area was measured using a ruler and the Kundin method [14]. Following, the researchers placed an external calibrator in the same plane as the skin lesion on intact skin at a minimum distance of 2 cm from the patient's skin lesion and, prior to taking the photograph, the researcher selected two reference points on the screen of the mobile device so that both the 2D and 3D versions of software would have a reference for calculation. Then, three images of the same wound were taken with each device using the application which automatically calculated the area (Fig. 1).

Fig. 1. Area measurement with application.

3.3. Main Outcomes and Measures

The evaluation criteria are the comparison of the concordance of skin lesion area calculation results

between the three measurement methods: ruler measurement, digital planimetry and the developed algorithm. At the same time, evaluate the software reproducibility with the calculation of skin lesion area between two different mobile devices. Also, evaluate the software repeatability with the calculation of skin lesion area between two consecutive measurements performed with the same device and by the same investigator.

Finally, assess the software usability by means of the completion of a usability questionnaire by the researcher, as well as the recording of the time required to complete the calculation of the skin lesion with each method used.

4. Results

The agreement between the two modes of operation of ML algorithm and digital planimetry is high with ICC values of 0.989 for 3D method and 0.993 for 2D method. In addition, the agreement between ML algorithm 2D and 3D calibration methods is associated with an ICC of 0.996, representing a high consistency between both modes of operation. The algorithms also have a high repeatability reflected in the ICC obtained, with 0.946 for 3D mode, and 0.971 for 2D mode. In addition, the reproducibility between different mobile devices is also high, with an ICC of 0.978 for both 3D and 2D methods. The usability is also highly elevated. According to the questionnaires conducted by the research team at the end of the study, the researchers would use the implemented software on a regular basis in clinical practice and consider it to be a fast, convenient, and simple method with greater ease of access to information than alternative methods currently available. In addition, it is significantly faster than digital planimetry (1.37 minutes versus 1.83 minutes; P = 0.0033). With respect to safety, no adverse events were reported during the study and no unanticipated risks were detected.

5. Conclusions

The conducted pre-market clinical investigation has demonstrated that the software-implemented algorithms are safe for both the intended user and the intended target population, and it meets the manufacturer's intended use: automated calculation of skin lesion area using a mobile device. The study has validated the performance, safety, and usability of the implemented method as a valuable tool in the measurement of skin lesions.

6. References

[1]. D. Marijanović, D. Filko, A systematic overview of recent methods for non-contact chronic wound analysis, *Appl. Sci.*, Vol. 10, Issue 21, 2020, pp. 1-28.

[2]. H. González de la Torre, J. Soldevilla-Ágreda, J. Verdú-Soriano, Clinical practice, *Wounds Int.*, Vol. 8, Issue 4, 2017.

[3]. C. Lindholm, R. Searle, Wound management for the 21st century: combining effectiveness and efficiency, *Int. Wound J.*, Vol. 13, 2016, pp. 5-15.

[4]. H. N. Wilkinson, M. J. Hardman, Wound healing: cellular mechanisms and pathological outcomes, *Open Biol.*, 2020, http://dx.doi.org/10.1098/rsob.200223.

[5]. C. K. Sen, Human wound and its burden: updated 2020 compendium of estimates, *Adv. Wound Care*, Vol. 10, Issue 5, 2021, pp. 281-292.

[6]. M. Á. Blasco Vera, L. Aunés García, P. Blanes Ortí, I. Ramos Romero, A. Hernández Sanfelix, Sistemas de medición de heridas, *Rev. Enfermería Vasc.*, Vol. 2, Issue 4, 2019, pp. 17-21.

[7]. G. Gethin, The importance of continuous wound measuring, *Wounds UK*, Vol. 2, Issue 2, 2006, pp. 60-68.

[8]. E. Nicholas, Wound Assessment Part -I: How to measure a wound, *Wound Essentials*, Vol. 10, Issue 2, 2015, pp. 51-55.

[9]. A. G.-P. Mohíno, *et al.*, Guia de prevención y manejo de úlceras por presión y heridas crónicas, Issue 1, 2004, pp. 1-14.

[10]. M. Wynne, National Wound Management Guidelines, *Health Service Executive*, 2018.

[11]. C. Dowsett, B. von Hallern, The triangle of wound assessment: a holistic framework from wound assessment to management goals and treatments, *Wounds Int.*, Vol. 8, Issue 4, 2017, pp. 34-39.

[12]. E. A. Ruiz, J. Rueda López, J. M. Sánchez Vicente, Actualización en la validez de las escalas de evaluación de la evolución de heridas, *Heridas y Cicatrización*, Vol. 11, 2021, pp. 15-21.

[13]. G. Sakas, Trends in medical imaging: From 2D to 3D, *Comput. Graph.*, Vol. 26, Issue 4, 2002, pp. 577-587.

[14]. J. C. Restrepo-Medrano, J. Verdú, Measure healing in pressure ulcers. What do we have?, *Gerokomos*, Vol. 22, Issue 1, 2011, pp. 35-42.

(043)

A General Framework for Reliability Assurance of Machine Learning-based Driving Functions in Powertrain Software

M. Chehoudi [1], I. Moisidis [1] and S. Peters [2]
[1] Mercedes-Benz Group AG, Béla-Barényi-Straße, 71059 Sindelfingen, Germany
[2] Institute of Automotive Engineering (FZD), TU Darmstadt, Otto-Berndt-Str. 2,
64287 Darmstadt, Germany
E-mails: moatez.chehoudi@mercedes-benz.com, ioannis.moisidis@mercedes-benz.com,
steven.peters@tu-darmstadt.de

Summary: Reliability assurance of vehicle software components faces new challenges in the era of Machine Learning (ML). The different development paradigms and the black-box nature of ML based models result in several unexpected insufficiencies during vehicle operation. In this paper, we propose a reliability assurance framework to mitigate these insufficiencies within a vehicle powertrain function. The framework is intended to maintain a consistent model performance by identifying inaccurate predictions at runtime. Furthermore, it aims to enhance the model performance iteratively by leveraging unreliable predictions for model retraining. We also provide a summary of the research methodology that is followed to develop and evaluate the framework. In the main section, we present state-of-the-art methods, which possess the potential to be further considered. Finally, we discuss the limitations of these approaches and the main research gaps. In conclusion, we observe a strong focus on automated driving while ignoring other important vehicle components that also include ML based modules. Furthermore, a lack of awareness is identified regarding the applicability of these approaches in real-time applications.

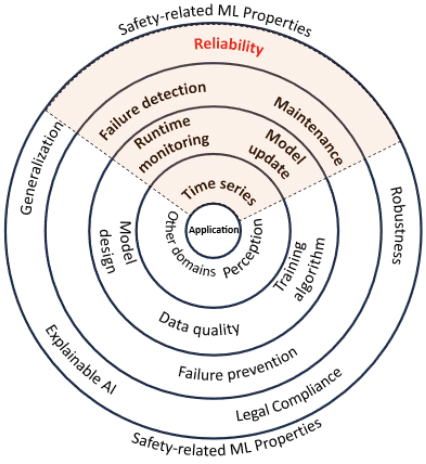
Keywords: Reliability assurance, Machine learning, Deep learning, Powertrain software, Runtime monitoring.

1. Introduction

Machine Learning is increasingly becoming a key instrument in developing automotive software. However, deploying ML-based software components is still restricted to a few applications due to safety and reliability concerns [1]. For many years, the software development process has followed the V-Model and standards such as ISO 26262 [2]. Given that ML models acquire the desired system behaviour from complex and high-dimensional data, standard verification and monitoring procedures do not apply to ML-based software [3]. Standard performance metrics in ML allow only an aggregated evaluation of the model performance, leaving large errors on individual predictions undetected [5]. Additionally, these measures are only possible in the presence of ground truth labels. Since these labels are often unavailable during operation, appropriate monitoring methods must be developed to assess the reliability of individual model predictions at runtime.

Recently, considerable research work has been achieved to address the safety challenges of ML-based software. Many aspects should be considered when developing ML models for safety-related applications. Fig. 1 provides a high-level illustration of the mainstream challenges in ML-safety. For instance, numerous studies have been conducted to create novel approaches for testing and validating ML models. Explainable Artificial Intelligence (XAI) is a significant area of research where scientists are working to create explanation approaches for how learning systems make decisions. Furthermore, new initiatives are under development, such as ISO PAS 8800, focusing on safe AI for road vehicles [32]. It aims to provide specific guidelines and recommendations for handling certification-related concerns with AI-based systems. Safety, robustness, transparency, and reliability are among the primary concerns. To meet the requirements outlined in similar standards, technical solutions must be developed.

Fig. 1. Research outline.



The research accomplished in the automotive sector so far has primarily focused on the safety of automated driving functions, e.g., [31]. Developing safe and reliable ML-applications within other vehicle software components such as powertrain still needs to be addressed [1]. Powertrain software functions are based on time series sensor signals. Unlike data in the

image domain, time series data exhibit temporal dependencies that introduce additional challenges to the modeling and validation process. Since prior approaches from the image domain are inappropriate, new reliability assurance methods must be developed [6].

This paper represents an initial comprehensive study in the automobile industry that exclusively focuses on powertrain-related ML reliability. The aim of this study is twofold: (1) to contribute to knowledge by identifying significant research gaps; (2) to explain why ML in powertrain software requires a distinct approach.

2. Research Outline

2.1. Research Objectives

Reliability in ML has emerged alongside other safety related aspects as a crucial concern, particularly in safety-relevant applications. Thus, we aim in this work to address the reliability assurance of ML from an industrial standpoint. Reliability refers to the consistency of model performance under different circumstances [8]. [9] proposed to classify reliability assurance measures in ML into: (1) failure prevention; (2) failure identification; and (3) maintenance.

Failure prevention can be addressed before and during model development, for example, by a good model design, a suitable algorithm, and adequate data collection. In contrast, failure identification requires runtime monitoring procedures, which assess the reliability of individual predictions (pointwise). The last measure in the reliability assurance principles proposed by [9] is the model update, which is referred to as maintenance. In this work, we aim to address the reliability of an already-trained ML model. This step is crucial to ensure the applicability of our approach on other existing use cases in the powertrain domain without any restrictions or assumptions about the model development process. Therefore, our research focuses on developing measures that align with the failure identification and maintenance principles and take into account the specific requirements within powertrain domain. Our research aims to answer the following research questions:

- Research Question 1: What approaches allow for the identification of unreliable individual model predictions at runtime?
- Research Question 2: Can the model performance be enhanced through an additional collection of instances with unreliable predictions?

Motivated by these research questions and the reliability assurance principles, we propose in Fig. 4 a general and scalable framework designed as an online monitoring system for ML-based driving functions within a vehicle control unit. The framework consists of a reliability estimation module, which processes the input signals of the model and computes a reliability score at inference time. Consequently, it identifies unreliable model predictions during operation and prevents their transmission to the target system promptly. As a result, this approach guarantees that the model operates within a restricted domain. Input signals that lead to a low reliability score are stored for further analysis and model retraining. The framework can also serve as a tool for targeted data acquisition during test drives to iteratively improve the model performance throughout the development process.

2.2. Methodology

We follow the research process in Fig. 2 to develop and evaluate the reliability assurance framework.

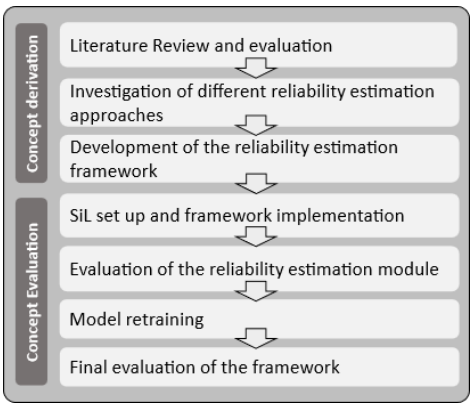


Fig. 2. Research methodology.

Concept derivation

The initial phase involves a comprehensive study of existing state-of-the-art methods. The goal is to assess current knowledge and practices related to ML reliability assessment, which align with our predefined criteria in Section 3.2. Based on our evaluation criteria, a specific method for reliability estimation is selected and adapted to fulfill the intended task in our framework. The next step involves setting up the software-in-the-loop environment and implementing the framework.

Concept evaluation

The reliability assurance framework is evaluated in a software-in-the-loop environment based on a real-world powertrain use case. The use case includes an LSTM, which processes multivariate time series for an early detection of a safety-related event in a vehicle. The LSTM model performs a binary classification task and predicts whether the event will occur or not. The evaluation of our approach involves running several simulations where the ML model operates within the designed framework. Our evaluation criteria are based on the error rate of the supervised ML model. For binary classification problems, a confusion matrix provides a good understanding of the model performance. As mentioned, reliability refers to maintaining the model performance under various conditions. The estimate of the true model

performance is always based on statistical assumptions about the data distributions during training and testing. Though, these assumptions usually don't apply to real world applications due to the infinite set of possible inputs [10]. Taking these facts into account, we define the acceptance criterion for our approach as the ability of the framework to maintain a similar model performance, which was evaluated during testing.

The second part of our framework involves a model retraining with new samples that are detected as unreliable. For this purpose, a triggering condition for the model retraining should be determined in advance. Furthermore, determining which unreliable predictions are informative and consequently enable to enhance the model performance is still an open question.

3. Related Work

This section provides an overview of state-of-the-art approaches for reliability estimation in ML. The structure of the review is illustrated in Fig. 3. Initially we clarify specific ambiguous terms, which often possess broad definitions in the literature. Then, we dive into the approaches for failure identification and pointwise monitoring in ML. On the application level, we narrow our research to cover methodologies relevant to the automotive sector and time series domain. Finally, we end up evaluating the state-of-the-art and selecting relevant approaches that are aligned with our predefined criteria.

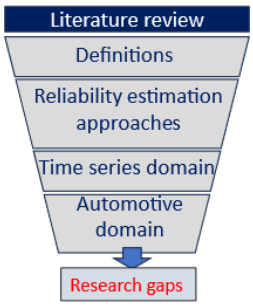


Fig. 3. Structure of the literature review.

3.1. Definitions

Reliability: refers to the ability of a model to perform as intended within specified performance limits and under various conditions [8, 9].

Maintenance: "The ease with which a software system or component can be modified to correct faults, improve performance or adapt to changed environments" [11].

Out-of-distribution (OOD): refers to test samples drawn from a distribution that is different from the training distribution. There is no general definition of the term distribution in the literature since it depends on the target application [12]. For example, in image classification problems, the distribution refers to the label distribution [12]. This paper defines a distribution as the spatial arrangement of the input data within the feature space. We also define an OOD sample as a test sample with an input far from this distribution.

Anomaly detection (AD): in contrast to OOD which is always associated with data unknown for the ML model, AD is a more general term and can be used for novelty / outlier detection in data, which are not explicitly used in ML context (e.g., fault diagnosis, cybersecurity).

3.2. Reliability Estimation Approaches

To develop the reliability assurance framework a targeted literature review has been carried out. This section provides a summarized overview of the relevant state-of-the-art approaches for detecting unreliable predictions in ML models. The selection process of the literature is based on the following criteria.

Inclusion Criteria:

- IN1: The paper addresses the main challenges related to the safety, reliability or trustworthiness in Machine and Deep Learning AND;
- IN2: The paper proposes a framework or an algorithm to assess the reliability of individual model predictions at runtime, AND;
- IN3: The paper evaluates the approach on a real-world application and discusses possible limitations.

Exclusion criteria:

- EX1: Approaches with specific assumptions about model structure, learning algorithm or data type.

Evaluation Criteria:

- EV1: Performance;
- EV2: Computational efficiency.

Reliability Assessment methods in the literature can be classified into:

- Uncertainty quantification (UQ);
- Out-of-distribution (OOD) detection.

Although UQ techniques may be used to detect OOD samples, we don't classify UQ as an explicit method for OOD detection for two reasons. First, these techniques aim to quantify the model uncertainty in the hole feature space, including in-domain space. Secondly, several investigations in [13-15] demonstrated empirically, that UQ often fails to detect OOD samples.

Uncertainty quantification in neural networks

Neural network outputs suffer from calibration issues that lead to overconfident predictions. To overcome this problem, multiple uncertainty quantification techniques have been addressed in the literature. These methods enable estimating the confidence of the neural network by computing a distribution over possible predictions instead of a single deterministic value [16]. Reliable uncertainty estimates and appropriate thresholds quantify when we can trust the model's predictions [17]. This paper

focuses on approaches that align with our predefined criteria and hold potential for an integration into our framework. For a detailed review of UQ approaches we refer to [16, 18].

Based on the source of uncertainty, a distinction is made between aleatoric and epistemic uncertainty [18]. Aleatoric uncertainty is irreducible since it is inherent to the underlying data properties (e.g., noise) [18, 19]. In contrast, epistemic uncertainty refers to the uncertainty in the model predictions, generally caused by a lack of knowledge. Therefore, reducing epistemic uncertainty can be achieved through increasing the amount of data [16].

Bayesian neural networks are among the most widely used techniques for estimating uncertainty. These networks are based on probabilistic models and are an extension of conventional networks. Due to their statistical nature, this type of networks is not based on deterministic parameters. Instead, a probability distribution over possible network parameters is acquired in the learning process to generate variance in the model predictions [16]. Monte Carlo dropout represents an approximation of the Bayesian methods that quantifies uncertainty without a parametric model. To generate a distribution, both approaches require multiple forward passes during runtime [7]. [17] introduced ensemble methods, which leverage multiple deterministic networks at inference by averaging the predictions to generate a probability distribution. The model uncertainty is then estimated based on the variance of the predictions [13, 17]. Single deterministic methods include, for example, temperature scaling or training an additional network for uncertainty estimation purpose. Test-time data augmentation is another method for estimation uncertainty, which is based on generating slightly different data for every single input data during run-time. Then, multiple inferences are executed using the original and the augmented data to compute a distribution over predictions.

Out-of-distribution detection

Since defining an in-domain boundary for a dataset in a high dimensional space is not straightforward, OOD-detection is still an open research problem [20]. One of the most straightforward metrics for OOD detection in ML is the Euclidean distance in the input space. The distance provides information about the dissimilarity between new test samples and the training samples. [4] implemented the Euclidean distance to determine, if the test sample is close to the training set. Additionally, the local model performance on these nearby data is leveraged as a reliability measure of the prediction. These criteria align with the density and local fit principles introduced in [21]. Other works tried to enhance the effectiveness of the distance metric, for example, through a feature decorrelation [22] or feature weighting by their respective importance in the model [23]. [24] introduced a trust score, which is based on the agreement between the original classifier and an additional nearest neighbour classifier. Since the Euclidean distance measure in the feature space does not consider internal representations of a deep neural network, [25] introduced the distance in the latent space as a novelty measure.

Another approach for OOD detection is training a classifier to learn the normal data distribution and using it to detect abnormal data. Training the classifier can be achieved for example by means of OCSVM, introduced by [26]. OCSVM is an unsupervised learning technique which applies the kernel trick and learns a decision boundary of a data distribution by estimating its density. During inference, new data points are classified into in-distribution or out-of-distribution. Another approach, introduced by [27], maps the input space into a hypersphere. The detection of abnormal behaviour is based on whether the new sample lies inside or outside the hypersphere [27].

OOD detection in time series

In contrast to image classification, where OOD detection methods have been excessively studied in the literature, time series data still need to be addressed. [6] proposed an algorithm for OOD detection based on deep generative models. [28] implemented statistical distance measures to detect unreliable predictions within a univariate time series forecasting model. Since only a few studies exist in the field of OOD detection in time series, we enlarged our scope of research to cover other domains such as natural language processing (NLP) and human activity recognition (HAR), which are also based on sequential data. The mainstream approach in this field is to use autoencoder or generative models (GAN) to distinguish normal from abnormal data [29]. This approach assumes, that an autoencoder can accurately reconstruct in-distribution samples. In contrast, abnormal samples are associated with a higher reconstruction cost [12]. OOD samples can be detected by comparing the reconstruction error to a predefined threshold.

Online monitoring in automotive applications

In recent years a few studies have been carried out to investigate monitoring approaches within automotive applications. [7] combined multiple monitoring approaches in a meta model to detect unreliable predictions in a traffic sign recognition model. The monitoring approaches were applied for different triggering conditions (e.g., OOD, adversarial samples). [30] implemented an online OOD detection method within a perception function for automated driving. The approach is based on reconstructing the image by using an autoencoder. The reconstruction error serves as a metric to detect OOD samples. [1] implemented OCSVM to detect OOD data in a time series regression model. Therefore, feature engineering on the time series was required to achieve a time independent representation of the data into a feature space.

3.3. Evaluation

In [16] an evaluation of several uncertainty estimation approaches based on different criteria, including computational effort during training and

inference was carried out. The research demonstrates that Bayesian methods and ensembles require the highest computational effort during training and inference. Additionally, ensemble methods require a high memory consumption during inference since several networks must be implemented in the target system. We notice that applying these techniques to sequence models (e.g., LSTM) can lead to a higher computational overhead because the entire sequence of data is needed for inference [19]. Since these UQ techniques align with our exclusion criterion EX1, they are excluded from further consideration. An exception is injection dropout, which does not require model retraining. Single deterministic techniques provide the lowest computational effort and memory consumption during training and inference, which makes them suitable for our framework. Similar to dropout and ensemble, test time data augmentation requires multiple forward passes for uncertainty quantification. However, there are no recommendations for a minimum required number of inferences to generate reliable uncertainty estimates. Therefore, we include it in our further consideration.

Distance metrics application on time series requires special tools, such as feature engineering or dynamic time warping. Furthermore, the training data must be stored in the vehicle during operation to compute the distances. The same problem applies to autoencoder and generative models, which require many parameters to be stored in the vehicle leading to a high memory consumption. We also mention that there is considerable disagreement in the literature about the performance of the reliability estimation methods. Furthermore, there are no unified quality evaluation metrics or reliability thresholds, making an objective performance evaluation of the different approaches impossible.

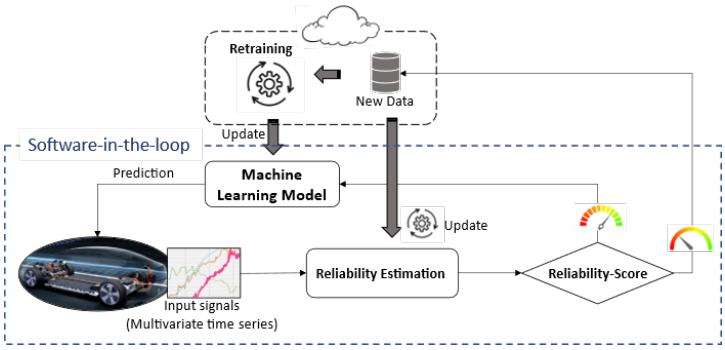


Fig. 4. Online Reliability Estimation Framework.

4. Conclusion

Our literature review shows a high awareness about the reliability challenges in ML, especially in safety-related applications or in healthcare. Our review reveals numerous publications in different domains dedicated to addressing ML insufficiencies. OOD detection and uncertainty estimation are the most common approaches for reliability estimation. However, the number of relevant studies drops significantly when taking into account the constraints inherent to powertrain software, such as computation time, complexity and data type compatibility. The literature in the automotive domain has tended to focus on perception functions for automated driving, leaving other ML applications in vehicle software uncovered. Consequently, there is a need for more knowledge about the effectiveness of the existing solutions for the powertrain domain. One of the biggest challenges is the fundamental contrast between the data types used in both domains. Time series data exhibit temporal dependencies that must be taken into account in the reliability assessment. These time dependencies differ completely from the spatial relationships of image pixels [6]. For example, in contrast to image data, it is more difficult to label OOD time series data manually. This leads to an additional complexity, if some OOD data are required for developing or evaluating the reliability estimation framework. Therefore, there is a need for new approaches, which consider these particularities. For example, distance metrics can be a suitable approach, if the data storage prerequisite and the temporal dependencies between different time steps are efficiently handled.

We also mention that the literature only covers the failure identification principle, discussed in Section 2.1. There are no measures of how to cope with these unreliable predictions once detected. During vehicle operation, it may be insufficient to detect unreliable model predictions, especially when other software modules rely on these predictions. Therefore, we address in our work the maintainability of the model, by leveraging unreliable predictions to retrain the model. Consequently, the amount of unreliable model predictions can be iteratively reduced to an accepted level. Except in the active learning domain, there is no guarantee of a correlation between unreliable predictions and model improvement. The literature recommends increasing the amount of data to reduce epistemic uncertainty. However, distinguishing epistemic uncertainty from other types of insufficiencies is not straightforward.

References

[1]. F. Korthals, M. Stöcker, S. Rinderknecht, Plausibility assessment and validation of deep learning algorithms software development, in *Proceedings of the International Stuttgarter Symposium*, 2021, pp. 91-105.

[2]. 0. Willers, S. Sudholt, S. Raafatnia, S. Abrecht, Safety concerns and mitigation approaches regarding the use of deep learning in safety-critical perception tasks, *arXiv Preprint*, 2020, arXiv:2001.08001.

[3]. D. Marijan, A. Gotlieb, M. Kumar Ahuja, Challenges of testing machine learning based systems, in *Proceedings of the IEEE International Conference on Artificial Intelligence Testing (AITest'19)*, Newark, CA, USA, 2019, pp. 101-102.

[4]. G. Nicora, M. Rios, A. Abu-Hanna, R. Bellazzi, Evaluating pointwise reliability of machine learning prediction, *Journal of Biomedical Informatics*, Vol. 127, 2022, 103996.

[5]. P. Schulam, S. Saria, Can you trust this prediction? in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS'19)*, Naha, Okinawa, Japan, 2019, pp. 1022-1031.

[6]. T. Belkhouja, Y. Yan, J. Rao Doppa, Out-of-distribution detection in time-series domain: a novel seasonal ratio scoring approach, *ACM Transactions on Intelligent Systems and Technology*, Vol. 15, Issue 1, 2023, 8.

[7]. L. Hacker, J. Seewig, Insufficiency-driven DNN error detection in the context of SOTIF on traffic sign recognition use case, *IEEE Open Journal of Intelligent Transportation Systems*, Vol. 4, 2023, pp. 58-70.

[8]. E. Haedecke, M. A. Pintz, Transparency and reliability assurance methods for safeguarding deep neural networks – a survey, in *Proceedings of the Workshop on Trustworthy Artificial Intelligence*, Grenoble, France, 2022.

[9]. S. Saria, A. Subbaswamy, Tutorial: Safe and reliable machine learning, *arXiv Preprint*, 2019, arXiv:1904.072042019.

[10]. R. Salay, R. Queiroz, K. Czarnecki, An analysis of ISO 26262: using machine learning safely in automotive software, *arXiv Preprint*, 2017, arXiv:1709.02435.

[11]. J. M. Faria, Machine Learning Safety: An Overview, *Safety-Critical Systems Club*, 2018.

[12]. J. Yang, K. Zhou, Y. Li, Z. Liu, Generalized out-of-distribution detection: a survey, *arXiv Preprint*, 2024, arXiv: 2110.11334.

[13]. A. Schwaiger, P. Sinhamahapatra, J. Gansloser, K. Roscher, Is Uncertainty Quantification in Deep Learning Sufficient for Out-of-Distribution Detection? https://ceur-ws.org/Vol-2640/paper_18.pdf

[14]. Y. Ovadia, E. Fertig, J. Ren, Z. Nado, et al., Can you trust your model's uncertainty? *arXiv Preprint*, 2019, arXiv:1906.02530.

[15]. D. Ulmer, G. Cinà, Know your limits: uncertainty estimation with ReLU classifiers fails at reliable OOD detection, *arXiv Preprint*, 2021, arXiv:2012.05329.

[16]. J. Gawlikowski, C. Rovile Njieutcheu Tassi, M. Ali, J. Lee, et al., A survey of uncertainty in deep neural networks, *arXiv Preprint*, 2022, arXiv:2107.03342.

[17]. B. Lakshminarayan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, *arXiv Preprint*, 2017, arXiv:1612.01474.

[18]. M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, et al, A review of uncertainty quantification in deep learning: techniques, applications and challenges, *arXiv Preprint*, 2021, arXiv:2011.06225.

[19]. A. Loquercio, M. Segù, D. Scaramuzza, A general framework for uncertainty estimation in deep learning, *arXiv Preprint*, 2020, arXiv:1907.06890.

[20]. M. Tan, Y. Yu, H. Wang, D. Wang, et al., Out-of-domain detection for low-resource text classification tasks, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*, Hong Kong, China, 2019, pp. 3566-3572.

[21]. J. A. Leonard, M. A. Kramer, L. H. Ungar, A neural network architecture that computes its own reliability, *Computers & Chemical Engineering*, Vol. 16, Issue 9, 1992, pp. 819-835.

[22]. E. Askanazi, I. Grinberg, Distance-based analysis of machine learning prediction reliability for datasets in materials science and other fields, *arXiv Preprint*, 2023, arXiv:2304.01146.

[23]. H. Meyer, E. Pebesma, Predicting into unknown space? *arXiv Preprint*, 2020, arXiv:2005.07939.

[24]. H. Jiang, B. Kim, M. Y. Guan, M. Gupta. To trust or not to trust a classifier, *arXiv Preprint*, 2018, arXiv:1805.11783.

[25]. J. P. Janet, C. Duan, et al., A quantitative uncertainty metric controls error in neural network-driven chemical discovery, *Chem. Sci*., Vol. 10, 2019, pp. 7913-7922.

[26]. B. Schölkopf, R. C. Williamson, A. Smola, et al., Support vector method for novelty detection, *Advances in Neural Information Processing Systems*, Vol. 12, 1999.

[27]. L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, Deep one-class classification, in *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden, 2018, pp. 4393-4402.

[28]. M. N. Akram, A. Ambekar, I. Sorokos, K. Aslansefat, D. Schneider, StaDRe and StaDRo: Reliability and robustness estimation of ML-based forecasting using statistical distance measures, *arXiv Preprint*, 2022, arXiv:2206.11116.

[29]. S. Ryu, S. Koo, H. Yu, G. Geunbae Lee, Out-of-domain Detection based on Generative Adversarial Network, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels*, Belgium, 2018, pp. 714-718.

[30]. T. Sämann, H.-M. Groß, Online out-of-domain detection for automated driving, *arXiv Preprint*, 2021, arXiv:2310.14675.

[31]. KI-Absicherung, https://www.ki-absicherung-projekt.de/

[32]. ISO/CD PAS 8800, https://www.iso.org/standard/83303.html

(046)

EEG Decoding with Conditional Identification Information

Pengfei Sun, Jorg De Winne, Paul Devos and Dick Botteldooren
Department of Information Technology, WAVES Research Group, Ghent University, Gent, Belgium
E-mails: {pengfei.sun, jorg.dewinne, p.devos, dick.botteldooren}@ugent.be

Summary: Decoding EEG signals is crucial for unraveling human brain and advancing brain-computer interfaces. Traditional machine learning algorithms have been hindered by the high noise levels and inherent inter-person variations in EEG signals. Recent advances in deep neural networks (DNNs) have shown promise, owing to their advanced nonlinear modeling capabilities. However, DNN still faces challenge in decoding EEG samples of unseen individuals. To address this, this paper introduces a novel approach by incorporating the conditional identification information of each individual into the neural network, thereby enhancing model representation through the synergistic interaction of EEG and personal traits. We test our model on the WithMe dataset and demonstrated that the inclusion of these identifiers substantially boosts accuracy for both subjects in the training set and unseen subjects. This enhancement suggests promising potential for improving for EEG interpretability and understanding of relevant identification features.

Keywords: EEG, Neural network, Classification, Human-computer interfaces.

1. Introduction

The interplay between humans and artificial intelligence (AI) remains suboptimal, lacking the depth of engagement and synchrony inherent to human-to-human interactions. In pursuit of bridging this gap, there has been a marked shift towards leveraging neurophysiological insights, particularly through the prism of electroencephalography (EEG), to elucidate underlying cerebral mechanisms and refine the human-computer interface. The WithMe [1] experiment exemplifies this approach by presenting subjects with specific auditory and visual stimuli, thereby enabling the differentiation between target and distractor stimuli, whilst concurrently capturing the resultant EEG data. However, another challenge that arises is decoding the collected EEG signals, and in particular how to effectively decode and analyze the data to extract meaningful information.

Recently, advancements in machine learning have shown notable advantages in extracting intricate information from EEG signals [2, 3]. Among these techniques, Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) stand out. CNNs process EEG signals as frames, synthesizing this data to make final decisions. RNNs, in contrast, retain information from previous inputs, showcasing an ability to recognize and remember temporal sequences, which is crucial for tasks needing long short-term memory of past events. Initial explorations employing deep neural networks (DNN) and conventional machine learning paradigms have yielded promising outcomes by directly processing EEG signals from WithMe experiment to classify the target/distractor [4]. The majority of neural network solutions for EEG decoding utilize fully supervised learning methods, meaning they refine their parameters based on hard-labeled data. However, this method tends to create models that are highly specialized for the tasks they're trained on, which may not perform well on different tasks or with new individuals [5]. In addition, the heterogeneity in individual brain activity patterns poses a significant challenge to the current deep learning frameworks, particularly in decoding EEG signals from subjects not represented in the training corpus.

Notwithstanding these challenges, the WithMe experiment has unveiled certain individual characteristics, notably sensory dominance [6], that substantially influence experimental outcomes. For instance, participants with auditory dominance exhibited superior performance across various metrics and conditions compared to their visually dominant peers [1]. This observation prompts a reevaluation of the role individual-specific traits play in modulating EEG signals in an attention and working memory task. It raises the intriguing possibility that integrating a compendium of these personal attributes into computational models could potentially enhance their representational capacity. By decoding the latent interplay between personal traits and EEG patterns, this research aspires to not only bolster decoding accuracy for familiar subjects but also extend predictive proficiency to novel individuals.

To tackle this issue and recognize that personal characteristics can impact experiment results, we propose a novel framework that incorporates conditional identification information into the EEG decoding process. Thus, a network employing fully supervised learning can utilize not only the hard label information but also the conditional identification information. This paper is dedicated to investigating the viability of this innovative methodology, with the ultimate aim of advancing human-AI interactions.

2. Overview

2.1. Overview of Framework

Fig. 1 depicts the structure of our proposed framework, comprising two main components: (1) Embedding the conditional identification information, employing a 16-neuron embedding layer designed to transform conditional identification information. Alongside, the pre-convolution layer, functioning as an identity layer in this study, encodes EEG data. (2) Decoding the integrated features, where this section is capable of utilizing various renowned neural network models to distinguish effectively between target and distractor stimuli.

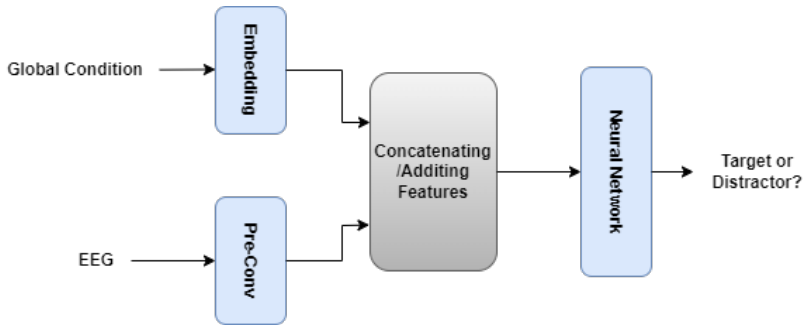


Fig.1. Overview of the Proposed Framework.

2.1. Conditional Identification Information

The conditional identification information of each individual is utilized for target/distractor classification. This auxiliary conditioner employs an embedding layer to encode the identification attributes of each subject, transforming them into a comprehensive subject embedding. These embeddings, along with the EEG patterns, are then synergistically fused and introduced into the neural network. Through this extension, we expect to enhance the model's capability to learn a more generalized representation across diverse individuals, more precisely accounting for the variability in their brain activity characteristics. In this paper, we choose four distinct variables to examine their influence on the outcomes: 'Auditive/Visual Dominance', 'Sex', 'Music Education', and 'Active musician'. The former is assessed with the experiment proposed in [6], the others can be obtained via a simple questionnaire.

3. Experiment and Results

3.1. Dataset

Our model was trained and evaluated using the dataset from the WithMe experiment [1]. This experiment presented target and distractor digits to the subject tasking them to remember and rename the targets. Four conditions were tested: simple sequence of visual stimuli, rhythmic presentation of targets, simple presentation supporting targets with a short beep, rhythmic presentation supported by beeps. The dataset encompasses data from a total of 42 participants. For training and internal testing, we randomly selected 38 participants further referred to as Within-subjects. The remaining 4 participants' data were reserved to assess the generalizability of the models and are referred to as Unseen-subjects. Specifically, we partitioned the WithMe data into a training set and two testing sets: Within-subjects, which comprises 18,176 training instances and 4580 testing instances, and Unseen-subjects, which includes 2400 testing instances. Preprocessing of the EEG data involved re-referencing each channel to the average activity of the mastoid electrodes. The data were then band-pass filtered between 1 and 30 Hz and subsequently downsampled to 64 Hz. Then, the data were segmented into 1.2 s epochs based on trigger events, with the final preprocessing step normalizing the EEG channel data to ensure zero mean and unit variance for each sample.

The data and code can be accessed via https://github.com/sunpengfei1122/Withme-EEG-dataset.

3.2. Implementation Detail

In our experiment, we use the Adam optimizer to optimize the weights with a constant learning rate of 0.0001 and a minibatch size of 128. The EEGNet architecture features convolutional layers, starting with 16 kernels for initial temporal and spatial feature extraction from EEG signals. This is followed by depthwise and separable convolutions using 32 and 64 kernels, respectively, for efficient feature learning. For the LSTM [7] and DMU [8] models, a single recurrent unit with 64 neurons is utilized. Specifically, for the DMU's delay gate, the total number of delays is set to 20, considering the short duration of each sample. These models are developed within the PyTorch framework, adhering to default training methodologies. All modules are trained and updated in an end-to-end manner.

3.3. Results

Table 1 delineates the performance of three baseline models (EEGNet, LSTM, and DMU) and their counterparts incorporating our conditional identification (IDs) information branch of each participant. Remarkably, the EEGNet model, when enriched with conditional information, exhibits substantial enhancements in performance in both within-subject and unseen-subject. Furthermore, the addition of conditional IDs to LSTM and DMU models also yields marked improvements, particularly in the recognition of unseen subjects, indicating that the network has acquired more generalized representation of EEG. Additionally, t-distributed Stochastic Neighbor Embedding (t-SNE) [9] visualizations across all individuals of the conditional identification embedding layer in Fig. 2 reveal a tendency for unseen subjects (person 38 to 41) to gravitate towards familiar centroids. Intriguingly, while up to 14 cluster centers (according to experimental data statistics) are theoretically possible given the 4-dimensional input IDs, only 7 prominent clusters emerge, suggesting that not all features exert a significant influence on the model's performance.

Table 1. The results of three models on WithMe dataset is presented and compared to the models with the global condition id information.

Datasets	***Models***	**Within Accuracy**	**Unseen Accuracy**
WithMe	EEGNet	81.67 %	76.42 %
	+ IDs	**86.29 %**	**79.08 %**
	LSTM	80.09 %	74.00 %
	+ IDs	**81.18 %**	**76.00 %**
	DMU	81.94 %	75.92 %
	+ IDs	**82.21 %**	**77.21 %**

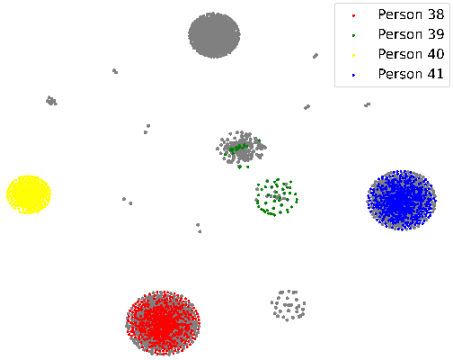


Fig. 2. Illustration of the intrinsic clustering pattern of identification information after embedding layer, as unveiled by t-SNE. The grey clustering pattern represents all the trained subjects, while the colorful pattern denotes the unseen subjects.

3.4. Analysis

To further validate our proposed framework, we focus on a key personal trait: Dominance. The WithMe study [1] demonstrated that participants with auditory dominance outperformed visually dominant individuals in all metrics and scenarios. As illustrated in Fig. 3, this distinction is shown as two distinct clusters based on dominance type. Subsequently, we evaluate EEGNet in two contexts: Auditory vs. Visual dominance for EEG classification. Fig. 4 reveals that, for the vanilla EEGNet, visually dominant individuals slightly outperform their auditory counterparts in within-subject tests but fare worse in unseen situations. However, when incorporating IDs information, the auditory group excels in both scenarios, consistent with our experimental findings. This observation may suggest that participants who performed better in the experiment tended to have clearer representations in their EEG signals that can be recognized by neural networks.

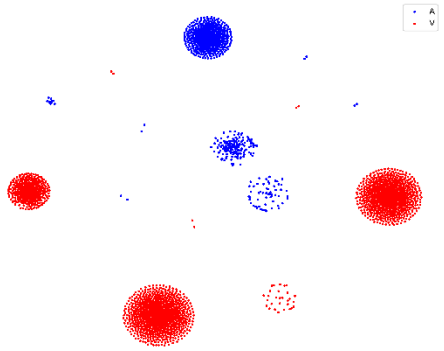


Fig. 3. Illustration of the intrinsic clustering pattern of Audio/Visual (A/V) dominance.

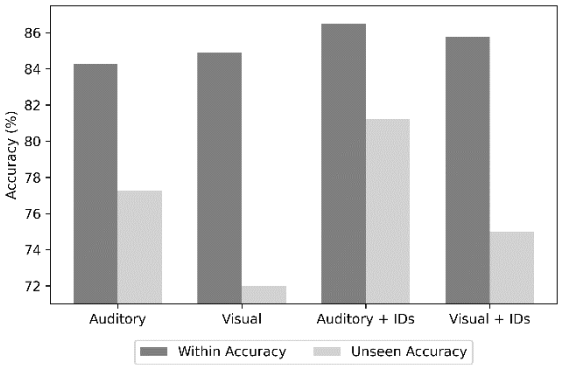


Fig. 4. Classification Performance on the WithMe Dataset based on Auditory/Visual dominance.

4. Conclusions

In this paper, we investigate the effectiveness of incorporating additional conditional identification information into neural network architectures for the classification of target versus distractor stimuli based on EEG. Through the deployment of an auxiliary global conditioner that utilizes an embedding layer to capture unique individual traits, our methodology not only enhances the model's precision in the same subjects but also amplifies its generalizability to

unseen subjects, adeptly navigating the variety of neural responses observed in diverse individuals. Our results suggest that incorporating a personalized and context-aware conditioner is a promising approach to enhance the performance and reliability of EEG classification in real-world scenarios.

Acknowledgements

This work was supported in part by the Flemish Government under the "Onderzoeksprogramma Artificiele Intelligentie (AI) Vlaanderen" and the Research Foundation - Flanders under grant number G0A0220N (FWO WithMe project).

References

[1]. J. De Winne, P. Devos, M. Leman, D. Botteldooren, With no attention specifically directed to it, rhythmic sound does not automatically facilitate visual task performance, *Frontiers in Psychology*, Vol. 13, 2022, 894366.

[2]. S. Cai, R. Zhang, M. Zhang, J. Wu, H. Li, EEG-based auditory attention detection with spiking graph convolutional network, *IEEE Transactions on Cognitive and Developmental Systems*, 2024, pp. 1-9.

[3]. S. Cai, P. Li, H. Li, A bio-inspired spiking attentional neural network for attentional selection in the listening brain, *IEEE Transactions on Neural Networks and Learning Systems*, 2023, pp. 1-11.

[4]. S. Mortier, et al., Classification of targets and distractors in an audiovisual attention task based on electroencephalography, *Sensors*, Vol. 23, Issue 23, 2023, 9588.

[5]. P. Sarkar, A. Etemad, Self-supervised ECG representation learning for emotion recognition. *IEEE Transactions on Affective Computing*, Vol. 13, Issue 3, 2020, pp. 1541-1554.

[6]. M. H. Giard, F. Peronnet, Auditory-visual integration during multimodal object recognition in humans: a behavioral and electrophysiological study, *Journal of Cognitive Neuroscience*, Vol. 11, Issue 5, 1999, pp. 473-490.

[7]. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Computation*, Vol. 9, Issue 8, 1997, pp. 1735-1780.

[8]. P. Sun, J. Wu, M. Zhang, P. Devos, D. Botteldooren, Delayed memory unit: modelling temporal dependency through delay gate, *arXiv Preprint,* 2023, arXiv:2310.14982.

[9]. L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, *Journal of Machine Learning Research*, Vol. 9, Issue 11, 2008, pp. 2579-2605.

(047)

Research on Adaptive Differential Privacy Preservation Method Based on Blockchain and Federated Learning

Bing Wu and Haiyan Kang
School of Information Management, Beijing Information Science and Technology University,
Beijing 100192, China
Tel.: 13520416496
E-mail: kanghaiyan@126.com

Summary: Federated learning, as a mechanism that coordinates multiple participants to train models together without sharing local data, is naturally privacy-preserving for data. However, during the process of federated learning model training, there is still a risk that malicious attackers can leak the privacy of sensitive data by stealing intermediate parameters and inferring the original user data. To address the above problems, an Adaptive Differential Privacy Blockchain Federated Learning (ADP-BCFL) method is proposed to realize the compliant utilization of distributed data under the premise of ensuring privacy and security. Firstly, utilizing blockchain to achieve secure storage and efficient querying of user summary data. Secondly, an adaptive differential privacy mechanism is proposed and designed, which acts in the process of federal learning parameter passing, adaptively adjusts the threshold size of parameter tailoring according to the parameter characteristics, controls the content of the introduced noise, and ensures a good global model accuracy while effectively solving the problem of inference attack. Finally, comparison experiments are conducted on MNIST and Fashion MNIST datasets to verify the effectiveness of the proposed method ADP-BCFL.

Keywords: Federated learning, Adaptive differential privacy, Privacy protection, Blockchain storage, Deep learning.

1. Introduction

In today's digital era, data has become the core support and basic elements for the breakthrough development of emerging technologies and sectors of the big data industry, such as artificial intelligence, cloud computing, mobile Internet, etc., which has greatly contributed to the rapid development of the digital economy. However, nowadays, the massive data distributed storage in the terminal equipment without sharing with the outside world led to the phenomenon of "data silos", data leakage and data theft and other data security incidents [1], as well as China's enactment of data security and privacy protection of the relevant legal documents, data security and privacy protection has gradually been the close attention of the state, enterprises, and individuals [2]. To a certain extent, this seriously restricts the centralized "sharing" of massive data, increases the difficulty of effective utilization of massive data, makes massive data unable to maximize the role of not being able to be effectively shared, and also restricts the breakthrough development of machine learning and other areas of deep learning.

The research on the fusion of blockchain and federated learning for co-construction of models has attracted extensive attention from both academia and industry [3-5]. For example, Fang et al. [6] proposed an edge computing privacy protection method based on blockchain and federated learning, which can detect malicious devices along with a certain percentage of poisoning attacks, and greatly improves the security of the federated learning training process. During the process of federated learning model training, there is a risk of privacy leakage due to privacy attacks [7], and thus should be combined with appropriate privacy protection methods. Lu et al. [8] applied the local differential privacy technique to blockchain federated learning to solve the problem of data security privacy protection in industrial internet by adding noise to the original data.

The above related researches have made important breakthroughs in data utilization and data privacy protection, but at the same time, there are still three urgent problems to be solved, which are (1) the problem of inefficient and unmanageable information retrieval from distributed data endpoints in federated learning, (2) the problems of inefficient training and high communication cost in the use of encryption methods, and (3) the problem of low training model accuracy due to the inclusion of inappropriate noise content in the use of perturbation methods. Through in-depth research on the above issues, the main contributions of this paper are as follows.

(1) The decentralized storage of simple summary information such as personal information of each local user and data information of the dataset is achieved with the help of blockchain [9], which improves the retrieval efficiency of relevant information and shortens the training time of the model.

(2) Propose a federated learning data sharing method based on adaptive differential privacy mechanism, i.e., Adaptive Differential Privacy Blockchain Federated Learning (ADP-BCFL), to solve the privacy leakage problem in the process of federated learning model training parameter transfer by utilizing differential privacy technique.

(3) Propose and design an adaptive gradient trimming mechanism to adaptively adjust the size of the trimming threshold using the ADAM algorithm to

effectively control the content of additive noise, reduce the impact of noise on model accuracy, and ultimately achieve the purpose of minimizing the loss of model privacy.

(4) Comparative experiments are conducted on MNIST and Fashion MNIST real datasets in terms of three aspects, namely, training accuracy, performance loss and time cost, respectively, to evaluate the effectiveness of the proposed method and its superiority over other algorithms.

2. ADP-BCFL Method Design

This paper proposes and designs a blockchain-oriented adaptive differential privacy mechanism-based privacy protection method for federated learning data (ADP-BCFL) in conjunction with the blockchain technology to realize the effective processing and compliant utilization of multi-party user data. The method adopts the core idea of combining "on-chain and off-chain" to construct, and its architecture is shown in Fig. 1, which mainly contains the following two methods, namely, (1) data storage and processing method, and (2) adaptive gradient trimming method.

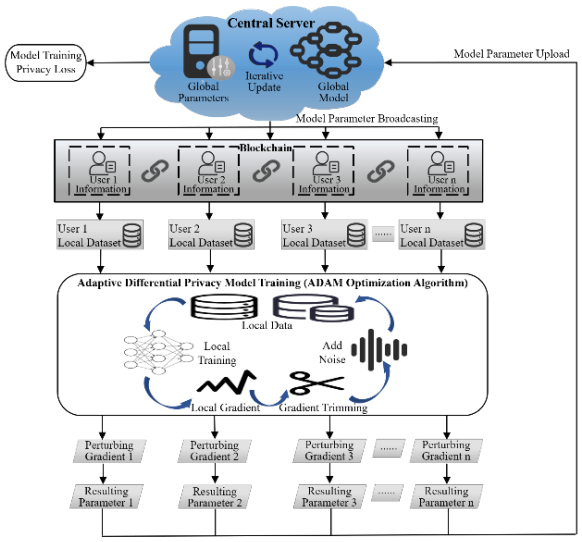


Fig. 1. Adaptive differential privacy based data sharing model for federated learning.

2.1. Data Storage and Processing Method

The data storage of ADP-BCFL method in this system mainly includes the following two parts, which are (1) The storage of simple data summary information on the blockchain, including local users' personal information, the type and size of dataset data, and other summary information, ensures the secure storage and efficient retrieval of uplinked data. And (2) Storage of user's local data, i.e., the original data used for model training is always stored locally in each user terminal, which is directly stored and managed by the data holders, providing a basic guarantee for the privacy and security of the data.

The ADP-BCFL approach to data processing in this system consists of the following three main parts, which are (1) the client uses the local data for model training and adds adaptive differential privacy noise to the process parameters, (2) the client uploads the trained local model updates to the server for aggregation, and (3) the server downlinks the aggregated updated global model parameters to the client.

2.2 Adaptive Gradient Trimming Method

For the gradient cropping according to a fixed global threshold C during the training process of the federated learning model, there may be a problem of adding excessive noise due to setting too large a threshold, or over-cropping the gradient due to setting too small a threshold. This paper combines the idea of adaptive moment estimation of ADAM stochastic optimization method, and proposes an adaptive differential privacy mechanism, which calculates the adaptive learning rate of different parameters through the estimation of first-order moments and second-order moments of the gradient. The ADAM optimizer assigns the weights with the corresponding learning rate according to their own characteristics, so as to enable each parameter to achieve faster and further movement to the maximum extent, and then the model achieves a faster convergence rate. The updating formula for the gradient is

$$\begin{aligned} E[\mathrm{g}]_t &= \beta_1 * E[\mathrm{g}]_{t-1} + (1-\beta_1) * \mathrm{g}_t, \\ E[\mathrm{g}^2]_t &= \beta_2 * E[\mathrm{g}^2]_{t-1} + (1-\beta_2) * (\mathrm{g}_t)^2, \end{aligned} \tag{1}$$

where $E[\mathrm{g}]_t$ is the cumulative gradient, $E[\mathrm{g}^2]_t$ is the cumulative square of the gradient, β_1 and β_2 are the smoothing constants (i.e., decay rates) used to smooth $E[\mathrm{g}]_t$ and $E[\mathrm{g}^2]_t$, respectively, $E^*[\mathrm{g}]_t$ is the bias-corrected cumulative gradient, $E^*[\mathrm{g}^2]_t$ is the cumulative square of the bias-corrected gradient, θ_t stands for the model parameters at the t round of training, lr denotes the learning rate, and ε_0 is the smoothing term used to prevent the divisor from going to 0, which is typically set to 10^{-8}. The optimizer's optimization process is top-down and top-up gradient, and the historical gradient derived from the already trained can be used to estimate the value of the gradient in the current round. Therefore, $E[\mathrm{g}^2]_{t-1}$ in the ADAM optimization algorithm can be regarded as an updated benchmark for the current gradient.

With the help of ADAM algorithm idea, according to the different situations of each round in the training process, the global gradient of the current round is predicted by combining the updating benchmark of the gradient, and the size of the threshold is flexibly adjusted, so as to determine the gradient trimming threshold C_t of the current round. η is the local

trimming factor, and the updating benchmark of the gradient $E[\hat{g}^2]_{t-1}$ is calculated as

$$E[\hat{g}^2]_0 = \vec{0},$$
$$E[\hat{g}^2]_{t-1} = \beta_2 * E[\hat{g}^2]_{t-2} + (1-\beta_2) * (\hat{g}_{t-1})^2 \quad (2)$$

Since initializing the square of the accumulated gradient $E[\hat{g}^2]_0$ to 0 will result in $C_1 = \eta \frac{\sqrt{E[\hat{g}^2]_0}}{1-(\beta_2)^0} = 0$, it cannot be used for gradient trimming. Therefore, a priori threshold G: when the cumulative square of the gradient at the beginning of training is insufficient (i.e., $E[\hat{g}^2]_{t-1} < G$), make the gradient trimming threshold take a fixed value G; when the square of the cumulative gradient satisfies $E[\hat{g}^2]_{t-1} > G$ as training continues, make the gradient trimming threshold take $C_t = \eta \frac{\sqrt{E[\hat{g}^2]_{t-1}}}{1-(\beta_2)^{t-1}}$.

To summarize, the process of local cropping of gradient $g_{i,t}$ and addition of noise by device i ($1 \le i \le M$) in round t of training can be represented as

$$g_{i,t} = \frac{g_{i,t}}{\max\left(1, \frac{\|g_{i,t}\|_2}{C_t}\right)} + N(0, {C_t}^2\sigma^2),$$
$$where\ C_t = \begin{cases} G, E[\hat{g}^2]_{t-1} < G \\ \eta \frac{\sqrt{E[\hat{g}^2]_{t-1}}}{1-(\beta_2)^{t-1}}, E[\hat{g}^2]_{t-1} > G \end{cases} \quad (3)$$

From the above equation, it can be seen that as the number of iterations increases, the threshold C_t of local cropping will continue to decrease as $\sqrt{E[\hat{g}^2]_{t-1}}$ decreases, which in turn will make the perturbation noise $\xi \sim N(0, (C_t\sigma)^2 I)$ added to the gradient smaller and smaller, in order to achieve the purpose of improving the effectiveness of the model on the basis of ensuring the efficiency of training.

3. Experiment and Analysis

In this section, comparative experiments are designed to verify the superiority of ADP-BCFL using ADAM optimization algorithm to implement adaptive differential privacy federation learning mechanism. Comparison experiments are conducted on real datasets MNIST and Fashion MNIST for three parameter variables: model global accuracy, model privacy loss, and algorithm running time.

3.1. Global Accuracy Comparison

This section explores how the three federated learning methods, FedAvg, LDP-FL, and ADP-BCFL, compare in terms of global accuracy on the MNIST dataset and the Fashion MNIST dataset.

The curves obtained from the experiment, Fig. 2 and Fig. 3, lead to the following conclusions.

(1) With the same number of participants, the global accuracy of the FedAvg method is the highest on the two datasets, suggesting that the introduction of a noise mechanism can have an impact on the accuracy of the federated learning model.

(2) With the same number of participants and privacy budget, the ADP-BCFL method proposed in this paper is trained on two kinds of datasets, and the global accuracy obtained is higher than that of LDP-FL method and FedAvg method. And there is no large difference in the final accuracy of the model compared to the FedAvg method, which indicates that the ADP-BCFL method has a better performance.

(3) Fashion MNIST dataset has more complex image data compared to MNIST dataset, so all four schemes perform better on MNIST dataset than on Fashion MNIST dataset.

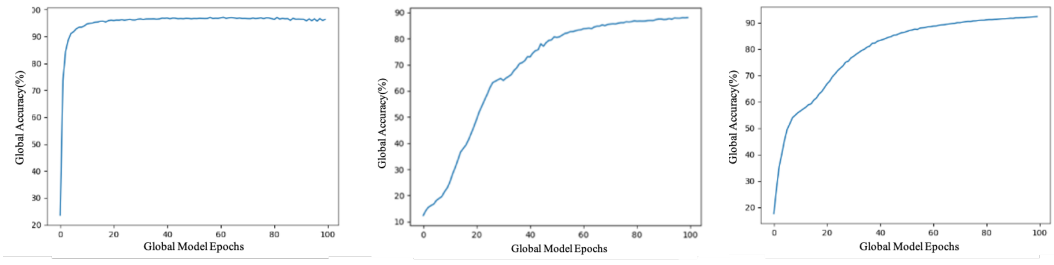


Fig. 2. Global accuracy of the MNIST dataset (left to right FedAvg, LDP-FL, ADP-BCFL)

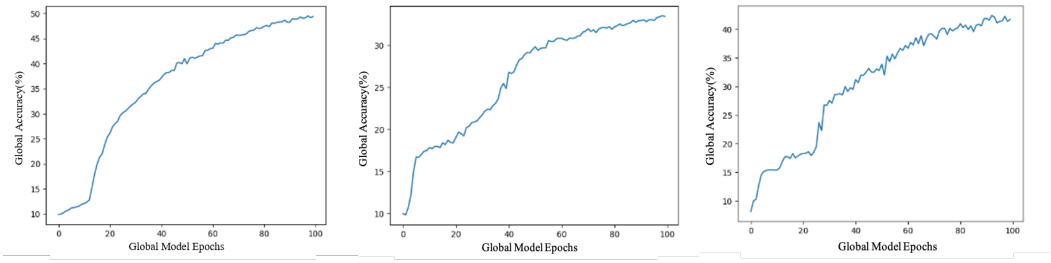


Fig. 3. Global accuracy of the Fashion MNIST dataset (left to right FedAvg, LDP-FL, ADP-BCFL).

3.2. Privacy Loss Comparison

This section explores the comparison of the performance loss of three optimization methods, ADAM, SGD, and ASGD, on the MNIST dataset and the Fashion MNIST dataset.

The curves obtained from the experiments, Fig. 4 and Fig. 5, show that the performance loss values of the ADAM optimization method utilized in this paper are smaller than those of the SGD optimization method and the ASGD optimization method for different iteration rounds on the two datasets, indicating that the ADP-BCFL optimization method outperforms the two compared optimization methods.

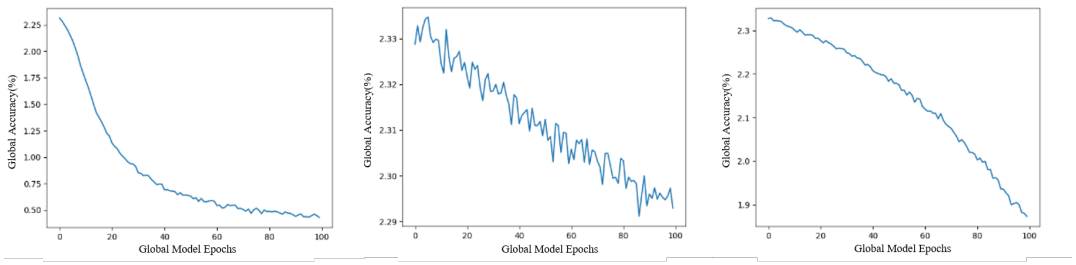


Fig. 4. Loss of privacy in the MNIST dataset (from left to right ADAM, SGD, ASGD).

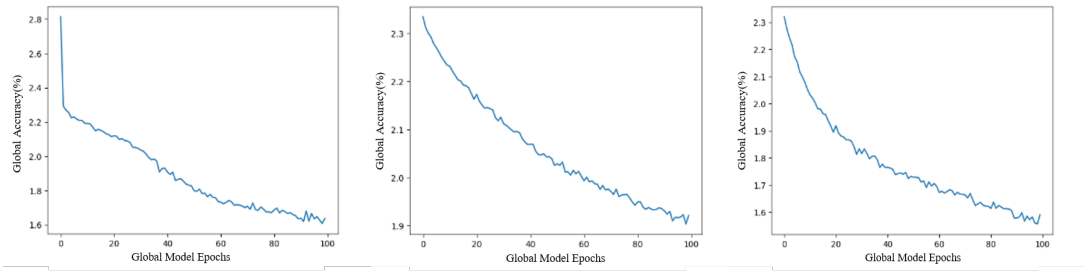


Fig. 5. Loss of privacy in the Fashion MNIST dataset (from left to right ADAM, SGD, ASGD).

3.3. Algorithm Runtime Comparison

This section explores three federated learning methods, FedAvg, ADP-BCFL, and LDP-FL, in terms of runtime comparison on the MNIST dataset and Fashion MNIST dataset.

The curves obtained from the experiments, Fig. 6, lead to the following conclusions.

(1) With the increase of the number of participants, the running time of all three methods on the two datasets increases, indicating that the increase of the number of participants leads to an increase in the running time of the algorithms.

(2) With the same number of participants, the FedAvg method has the shortest running time. Among the two federated learning privacy preserving schemes that introduce noise mechanisms, the running time of the ADP-BCFL method proposed in this paper is significantly lower than the running time of the LDP-FL method, indicating the effectiveness of the ADP-BCFL method.

(3) The Fashion MNIST dataset has more complex image data compared to the MNIST dataset, so the running time of the three methods on the MNIST dataset is shorter than that on the Fashion MNIST dataset.

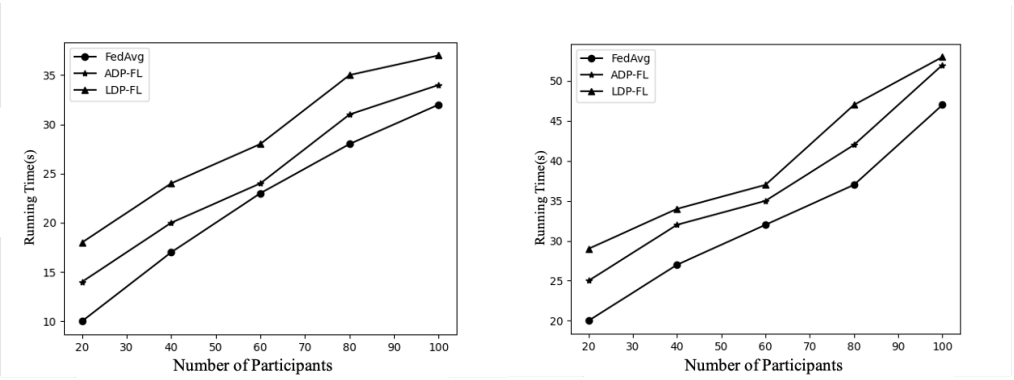


Fig. 6. Variation in running time with the number of participants (left to right MNIST, Fashion MNIST).

4. Conclusions

In this paper, a federated learning ADP-BCFL method based on adaptive differential privacy mechanism is proposed by combining blockchain technology, federated learning mechanism and data privacy protection methods. Experimentally verified, the method in this paper effectively solves the problem of inference attack in federated learning under the premise of guaranteeing model accuracy and privacy, and solves the problem of large loss of model privacy caused by the fixed gradient tailoring size that makes the addition of perturbation noise content inappropriate. Future work will focus on the fusion of federated learning with other advanced techniques and privacy-preserving methods in the construction of training models with higher security and accuracy.

Acknowledgements

This work is partially supported by the National Social Science Foundation. China (No.21BTQ079), the Humanities and Social Sciences Research Foundation of the Ministry of Education, China (No.20YJAZH046), Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing Fund, and Scientific Research Project of Beijing Educational Committee (KM202011232022).

References

[1]. R. X. Liu, H. Chen, R. Y. Guo, D. Zhao, W. J. Liang, C. P. Li, Survey on privacy attacks and defenses in machine learning, *Ruan Jian Xue Bao/Journal of Software*, Vol. 31, Issue 3, 2020, pp. 866-892 (in Chinese).

[2]. KANG Haiyan, JI Yuanrui, ZHANG Shuxuan. Enhanced Privacy Preserving for Social Networks Relational Data Based on Personalized Differential Privacy [J]. Chinese Journal of Electronics. 2022, 31(4): 741-751.

[3]. Y. X. Liu, H. Chen, Y. H. Liu, C. P. Li, Privacy-preserving techniques in federated learning, *Ruan Jian Xue Bao/Journal of Software*, Vol. 33, Issue 3, 2022, pp. 1057-1092 (in Chinese).

[4]. Y. Li, C. Chen, N. Liu, et al., A blockchain-based decentralized federated learning framework with committee consensus, *IEEE Network*, Vol. 35, Issue 1, 2020, pp. 234-241

[5]. D. Dillenberger, P. Novotný, Q. Zhang, P. Jayachandran, et al., Blockchain analytics and artificial intelligence, *IBM Journal of Research and Development*, Vol. 63, Issue 2/3, 2019, pp. 5:1-5:14.

[6]. C. Fang, Y. B. Guo, Y. F. Wang, Y. J. Hu, J. L. Ma, H. Zhang, Y. Y. Hu, Edge computing privacy protection method based on blockchain and federated learning, *Journal on Communications*, Vol. 42, Issue 11, 2021, pp. 28-40.

[7]. B. Hitaj, G. Ateniese, F. Perez-Cruz, Deep models under the GAN: information leakage from collaborative deep learning, in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS'17)*, 2017, pp. 603-618.

[8]. Y. L. Lu, X. H. Huang, Y. Y. Dai, et al., Blockchain and federated learning for privacy-preserved data sharing in industrial IoT, *IEEE Transactions on Industrial Informatics*, Vol. 16, Issue 6, 2020, pp. 4177-4186.

[9]. L. P. Wang, Z. Guan, Q. S. Li, Z. Chen, M. S. Hu, Survey on blockchain-based security services, *Ruan Jian Xue Bao/Journal of Software*, Vol. 34, Issue 1, 2023, pp. 1-32 (in Chinese).

(048)

Privacy-preserving Indoor Localization Based on Dummy Fingerprint and Homomorphic Encryption

Ying Li and Haiyan Kang
Department of Information Security, Beijing Information Science and Technology University,
Beijing, China
Tel.: + 13520416496
E-mail: kanghaiyan@126.com

Summary: Fingerprint localization technology based on Wi-Fi signals has received widespread attention due to its advantages of low cost and easy implementation. However, the privacy leakage problem during the localization process restricts its development. We analyze the privacy threats and propose a novel privacy preserving indoor fingerprint localization scheme. Firstly, in order to hide the user's real fingerprint, a single-point localization dummy fingerprint generation algorithm was designed, which protected the user's request privacy by adding k-1 dummy location fingerprints. Secondly, considering that an attacker may launch a speculative attack on location privacy by using the background knowledge such as the information submitted by the user in the last location request, a continuous request dummy fingerprint generation algorithm is designed. Finally, Paillier homomorphic encryption algorithm was used to protect the user's location privacy from untrusted servers and attackers. Experimental results show that the proposed scheme not only achieves reliable privacy protection but also has low computational overhead.

Keywords: Indoor localization, Wi-Fi fingerprint, Privacy protection, Dummy location, Paillier homomorphic encryption.

1. Introduction

The popularity of smart phones and the rapid development of various mobile applications in indoor environments have promoted the increasing demand for indoor positioning [1-2]. Wi-Fi devices are widely deployed in indoor environments and basically do not require additional special equipment support. Therefore, fingerprint localization technology based on Wi-Fi signals has become one of the most commonly used methods in indoor positioning [3], which usually includes offline phase and online phase. In the offline phase, the localization provider (LP) constructs a Wi-Fi fingerprint database by measuring the Received Signal Strength (RSS) values of each Wi-Fi Access Point (AP) in the indoor space from a preset series of reference points. In the online phase, the mobile terminal collects the RSS values of each AP received at the current position and sends it to the LP to request the positioning service. However, the localization mode through communication and interactive computing produces a unique privacy leakage problem. During the localization process, the user's location request information may be stolen by untrusted indoor positioning system or other malicious attackers.

At present, the privacy protection work in the indoor location scene mainly adopts the scheme based on cryptography, which uses encryption algorithms to process fingerprint data and complete the localization related algorithms in the ciphertext domain, but the calculation and communication overhead of the scheme is too large [4]. Literature [5] studies the use of differential privacy to inject noise into the user's fingerprint data to protect location privacy. Dummy location technology is an effective method to protect privacy information with the advantages of simple deployment and without affecting the quality of service [6]. However, the difficulty of this strategy is that the generated dummy location fingerprint cannot be distinguished from the real location fingerprint.

In order to solve the problem of privacy leakage in the online phase of indoor localization, this paper designs an indoor localization scheme combining dummy location and homomorphic encryption technology with the help of a semi-trusted third-party server. The main contributions as follows:

a. A single-point localization dummy fingerprint generation algorithm is designed, and the historical request frequency is used as side information to generate dummy measurement information that is more consistent with the true distribution, so as to solve the problem that the dummy location fingerprint is easy to be filtered by attackers.

b. A continuous request dummy fingerprint generation algorithm is designed, which considers the correlation information of the locations in the continuous positioning requests to generate a fingerprint set that maximizes the transfer entropy, so as to solve the problem that attackers use continuous positioning requests to reduce the location anonymity.

c. Paillier homomorphic encryption algorithm is introduced to realize the secure transmission of user positioning results, which protects the user's location privacy and database information from untrusted servers and attackers.

d. The performance of the proposed scheme is fully verified on the simulation data set. The experimental results show that the proposed scheme achieves the protection of privacy information without affecting the positioning accuracy.

2. Localization Scheme Based on Dummy Fingerprint and Homomorphic Encryption

The proposed scheme consists of a mobile terminal, a semi-trusted third-party anonymous server and a location server. Mobile terminals include smart phones, tablets and other devices that can collect information from nearby APs. Users to be located collect Wi-Fi RSS sampled from different APs at their current location, generate and encrypt the indicator vector V using Paillier homomorphic encryption algorithm, and then send them to the anonymous server together. The anonymous server is composed of the dummy fingerprint generation algorithm, which is responsible for storing and processing the user's historical positioning requests, helping users quickly generate positioning query fingerprint sets and send them to the LP for positioning requests. LP uses the positioning algorithm to calculate the location information of each fingerprint to match the corresponding physical coordinates from the database. Then, based on the homomorphic characteristics of Paillier algorithm, the homomorphic dot product operation is performed on the encrypted indicator vector and the positioning result, and the encrypted positioning result corresponding to the user's real fingerprint is obtained by matrix multiplication selection and returned to the user. The overall workflow of the scheme is shown in Fig. 1.

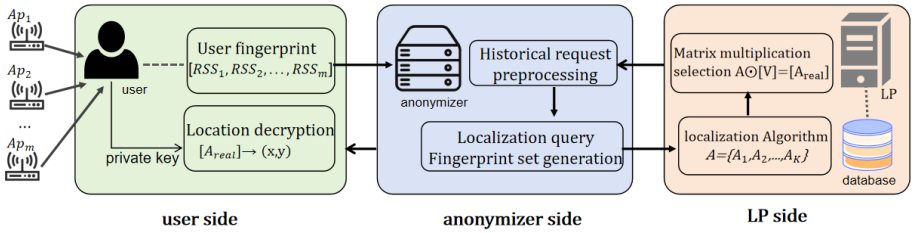


Fig. 1. Workflow of the proposed scheme.

2.1. Localization Query Fingerprint Set Generation

Considering the problem that attackers may use background knowledge to reduce fingerprint anonymity, a method based on entropy measurement was proposed to generate location query fingerprint set, which included two parts: a single location dummy fingerprint generation (SLFG) algorithm and a continuous request dummy fingerprint generation (CRFG) algorithm, aiming to avoid the attack of attackers to the greatest extent and protect the user's location privacy.

SLFG algorithm defends against the single point localization attack. The main process is as follows: (1) The anonymous server uses the k-means algorithm to divide the historical localization request fingerprint and calculates the query probability of each cell block, (2) The cell blocks are sorted based on the query probability, and 2k cell blocks before and after the actual fingerprint are selected to form the candidate cell set, (3) The location dispersion and entropy are combined to optimize the selection problem of anonymous cell, and the cell block with the maximum objective function value is selected to be added to the anonymous cell set by k-1 rounds of cycle. The objective function is defined as:

$$\{(-\sum_{V_d \epsilon S_{k-1}} q_d \log_2 q_d - q_i \log_2 q_i)\omega_i\}, \quad (1)$$

where $\omega_i = \lambda \prod_{V_d \epsilon S_{k-1}} dis(V_d, V_i)\, dis(V_{real}, V_i)$, (4) A fingerprint is randomly selected from each anonymous cell block and the user's real fingerprint together to form the final localization query fingerprint set.

In the case of continuous user requests for positioning, CRFG algorithm considers the new side information of the user's continuous position change. Based on the dummy fingerprint set submitted by the user in the last positioning request, it calculates the possibility that the members of the fingerprint set are the user's real fingerprint, and introduces the concept of information entropy to select the fingerprint set that maximizes the entropy value. Transfer entropy is defined as follows.

$$H = -\sum_{y=1}^{k_{j+1}} \Pr(l_y^{j+1} = l_{real}^{j+1} \mid S_j) \log_2 \Pr(l_y^{j+1} = l_{real}^{j+1} \mid S_j) \quad (2)$$

2.2. Matrix Multiplication Selection

For the request fingerprint set with anonymity budget k, there are k query results $A = \{A_1, A_2, \ldots, A_k\}$. According to reference [7], based on the homomorphic property of Paillier algorithm, the server performs the homomorphic dot product operation on the indicator vector [V] and the positioning result A, and the specific calculation process is as follows:

$$\begin{aligned} A \odot [V] &= (A_1 \quad \ldots \quad A_k) \odot \begin{pmatrix} [V_1] \\ \ldots \\ [V_k] \end{pmatrix} = \\ &= (A_1 \otimes [V_1]) \oplus \ldots \oplus (A_{d^*} \otimes [V_{d^*}]) \oplus \ldots \oplus (A_k \otimes [V_k]) = \\ &= [A_1 V_1] \oplus \ldots \oplus [A_k V_k] = [A_1 V_1 + \ldots + A_k V_k] = \\ &= [0 + \ldots + A_{d^*} + \ldots + 0] = [A_{real}] \end{aligned} \quad (3)$$

3. Experimental Analysis

We conduct simulation experiments to evaluate the performance of the proposed scheme, simulating a simplified indoor environment based on the log-distance path loss model, with a region area of 20 m×15 m, a total of 300 reference points with a distance of 1m and 12 access points are considered, and the target is set to move in random directions at a speed of 1 unit length per second. It is mainly evaluated from the aspects of computational cost and privacy protection.

Degree of privacy protection: Fig. 2 compares the conventional DLS algorithm for generating dummy locations, our proposed CRFG algorithm and the random scheme for randomly selecting dummy fingerprints, and gives the optimal location entropy value as a benchmark. Fig. 3 compares the performance of our proposed CRFG algorithm and SLFG algorithm in terms of transfer entropy with the change of k value.

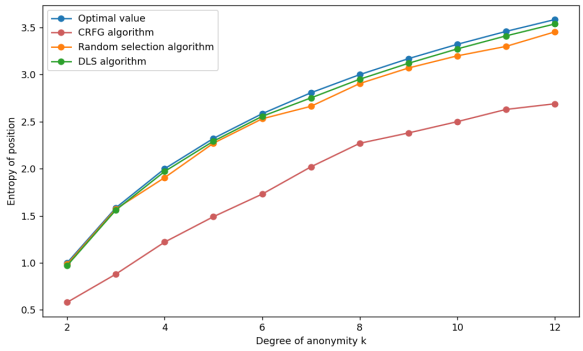


Fig. 2. Location entropy of different k.

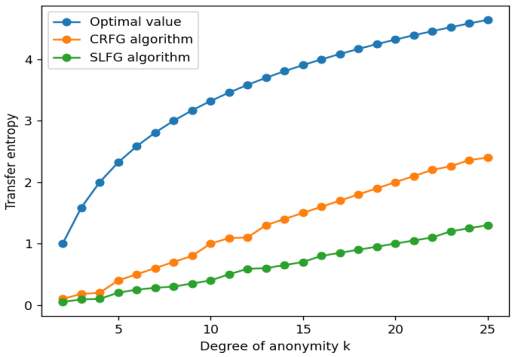


Fig. 3. Transfer entropy of different k.

The experimental results are analyzed as follows:

a. Both our algorithm and DLS take into account information such as historical query probabilities when generating fingerprint sets, so they can almost achieve near-optimal performance;

b. CRFG algorithm has a higher transition entropy than single point positioning, that is, it reduces the possibility of breaking k-anonymity, thus providing users with a higher level of location privacy;

c. The entropy increases with the increase of the privacy parameter k, because the greater the k, the less the probability that the untrusted LSP will recognize the user's fingerprint.

Computational time cost: Fig. 4 explores the impact of anonymity degree k on the actual running time of each stage, and Fig. 5 compares the time overhead of our work with that of PriWFL [8] scheme which uses fully homomorphic encryption to protect user location privacy.

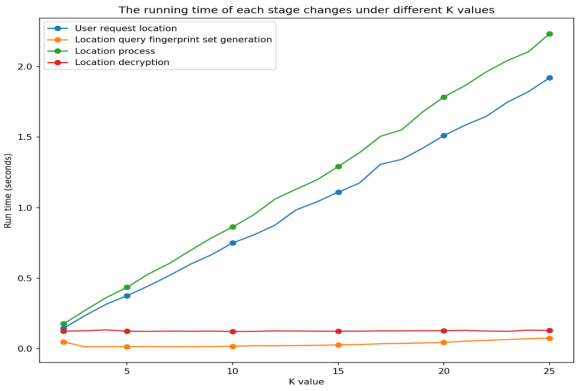


Fig. 4. The running time of each stage of the scheme.

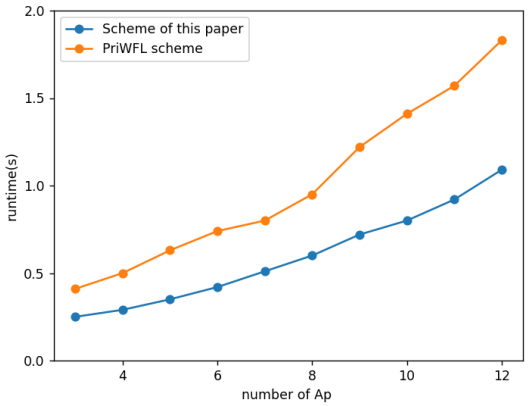


Fig. 5. Effect of the number of APs on the running time.

The experimental results are analyzed as follows:

a. The user's time overhead mainly comes from the encryption of the selection vector and the decryption of the positioning result;

b. Our work takes less time than the PriWFL scheme, because the PriWFL scheme needs to calculate the distance with all the fingerprints in the ciphertext domain, which incurs a large overhead;

c. The run time increases with the number of aps because more aps led to more RSS data, which greatly increases the computational cost.

4. Conclusions

Based on virtual location and homomorphic encryption technology, the privacy protection indoor

positioning scheme realized by third-party server structure solves the privacy leakage problem in Wi-Fi indoor positioning service. In addition, our work does not interfere with the Wi-Fi signal measured by the user, so it has no impact on the localization error of the Wi-Fi fingerprinting localization algorithm. Finally, the simulation results show that the proposed scheme has high reliability, which not only improves the privacy security, but also meets the real-time requirements of positioning with low computational overhead.

Acknowledgements

This work is partially supported by the National Social Science Foundation of China (No.21BTO079), The Ministry of Education in of Humanities and Social Science Project (No.20YJAZH046), Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing project (No.GJJ-22-03).

References

[1]. R. Z. Chen, et al., Application status, development and future trend of high-precision indoor navigation and tracking, *Geomatics and Information Science of Wuhan University*, Vol. 48, Issue 10, 2023, pp. 1591-1600.

[2]. X. Zhu, W. Qu, et al., Indoor intelligent fingerprint-based localization: principles, approaches and challenges, *IEEE Communications Surveys & Tutorials*, Vol. 22, Issue 4, 2020, pp. 2634-2657.

[3]. Z. H. Wang, Y. Xu, Survey on privacy protection indoor positioning, *Journal on Communications*, Vol. 44, Issue 9, 2023, pp. 188-204.

[4]. Z. H. Hu, Y. Z. Li, et al., PriHorus: privacy-preserving RSS-based indoor positioning, in *Proceedings of the IEEE International Conference on Communications (ICC'22)*, Seoul, Korea, 16-20 May 2022, pp. 5627-5632.

[5]. P. Zhao, H. B. Jiang, et al., P3-LOC: a privacy-preserving paradigm-driven framework for indoor localization, *IEEE Transactions on Networking*, Vol. 26, Issue 6, 2018, pp. 2856-2869.

[6]. P. Zhao, W. Liu, G. L. Zhang, et al., Preserving privacy in Wi-Fi localization with plausible dummy locations, *IEEE Transactions on Vehicular Technology*, Vol. 69, Issue 10, 2020, pp. 11909-11925.

[7]. Y. Wu, et al., Enhanced privacy preserving group nearest neighbor search, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 33, Issue 2, 2021, pp. 459-473.

[8]. H. Li, L. Sun, H. Zhu, et al., Achieving privacy preservation in Wi-Fi fingerprint-based localization, in *Proceedings of the IEEE Conference on Computer Communications*, Toronto, Canada, 27 April – 2 May 2014, pp. 2337-2345.

(051)

Application Pre-trained Network – ResNet50 for the Classification of Electronic Components

Lien Pham Thi [1], Long Nguyen The [2,3] and Huong Nguyen Thu [1,2]
[1] University of Information and Communication Technology, Thai Nguyen University,
70000, Thai Nguyen, Viet Nam
[2] Department of Informatics, Institute for Cybersecurity and Digital Technologies,
MIREA – Russian Technological University
[3] Laboratory of Artificial Intelligence and Machine Learning, Institute of Information Technology
and Data Science, Irkutsk National Research Technical University, 664074, Irkutsk, Russia
Tel.: +79246039341
E-mails: ptlien@ictu.edu.vn, nguen@mirea.ru, thuhuongyb@gmail.com

Summary: Electronic component classification is frequently a straightforward assignment that involves classifying a single object against a plain background. This is due to the fact that many applications make use of a technological procedure that has fixed camera positions, consistent illumination, and a predetermined set of components that are classed. To date, no significant effort has been made to create a technique for object classification in industrial applications under the aforementioned circumstances. For this reason, the classification of a certain technical process is the main emphasis of this work. Using a stationary camera, the technique sorts electronic components on an assembly line. The study examined each of the key processes needed to construct a classification system, including the generation of databases, neural networks, and images. Using the suggested image acquisition technique, an image dataset was created in the first phase of the experiment, after which pre-trained networks were designed and evaluated. According to the findings, the pre-trained network (ResNet50) gets the maximum accuracy of 97.5 %.

Keywords: Artificial intelligence, Data-augmentation, Electronic component classification, Resnet-50, Pre-trained neural network.

1. Introduction

Many computer vision problems focus on the classification of a single object against a simple background. It is necessary to define appropriate picture attributes for this purpose. Various visual characteristics are appropriate for various uses. The labelled data set and the features must be created in order for the classifier to be trained. Programmer understanding the domain of the analyzed images was necessary for the meticulous handwork involved in developing these applications. picture processing methods including picture enhancement, image filtering, and morphological processes were needed for this step. They made it easier to define appropriate characteristics for classes that are easily separable in low-dimensional space. Some algorithms for more difficult jobs, like mean-shift clustering, GrabCut, and watershed segmentation, were created based on the aforementioned principles.

Another novel network architecture is GooggleNet to V4 [1]. Four parallel branches are present in the conception modules it employs. The network's breadth and flexibility to handle varying input image resolutions and scales are enhanced by this architecture.

Deep learning can be accomplished practically with pre-trained models. A pre-trained model is one that has already undergone extensive training on a sizable dataset, typically related to an image classification problem. As a result, its spatial feature hierarchy can function as a general model for a range of computer vision issues. Pre-trained models can be used in two ways: fine-tuning and feature extraction. In order to extract important features from a fresh image, feature extraction makes advantage of the representation that a previously trained model has learned. Convolutional layers from previously trained networks are adopted, a new classifier is used in place of dense layers, and fresh samples are used to train the classifier. In addition to the new classifier, several convolutional layers in the fine-tuning techniques are also trained to identify other features in the images.

A components' package classification system based on a custom convolutional neural network was introduced in [2]. The proposed model could identify the 2D pattern of electronic components using nineteen features of surface mounting devices. The experiments demonstrated a 95.8 % accuracy of classification. Zhang [3] developed another custom network to classify electronic components into eleven categories. The custom network outperformed other pre-trained networks, such as Xception, VGG16, and VGG19, obtaining the highest accuracy in single-category and diverse component classification.

To solve the problem of classifying electronic components using a small dataset, which proposed a Siamese network. According to the authors, this solution improves the classification quality of electronic components and reduces the training cost. In [4, 5] authors proposed a custom convolutional network for classifying capacitors, resistors, and

diodes. To analyse its performance, she compared it with the pre-trained networks: AlexNet, ShuffleNet, SqueezeNet, and GoogleNet.

The technological process often employs constant lighting conditions, a static camera position and a fixed set of classified components, which can be completely different from the set dedicated to another task. Consequently, each process should possess its classification system based on an image acquisition module and a dedicated classifier trained on assigned objects. The system should be accurate and flexible, facilitating straightforward dataset creation and classifier development.

Therefore, the motivation for the present work was to develop an accurate and flexible system for electronic part classification in industrial applications. To this end, an approach was proposed for a specific technological process of radio communication device manufacturing. It was aimed at classifying ten electronics components appointed by a product engineer. The components were utilised to construct a dataset, which could be effective for neural network training. The tested network structures employed pre-trained and custom networks since both structures have proved their applicability in previous research.

The main contributions of this paper can be summarised as follows: The results are encouraging and show that the present method can accurately classify components in a specific technological process. Additionally, since its straightforward implementation, it can be effortlessly adapted to similar applications.

2. Methodology

Creating a system for classifying electronic parts for industrial use was the goal of the project. First and foremost, a vision system was created with this goal in mind. It made it easier to gather images in order to compile a dataset of electronic parts. A convolutional neural network baseline model was created using the generated dataset. It made it possible to verify the accuracy of the dataset and establish the foundation for neural network structures. The next step involved designing the network architectures. Pre-trained networks and a custom model were considered for this aim. Their single graphics processing unit (GPU) was used for the programming implementation, which was based on the publicly available TensorFlow 2 and Keras libraries. The steps that were taken are fully explained in this section.

2.1. Dataset

The dataset was collected using the designed vision system. It includes 3994 images of eleven classes: Class 0 (USB), Class 1 (integrated circuit), Class 2 (fan), Class 3 (background), Class 4 (coil), Class 5 (AUX), Class 6 (USB2), Class 7 (communication unit), Class 8 (connector), Class 9 (display) and Class 10 (processing unit). The classes constitute a set of fundamental electronic components for which automatic classification is profitable for the particular manufacturing process.

The brightness of captured photographs can be affected by the camera's settings, even though it is presumed that lighting conditions are constant. Furthermore, there's a chance that the camera's distance from the object needs to be significantly changed. As a result, the process that was developed makes the assumption that photos should be taken in a variety of lighting and distance scenarios.

2.2. Pretrained Networks

Since fine-tuning demands a larger dataset, feature extraction was selected for pre-trained network investigations. Based on the literature review, the most promising network architectures were chosen:

- VGG16;
- VGG19;
- ResNet50.

The dense layers of the above networks were replaced with new classifiers (Table 1). Additionally, an image processing step was employed. It was because adopted networks expect different image formats at the input layer. For example, VGG16 demands images converted from RGB to BGR and a zero-centred colour channel. This operation, as well as the programming implementation of the pre-trained networks, is very straightforward using the applications module of the Keras library.

Table 1. Parameter of training and testing model (FC – Fully connected, SF – SoftMax function).

VGG-16	VGG-19	ResNet-50
FC-16	FC-32	FC-64
SF-10	SF-10	SF-10

The Resnet architecture follows two fundamental design principles. First, regardless of the size of the output feature map, there is the same number of filters in each layer. Second, to maintain the time complexity of each layer when the size of the feature map is halved, it has twice as many filters. The building block for Resnet employs a bottleneck design, which lowers the number of parameters and matrix multiplications. This makes training each layer considerably faster. Instead of using two layers, it employs a stack of three layers. The Resnet-50 model is a pre-trained convolutional neural network with 50 layers depth and can classify up to 1500 class (object image). To adapt this model to our classification problem, we have frozen the weights of the ten first layers and we have added one fully connected layer at the end, having 2 neurons since we have a 2-class classification problem. For the classification, we have divided each set into 75 % for the training phase and 25 % for the

test phase and we have trained the Resnet-50 model for only 20 epochs.

To develop an electronic components classification method based on convolutional neural networks, the proposed models are evaluated. For this purpose, numerous experiments were performed. Firstly, the different custom model structures were trained, and their performances were assessed using accuracy rates and learning curves. In the same way, pre-trained models with different classifiers were analysed. Finally, a comparative analyse was performed using the most promising custom and pre-trained models to find the best network for the classification task.

Due to the stochastic nature of deep learning models, each network was trained ten times, and the mean accuracy was considered. Each of the ten training steps was performed with a different random seed value (random split of train and validation sets). However, the same ten training steps were deployed for all networks. Consequently, each network was trained with the same random seed values and hence with the same random splits of the dataset into test and validation sets. This procedure ensured the same training conditions for the analysed models. For final analyses, each chosen model was trained three times using the same random seed, and the best performance was evaluated.

Each training step utilised a checkpoint mechanism. It allowed saving the network's weights if the achieved accuracy at the epoch's end was higher than the previously recorded one. In this way, the best models obtained during training were saved.

3. Analysis and Evaluate Experimental Results

When we classified our images with Resnet-50 model, we have obtained a value of 97.58 % for the training rate, a value of 97.04 % for the accuracy rate and 4 h 17 min 05 s for the execution time. Fig. 1 illustrates the results obtained for the Resnet network. They indicate that only the network with the simplest classifier achieved good accuracy (>97 %).

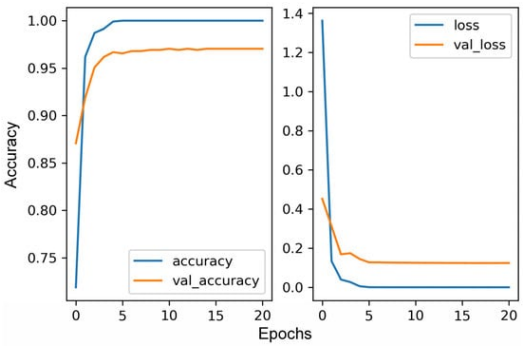


Fig. 1. The result of accuracy of ResNet-50.

The generated learning curves (Fig. 2) suggest that overfitting is presented in this model.

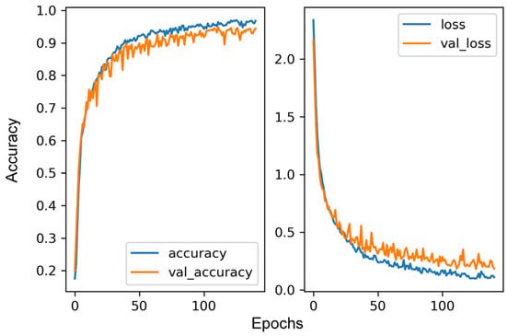


Fig. 2. The accuracy of learning curves ResNet-50.

Data augmentation yielded only a slight improvement in accuracy. The regularisation techniques also had a minor impact on the ResNet50 performance (Table 2). Data augmentation slightly increased accuracy for all models, while the application of dropout and weight regularisation improved the performance of the simplest model.

Table 2. Classification results of VGG-16, VGG-19, Resnet-50 model.

	VGG-16	VGG-19	Resnet-50
Training rate (%)	92.25	92.30	97.58
Accuracy (%)	92.10	90.05	97.04

4. Conclusions

The challenge of categorizing electronic components for industrial applications was tackled in this work. According to the results, the ResNet50 architecture-based solution provides the highest classification accuracy. This paper developed a classification system, including creating databases, developing neural networks, and acquiring images. To enable other researchers to duplicate and alter the aforementioned procedures, each one is explained in great detail. Furthermore, the dataset and code are made available for simple implementation. The suggested remedy is intended for technological procedures that have a fixed camera position, consistent lighting, and a particular group of components that are classified. Still, a lot of image processing apps can make use of the aforementioned circumstances. For object classification, an indoor video surveillance system can make use of the current technique.

References

[1]. C. Liu, S. Weng, Fusion of electronic nose and hyperspectral imaging for mutton freshness detection using input-modified convolution neural network, *Food. Chem.*, Vol. 385, 2022, 132651.

[2]. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in *Proceedings of the 3rd International Conference on Learning Representations*, USA, 7-9 May 2015, pp. 1-14

[3]. X. Han, Z. Zhang, Pre-trained models: Past, present and future, *AI Open*, Vol. 2, 2021, pp. 225-250.

[4]. R. C. Salvador, A. A. Bandala, DeepTronic: an electronic device classification model using deep convolutional neural networks, in *Proceedings of the IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology (HNICEM'18)*, 2018, pp. 1-5.

[5]. Y. J. Wang, Y. T. Chen, An artificial neural network to support package classification for SMT components, in *Proceedings of the 3rd International Conference on Computer and Communication Systems (ICCCS'18)*, 27-30 April 2018, pp. 130-134.

(052)

Psychophysiological Signals Underlying Sexual *presence* in VR: Case Study of an Atypical Arousal Pattern

M. Brideau-Duquette [1 2], S. Saint-Pierre-Côté [1 3] and P. Renaud [1 2]
[1]Université du Québec en Outaouais, Centre Interdisciplinaire de Recherche et d'Innovation en Cybersécurité et Société, 5, rue Saint-Joseph Saint-Jérôme, Québec, Canada
[2] Institut National de Psychiatrie Légale Philippe-Pinel, Laboratoire Immersion Forensique, 10 905 Henri-Bourassa Est, Montréal, Québec, Canada
[3] École de Technologie Supérieure, 1100 Notre Dame Ouest, Montréal, Québec, Canada
E-mail: patrice.renaud@uqo.ca

Summary: The burgeoning field of cybersexuality underscores the need to understand *sexual presence* within virtual reality (VR)—a state of sexual arousal and personal erotic perception experienced in virtual environments. This study introduces a novel method for analyzing psychophysiological signals to quantify sexual presence. Our research demonstrates the utility of this method, with empirical findings showing it accounts for a substantial variance in subjective sexual experiences among different genders during VR immersion with neutral and sexually personalized avatars. A notable aspect of this study is the identification of gender-specific patterns in response to VR stimuli, which has implications for both mental health and cybersecurity. This paper furthermore presents results from an atypical participants sexually aroused in the presence of a neutral android-like virtual character.

Keywords: Virtual reality, Immersion, Psychophysiological signals, Sexual arousal, Sexual presence, Case study.

1. Introduction

With advancements in extended reality (XR) and artificial intelligence (AI) technologies rapidly gaining traction, particularly among Generation Z, there is a notable surge in interest in the cybersexual domain [1-2]. Additionally, the advent of multimodal head-mounted displays (HMDs) suitable for XR, integrating EEG, eye-tracking, and various physiological signals, has been highlighted as a substantial advancement [3-4].

Sexual Presence: Definition

Sexual telepresence, or sexual presence, is defined as a psychophysiological state of technologically induced sexual arousal, which includes not only a personal erotic perception but also an element of illusion, a convincing yet virtual experience that mimics physical interactions [5-6]. The technological trajectory from smart sex toys to XR-based sexual interactions and onward to humanoid sex robots illustrates the increasing use of pervasive computing to enhance intimate encounters. These technologies are not solely focused on fulfilling sexual desires but are designed to foster a sense of presence and intimacy, utilizing the principles of ubiquitous computing.

This paper briefly outlines a newly developed methodology for examining sexual presence and shares insights from recent findings on measuring this peculiar subjective experience [7-8-9-10]. It also uses a case-study design to analyse an atypical arousal pattern found in one of our participants.

2. Methodology

We developed a digital avatar customization tool using Unity (Version 2018.4.4), offering fifteen features for users to intuitively modify avatars from the Genesis 8 collection, with tools like dropdown menus and sliders for detailed changes [7]. Users can alternate between facial and full-body views and rotate the avatar for a complete perspective. These avatars were animated using motion capture of male and female collaborators for sexual scenarios, and with pre-set motions from the "Y Bot" humanoid for neutral scenarios, utilizing Unity (Version 2020.3.36) for constructing and rendering digital environments, ensuring uniform lighting. The experiments were conducted at École de Technologie supérieure in Montréal, utilizing a high-spec computer with an Nvidia GeForce RTX 3080 graphics card, Intel Core i7-10700K CPU, and 32 GB RAM, alongside an HTC Vive Pro Eye Head-Mounted Display for immersive experiences.

Penile tumescence was measured using penile plethysmography (PPG), where a mercury-filled, stretchable rubber band is placed around the penis mid-shaft to detect engorgement through changes in electrical conductivity and mercury level. These changes, indicating variations in penile circumference, are calibrated and processed using Limestone Technologies' DataPac and PrefTest Professional Suite.

The vaginal plethysmograph (VPG) from Biopac employs an infrared LED to project light onto the

vaginal wall, with reflections measured by a phototransistor indicating blood volume changes. These changes are represented in millivolts from a baseline, analyzing the vaginal pulse amplitude to assess vascular pressure variations. Signal processing includes filtering for noise reduction and calculating statistical metrics like mean, standard deviation, variation coefficient, and area under the curve to understand physiological responses.

Fig. 1. Appearance of the android-like (left) and realistic female sexually alluring (right) avatars; see reference [8].

Eye movement was tracked using the HTC Vive Pro Eye Head-Mounted Display with SRanipal SDK, which captures eye motion at 90 Hz. This technology offers precise tracking capabilities, with an accuracy ranging between 0.5 and 1.1 degrees, allowing for detailed recording of gaze behavior. The Gaze Radial Angular Deviation (GRAD) is described as the angle formed between two vectors: one extends from the center of the eye to a virtual measurement point (VMP), located on the target area, and the other represents the normalized direction of gaze from the SrAni-pal system. VMPs were placed at the facial, breast, and genital areas of the virtual characters to monitor (see Fig. 2).

Electroencephalogram (EEG) data were collected using a cap with 32 active electrodes following the 10–20 system, amplified by Brain Vision's ActiChamp and recorded with MOVE and Recorder software. The EEG setup featured a 500 Hz sampling rate and filtering to address frequency-specific noise, establishing a real-time reference at the Cz electrode.

In itself subjective sexual presence as a psychological construct is measured from a questionnaire [5].

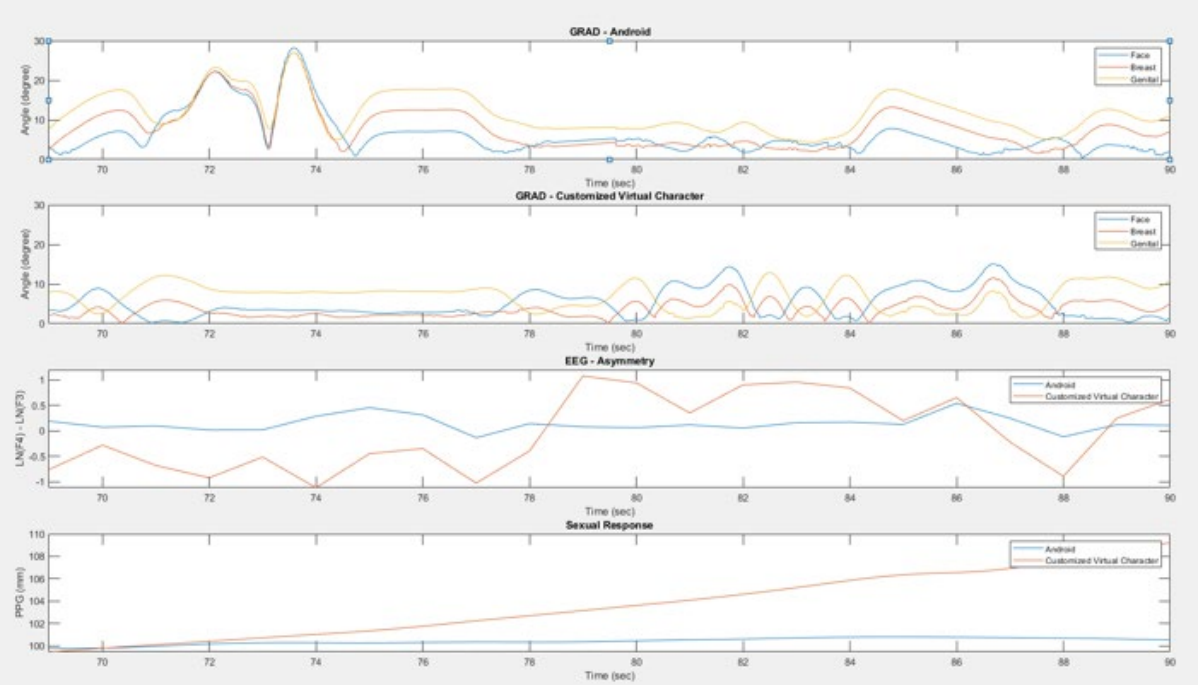


Fig. 2. Psychophysiological signals from a typical male participant; from top to bottom for both android neutral virtual character and the sexually customized one: GRAD, EEG asymmetry in F3-F4 and sexual response measured from PPG.

Recruitment for the study was achieved through social media posts, printed leaflets, email campaigns, and word of mouth. This research was a component of a broader project detailed in Saint-Pierre Côté et al., 2023. The study included eleven cisgender male participants with an average age of 26.1 and a standard deviation of 8.80. Briefly, the participants were tasked with creating a personalized nude avatar of the opposite sex and an environment designed to maximize their sexual interest. In the virtual reality (VR) setting, they were exposed to various combinations of avatars and environments, including their personalized creations and a standard set provided for all participants of the same sex, as well as an android-like avatar.

The case study results discussed below pertain exclusively to this male sample.

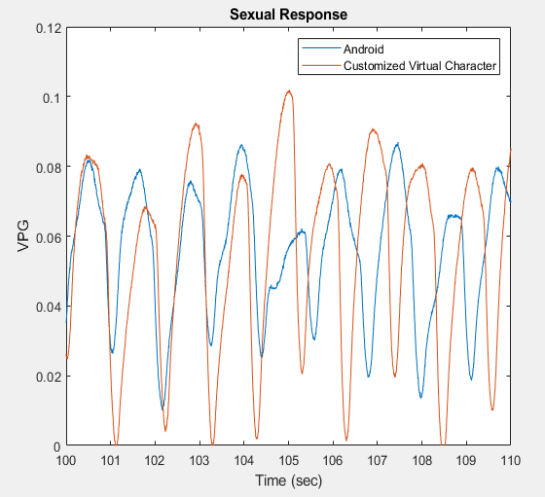


Fig. 3. Typical female sexual response measured from PPG for both android neutral virtual character and the sexually customized one.

Data Analysis: EEG Dynamic Cross Entropy (DCE)

Explained elsewhere [8], EEG DCE computation was used to quantify brain synchronization during sexual arousal prompted in virtual immersion.

The processing of EEG data was executed through Analyzer 2.1 by Brain Vision, identifying and removing noisy channels through visual examination. A subsequent ocular ICA using the mean slope approach, condition-specific data sets, and the Infomax along with sum of squared correlations techniques in Analyzer facilitated the elimination of components primarily consisting of vertical and, where necessary, horizontal eye-movement EEG artifacts. The data were then re-referenced to the average mastoid signal. For the eleven participants, the 30-second data segments were free from additional artifacts and remained unmodified.

Dynamic Cross-Entropy (DCE), as outlined by Aur and Villa-Rodriguez [11], serves to gauge the regularity and intricacy across several time series, especially within various frequency bands (see Eq. 1). It diverges from conventional entropy assessments by gauging the joint intricacy of multiple signals rather than a solitary one. DCE employs entropic metrics like Sample Entropy (SampEn) for each band-pass-filtered signal to focus on pertinent frequencies. The aggregate entropic values are then averaged to deduce the DCE, showcasing the collective regularity or intricacy, thereby aiding in the analysis of activity synchronization across different channels, such as in EEG, revealing the spatial and synchrony patterns of neural activities. The DCE's sensitivity to time series regularities is contingent on the chosen parameters for entropy evaluation, utilizing an m value of 2 and a r value of 3.57 in all DCE analyses as per Aur and Villa-Rodriguez (2017). Custom scripts, primarily in Python version 3.10.0, were employed for DCE calculations, incorporating relevant libraries and adhering to Aur and Villa-Rodriguez's methodology.

$$DCE(y_1, y_2 \dots, y_c)^{(j)} = \frac{1}{\sum_{i=1}^{c} En(y_i)^{(j)}}$$

Equation 1. Mean DCE calculation, where y_i, is the individual signal (i.e., time series); $En(y_i)$ is the entropy measure (here, SampEn) for a given y_i; c is the number of signals considered together.

The procedure for DCE calculation involves: 1) Band-pass filtering of each signal to target specific frequency bands; 2) applying complexity entropy metrics, notably SampEn, to each filtered signal segment; 3) normalizing the averaged DCE across all segments and signals via the min-max method, adjusting the DCE values relative to the lowest and highest DCE values noted. The DCE analysis encompassed the alpha (8 – 13 Hz), beta (13 – 30 Hz), and their sub-bands including low alpha (8 – 10.5 Hz), high alpha (10.5 – 13 Hz), low beta (13 – 20 Hz), and high beta (20 - 30 Hz).

3. Results

Findings reported in other studies demonstrated that this methodology accounted for 73 % of the variance in subjective sexual presence among female participants [9], and explained 86 % of the variance in responses of a male sample when comparing their immersion experiences with a neutral android versus a personalized sexual avatar [8].

Furthermore, a case study results demonstrate that the area under the curve (AUC) of the EEG DCE signal allows for the detection of an atypical male case who exhibited a significant sexual response to the sexually neutral stimulus, namely the android character (see Fig. 1).

This participant (23 yr old) is the only one among our 11 male cisgender heterosexual participants who displayed a significant erectile response in this condition. His response, expressed in mm of circumferential stretching of the penile gauge is of 6.4 while the group average is 1.21 mm (SD=1.28). This gives him a Z-score of 4.05 (which corresponds to a p-value less than 0.0001). It is worth mentioning however that, following the group, the participant's erectile responses to the human sexual avatar were greater than that for the android character. Therefore, the intriguing aspect of this case lies in the fact that the individual experienced sexual arousal under a non-sexual condition.

This participant's EEG DCE AUC for F3-F4 frontal leads in alpha is of 76902 while the average value for the rest of the group is 27983 (SD=8921). Z-score for this value equals 5.5 standard deviations from the mean, which corresponds to a p-value less than 0.0001, signifying that the result is highly significant statistically. Such a result could suggest that the participant's brain activity, as measured by the EEG DCE AUC in the specified condition, is

atypically higher than that of his peers, potentially indicating a notable difference in alpha frontal activity synchronization. This difference could explain why this particular individual presented this unusual sexual response to such a stimulus.

This EEG finding aligns with another unique aspect of this case: specifically, his average Gaze Radial Angular Deviation (GRAD) from the genital Virtual Measurement Point (VMP) of the neutral character exceeded that of the rest of the group. With an average GRAD of 7.43 deg, this participant stands at 2.9 SD of the sample average (5.5 deg; SD=1.3). This Z-score corresponds to a p-value less than 0.01. His gaze behavior exhibited greater dispersion in the vicinity of the animated character with a neutral sexual portrayal.

On a more psychosocial level, this individual appears to overinvest in the sexual sphere, reporting an average of 13 orgasms per week compared to an average of 4.15 for the rest of the group (SD=3.24), with a z-score of 2.73 (p-value less than 0.01). He also reported a high score on the Game scale of the Immersive tendencies questionnaire [12]. This scale is specifically designed to assess an individual's propensity to engage in play, be it in video games, simulations, or other immersive play activities. A high score on this scale indicates that this individual has a strong tendency to actively engage and immerse himself in playful activities.

Finally, he reported an unusually high score of naturalness for the android neutral character on the Uncanny valley questionnaire [12] compared to the rest of the sample (AVG=11,2; SD=15). His Z-score on that matter is of 1,92 (p-value less than 0.05). This same participant did not show an unusual score on that matter when assessing the sexual avatar with the same questionnaire.

To summarize, this case study highlights an atypical sexual arousal case, specifically an erectile response in an adult male within a sexually neutral virtual context. While it is certainly possible for someone to experience a spontaneous erection in a sexually irrelevant context (due to fantasies or purely metabolic reasons), the case at hand is peculiar in several respects. It exhibits elements of sexual overinvestment, a strong propensity to immerse in gameplay, and a subjective assessment of the android character's realism leaning towards a more pronounced naturalness or realism compared to the rest of the group.

However, what is particularly interesting here is how the brain dynamics in the frontal region, interacting with the perceptuo-motor extraction process governing the interaction with the virtual stimulus, distinguishes this case from the rest of his group. In light of the previously mentioned psychosocial dimensions, perhaps we have here elements that explain the uniqueness of this case. At the very least, this difference in terms of psychophysiological mobilization in an immersive context constitutes a trail for a counterpoint analysis that is interesting to better understand how the feeling of sexual presence emerges.

Although interesting, these results and especially the single-case method from which they derive, have numerous limitations. Firstly, the very limited number of participants restricts the ability to generalize the results; a larger sample would allow for further validation of the characteristics of this atypical case, should they prove to be true. It is also difficult to establish a causal pattern following the isolated factors because other uncontrolled variables in the study may come into play. Furthermore, the statistical methods used are very limited and are employed here primarily for exploratory purposes.

4. Conclusions

As the global market for virtual reality (VR) pornography is projected to surpass a billion-dollar valuation by the year 2027, the phenomena of cybersexuality and sexual presence demand earnest consideration [1]. Accompanying this remarkable market expansion are potential challenges related to mental health and cybersecurity. Consequently, it becomes imperative to develop innovative and effective methodologies for the analysis and more identification of sexual presence through objective physiological and behavioral indicators [10].

In conclusion, this study contributes to the understanding of sexual presence by focusing on an atypical case identified through psychophysiological signals. The use of multimodal HCI technologies, as per reference [10], could allow for a more nuanced observation of individual responses to virtual sexual stimuli.

Our research underlines the importance of individual differences in the study of virtual environments and sexual presence. It points to the potential of these technologies to provide personalized insights, which can be critical for advancing research and practical applications in the field of sexual health, cybersecurity and virtual reality experiences.

Acknowledgements

The results discussed in this paper come from preliminary analyses of data collected in the first two authors' PhD thesis.
Thanks to Conseil de Recherche en Sciences Humaines du Canada, les Instituts de Recherche en Santé du Canada and les Fonds de Recheche du Québec.

References

[1]. Market Research Engine, Virtual and augmented reality Market Size, Share, Analysis, Report, Retrieved from https://www.marketresearchengine.com/virtual-and-augmented-reality-market., 2023.

[2]. Pornhub, 2022-year-in-review, Retrieved from https://www.pornhub.com/insights/2022-year-in-review, 2022.

[3]. A. Krugliak and A. Clarke, Towards real-world neuroscience using mobile EEG and augmented reality, *Scientific Reports,* 12, 2022, Article 2291.

[4]. M.-A. Moinnereau., A. A Oliveira, and T. H. Falk, Quantifying time perception during virtual reality gameplay using a multimodal biosensor-instrumented headset: a feasibility study, *Frontiers in Neuroergonomics*, 4, 2023, 1189179.

[5]. Brideau-Duquette, M., & Renaud, P, 2023, Sexual Presence: A Brief Introduction, in Encyclopedia of Sexual Psychology and Behavior, *Cham: Springer International Publishing*, pp. 1-9.

[6]. Fontanesi, L., & Renaud, P, Sexual presence: Toward a model inspired by evolutionary psychology, *New Ideas in Psychology,* 33, 2014, pp. 1-7.

[7]. S. Saint-Pierre Côté, M. Brideau-Duquette, D. Labbé, and P. Renaud, Sexual presence in virtual reality: a psychophysiological exploration, in *Proceedings of the IEEE Virtual Reality Conference,* Orlando, Florida, March 2024, P1135.

[8]. M. Brideau-Duquette, S. S. P. Côté, J. Pfaus, P. Renaud, First Probe into Frontal EEG Dynamic Cross-Entropy associated with Virtual Sexual Content, in Tareq Ahram, Waldemar Karwowski, Dario Russo and Giuseppe Di Bucchianico (eds.), Intelligent Human Systems Integration (IHSI 2024): Integrating People and Intelligent Systems. AHFE (2024) International Conference, vol 119, *AHFE International, USA*, 2024, pp. 110-119.

[9]. S. Saint-Pierre-Côté, M. Brideau-Duquette, D. Lafortune, J. Pfaus, P. Renaud, Learning mechanisms in virtual reality therapy for sexual disorders: delving into sexual presence, in *Proceedings of the 16th International Conference on Computer Supported Education,* Angers, 2024, accepted, 8 p.

[10]. C. Galaup, L. Séoud, P. Renaud, Multimodal HCI: a review of computational tools and their relevance to the detection of sexual presence, in Tareq Ahram, Waldemar Karwowski, Dario Russo and Giuseppe Di Bucchianico (eds.), Intelligent Human Systems Integration (IHSI 2024): Integrating People and Intelligent Systems. AHFE (2024) International Conference, Vol. 119, *AHFE International, USA*, 2024, pp. 137-143.

[11]. D. Aur, F. Vila-Rodriguez, Dynamic cross-entropy, *Journal of Neuroscience Methods*, 275, 2017, pp. 10–18.

[12]. B. G. Witmer, M. J. Singer, Measuring presence in virtual environments: a presence questionnaire, *Presence Teleoperat. Virt. Environ.*, 7, 1998, pp. 225–240.

[13]. C. C. Ho, K. F. MacDorman, Revisiting the uncanny valley theory: Developing and validating an alternative to the Godspeed indices, *Computers in Human Behavior,* 26, 6, 2010, pp. 1508-1518.

(053)

Investigating the Impact of Loop Closing on Visual SLAM Localization Accuracy in Agricultural Applications

F. Schmidt, F. Holzmüller, M. Kaiser, C. Blessing and M. Enzweiler
Institute for Intelligent Systems, Faculty of Computer Science and Engineering, Esslingen University,
Flandernstraße 101, 73732 Esslingen am Neckar, Germany
E-mail: fabian.schmidt@hs-esslingen.de

Summary: This study investigates the impact of Loop Closing on the localization accuracy of Visual Simultaneous Localization and Mapping (Visual SLAM) systems in unstructured agricultural environments, focusing on ORB-SLAM3, VINS-Fusion, and OpenVINS enhanced with Loop Closing. We assess each systems' performance in various scenarios to determine their efficiency and accuracy using Absolute Trajectory Error (ATE) and computational resource metrics. ORB-SLAM3 demonstrates a modest increase in computational demand with notable accuracy gains, making it efficient for applications where resources are limited. VINS-Fusion benefits from Loop Closing in smaller environments but faces significant challenges and high computational costs in larger settings. OpenVINS, when integrated with VINS-Fusion's Loop Closing, achieves consistent accuracy improvements but with substantial memory usage. The combination of OpenVINS and Maplab's offline Loop Closing offers the most significant accuracy enhancements, although it lacks computational performance data due to its offline nature. This research highlights the importance of selecting appropriate SLAM configurations based on environmental complexity and computational constraints.

Keywords: Visual SLAM, Localization, Loop closing, Unstructured environments, Agricultural robotics.

1. Introduction

Visual SLAM techniques have enabled a multitude of applications in robotics, particularly in outdoor scenarios where GPS may be unreliable. The ability of Visual SLAM systems to autonomously construct and update maps of their surroundings while simultaneously determining their own position has had great impact on the field of autonomous navigation. However, the accumulation of drift, resulting from errors in estimating the robot's motion over time, remains a significant challenge in practical applications. Especially for applications that have a long runtime, it is important to create a consistent and robust system that shows sufficient resistance to drift.

Loop Closing is a crucial technique developed to reduce drift in Visual SLAM systems. By identifying and correcting loops in the robot's trajectory, Loop Closing aims to ensure long-term localization accuracy. Whilst theoretically promising, Loop Closing introduces practical challenges, especially regarding the limited availability of computational resources in mobile robots.

This paper aims to investigate the impact of Loop Closing on Visual SLAM performance in outdoor robotics, considering the constraints of mobile platforms. The study provides an exhaustive analysis of multiple aspects concerning Loop Closing and its impact on the deployment of Visual SLAM systems in real-world environments. Our contributions include:

1. Benchmarking open-source Visual SLAM methods: We conduct a comprehensive evaluation of various open-source Visual SLAM methods in unstructured outdoor environments;
2. Quantitative analysis of Loop Closing Effects: We assess the influence of Loop Closing on localization accuracy across diverse driving scenarios and environmental conditions;
3. Resource analysis: We examine the additional computational overhead imposed by Loop Closing compared to SLAM methods without Loop Closing.

The remainder of this paper is structured as follows: Section 2 highlights existing Visual SLAM benchmarks in various environments and settings. Section 3 details the experimental setup, including specific Visual SLAM methods, dataset description, and evaluation criteria. Section 4 presents the results and discusses their implications and relevance for autonomous navigation. Finally, Section 5 concludes the paper, summarizing key findings and proposing future research topics.

2. Related Work

The literature on SLAM systems provides insights into applications, difficulties, and performance in a variety of scenarios, highlighting their significance in the development of robotic navigation. This chapter focuses on benchmarks for Visual and Visual-Inertial SLAM, essential for the evaluation and comparison of their performance.

Numerous benchmarks [1-4] have been established to systematically evaluate the efficacy of contemporary Visual and Visual-Inertial SLAM algorithms through the use of renowned datasets such as EuR oC [5], KITTI [6], and TUM RGB-D [7]. These benchmarks offer a systematic framework to assess algorithms across diverse environments, from indoor

spaces to unpredictable outdoor settings, incorporating various robotic platforms, such as drones, mobile robots, and cars. Furthermore, the benchmarks [1] and [4] evaluate the computational performance of the SLAM algorithms in addition to the localization accuracy, with [4] differentiating between various embedded computing platforms.

Apart from common benchmarks, [8] and [9] specifically explore Visual-Inertial SLAM applications in particular contexts, demonstrating their adaptability and potential constraints in challenging environments. [8] evaluates ten open-source Visual-Inertial SLAM algorithms in marine settings, a notably challenging environment for SLAM technologies due to factors like low visibility and dynamic lighting conditions. The comprehensive analysis uses datasets from underwater robots, offering insights into the performance of direct and feature-based SLAM methods. [9] investigates the feasibility of applying monocular Visual-Inertial SLAM methods to freight railways. This study demonstrates that Visual-Inertial methods encounter considerable challenges in such environments, particularly with scale estimation errors and a high propensity for failure due to the constrained motion patterns inherent to such settings.

Benchmarks in agricultural environments underscore the nuanced demands of applying SLAM technologies to agriculture, characterized by dynamic conditions and complex landscapes. [10] evaluates SLAM systems in a simulated vineyard, illustrating the potential of SLAM in precision agriculture. [11], covering state-of-the-art stereo Visual-Inertial SLAM systems on an arable farming dataset, sheds light on the operational challenges these technologies face within agricultural contexts, specifically in soybean fields. This study not only assesses performance metrics, such as accuracy and robustness, but addresses the adaptability of SLAM systems to the distinct characteristics of arable farming, e.g., variability of crop height and the impact of environmental factors like wind and lighting conditions.

In this paper, we identify key gaps in the existing literature that we aim to address. Notably, there is a lack of benchmarks in unstructured outdoor environments, such as gardens or parks, where SLAM technologies could significantly contribute to agricultural applications. These environments present unique challenges due to their less defined landscapes and potential for rapid changes in environmental conditions. Furthermore, existing benchmarks did not specifically focus on various driving scenarios within these landscapes, nor have they quantitatively assessed the influence of Loop Closing and its associated computational effort. This paper aims to fill these gaps by presenting a comprehensive evaluation of SLAM algorithms in these underexplored environments, providing insights into performance and computational demands across different operational scenarios.

3. Experimental Setup

In this section, we first describe the dataset used for the benchmark. Then, we subsequently introduce the selected Visual-Inertial SLAM algorithms chosen for evaluation on the aforementioned dataset. Lastly, we describe the evaluation methodology in more detail.

3.1. Data

For data recording, we utilized an unmanned ground vehicle (UGV) equipped with an Intel RealSense T265. The camera is augmented with an Inertial Measurement Unit (IMU) and captures imagery at a frequency of 30 Hz with a resolution of 848 × 800 px, while the IMU records data at 65 Hz. At a frequency of 3 to 6 Hz, 3 DOF positional ground truth data was obtained with a Leica TS16 tachymeter resulting in a positional accuracy below 1 mm.

We recorded various sequences, focusing on unstructured outdoor environments that encompass diverse garden sizes and a park-like expanse, as depicted in Fig. 1. Additionally, we considered a variety of driving scenarios, delineated into two categories: *Perimeter* and *Lane*. In the *Perimeter* scenario, the robot traverses irregular paths, completing multiple circuits around the perimeter of the designated area. Conversely, in the *Lane* scenario, the robot adheres to parallel trajectories resembling lanes, incorporating multiple 180° turns. Further information on the different locations and driving scenarios are shown in Table 1.

Fig. 1. Different dataset recording environments. Top left shows the *Garden Small*, top right *Garden Medium*, bottom left *Garden Large* and bottom right *Park* scenario.

Table 1. Dataset characteristics.

Sequence	Scenario	Duration [s]	Distance [m]
Garden Small	L	162	45.2
	P	360	167.4
Garden Medium	L	264	89.0
	P	467	167.4
Garden Large	L	351	123.4
	P	789	299.7
Park	L	210	43.7
	P	438	164.2

3.2. Algorithms

For this benchmark, we exclusively focus on open-source SLAM algorithms that utilize the ROS framework. One of the methods analyzed is ORB-SLAM3 [12], which is a multimodal feature- and optimization-based approach including an integrated Loop Closing mechanism. The system is structured into three threads: tracking thread, local mapping thread, and loop and map merging thread. The tracking thread determines the current frame's pose by minimizing reprojection errors with ORB features and selectskeyframes. The local mapping thread improves the map by adjusting keyframes locally. Lastly, the loop and map merging thread identifies revisited areas using a bag-of-words keyframe database and executes loop closures for map accuracy by applying global bundle adjustment.

Another method examined is VINS-Fusion [13], a versatile sensor fusion framework that leverages both visual and inertial cues for state estimation. This system operates through multiple modules. Initially, the state estimation module calculates the device's pose by fusing data from cameras and IMUs, ensuring accurate trajectory and orientation determination. The mapping module then refines this information by integrating environmental features, enhancing the spatial awareness of the system. In addition, VINS-Fusion incorporates Loop Closing that identifies previously visited locations and employs a global optimization process to minimize drift over time.

Another approach is OpenVINS [14], a feature- and filter-based method of visual-inertial navigation, utilizing tightly-coupled integration of camera and IMU data for precise state estimation. It employs feature tracking alongside a Kalman filter framework and sliding window optimization, efficiently handling high-frequency IMU data and visual inputs to update poses and velocities accurately. Although OpenVINS does not include Loop Closing by default, it is designed with modularity in mind, allowing for the integration of the Loop Closing module from VINS-Fusion, as applied in this study. Further, Maplab [15] can be used to enhance OpenVINS by adding Loop Closing capabilities in an offline manner. This is achieved by processing maps generated by OpenVINS through Maplab's optimization and loop closure detection tools, which can identify revisited areas and perform global map optimizations.

3.3. Evaluation

In Visual and Visual-Inertial SLAM systems, maintaining the global accuracy of the predicted trajectory is crucial. This accuracy is assessed by measuring the absolute differences between the estimated trajectory and the ground truth trajectory. Since these trajectories may be presented in different coordinate systems, they must be aligned first. Umeyama's method [16], which identifies the transformation that yields the optimal least-squares solution to map the estimated trajectory onto the ground truth trajectory, can be used to solve this alignment in closed form. We then compute the root mean squared absolute trajectory error (RMSE ATE) as the main metric to quantify the deviation between estimated and ground truth positions. It is defined by

$$RMSE(p_{1:n}, \hat{p}_{1:n}) = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \|p_i - \hat{p}_i\|^2},$$

where p is the ground truth trajectory, $\hat{p}$ the estimated trajectory after alignment, and n the total number of corresponding points in the trajectories. To verify the reliability and reproducibility of our results, we assess the SLAM algorithms by conducting five trails on each dataset sequence before calculating the average ATE using the open-source evaluation toolbox Evo [17].

In addition to evaluating accuracy, we analyze CPU as well as memory utilization to understand the computational demands of each Visual SLAM method and Loop Closing. Since we rely on ROS-based algorithms, the CPU and memory utilization of the corresponding ROS nodes can be determined accurately. All experiments have been conducted on an Intel Core i9-13900HX with 64 GB RAM, operating within a Docker container based on Ubuntu 20.04.

4. Results and Discussion

In this section, we present the outcomes of our comprehensive assessment, which investigates the effects of implementing Loop Closing on the localization accuracy and computational demands of various SLAM algorithms. This evaluation utilizes the default configurations and parameters of the open-source algorithms to ensure that the analysis accurately reflects their typical performance. All methods are used in stereo-inertial mode, operating on the data provided by the Intel RealSense T265.

4.1. Localization Accuracy

The ATE is the primary metric used to assess localization accuracy. Table 2 provides a comparison of the RMSE ATE for the SLAM algorithms tested across the different scenarios.

For ORB-SLAM3, the integration of Loop Closing showed no improvements in the *Garden Small* sequences, maintaining an ATE of 0.53 m and 0.56 m for *Perimeter* and *Lane* scenarios, respectively. A notable enhancement was observed in the *Garden Medium* sequence's *Perimeter* scenario, where the ATE decreased from 0.77 m to 0.48 m with Loop Closing, illustrating its potential to significantly mitigate drift. However, ORB-SLAM3 encountered limitations in the *Garden Medium Lane* scenario, indicating possible challenges with complex navigation paths that Loop Closing could not overcome since every trial failed. Minor reductions in ATE were also seen in larger sequences like *Garden*

Large and *Park*, demonstrating the effectiveness of Loop Closing in various contexts, albeit with limitations in certain complex scenarios.

VINS-Fusion showed accuracy improvements with Loop Closing in smaller environments, such as the *Garden Small* and *Garden Medium* sequences. However, when it comes to larger settings, the method exhibited inconsistent performance and outright failures. For instance, VINS-Fusion failed to produce results for the *Garden Large* scenarios. This indicates a distinct limitation in handling expansive and complex environments with this method. In contrast, for the *Park* sequence, albeit large, we saw a reduction in ATE with Loop Closing activated, suggesting some potential for improvement in large spaces. Nonetheless, the inconsistent outcomes and failures in certain scenarios emphasize that VINS-Fusion requires significant enhancements to reliably extend the advantages of Loop Closing across a broader range of environments.

Table 2. RMSE ATE in meters of different SLAM algorithms with and without Loop Closing in various environments and changing driving scenarios where P stands for *Perimeter* and L for *Lane*. An x indicates that the method failed.

Method	Loop Closing	Garden Small		Garden Medium		Garden Large		Park		Average	
		P	L	P	L	P	L	P	L	P	L
ORB-SLAM3	Off	0.53	0.56	0.77	x	1.12	0.82	0.94	0.52	0.84	0.63
	On	0.53	0.56	0.48	x	1.06	0.69	0.89	0.49	0.73	0.58
VINS-Fusion	Off	1.91	1.30	2.13	0.89	x	x	2.01	x	2.02	1.09
	On	1.85	1.16	2.03	0.87	x	x	1.90	x	1.93	1.02
OpenVINS & VINS-Fusion	Off	0.77	0.60	1.29	0.66	1.13	10.29	1.68	0.43	1.22	3.00
	On	0.74	0.53	1.03	0.60	0.94	10.25	1.59	0.59	1.07	2.99
OpenVINS & Maplab	Off	0.77	0.60	1.29	0.66	1.13	10.28	1.67	0.46	1.21	3.00
	On	0.58	0.48	0.56	0.51	0.72	0.76	1.02	0.45	0.72	0.55

Examining OpenVINS along with the Loop Closing module of VINS-Fusion or the offline optimization of Maplab demonstrated the most consistent and substantial improvements in localization accuracy with Loop Closing. Note that the results of OpenVINS without Loop Closing differ minimally, as all trials were made for each of the two Loop Closing variants respectively. OpenVINS paired with VINS-Fusion demonstrated a uniform enhancement across all tested scenarios, particularly reducing the ATE in the *Perimeter* scenario. More impressively, the integration of OpenVINS with Maplab yielded substantial accuracy gains, especially notable in the *Garden Large Lane* scenario where the ATE decreased from 10.28 m to 0.76 m, whereby the combination of OpenVINS and VINS-Fusion was not able to correct this substantial drift. Overall, the integration of Maplab had the greatest impact on localization accuracy and provides the best results.

4.2. Computational Efficiency

The implementation of Loop Closing affects the computational resources required by the SLAM algorithms as can be seen in Fig. 2.

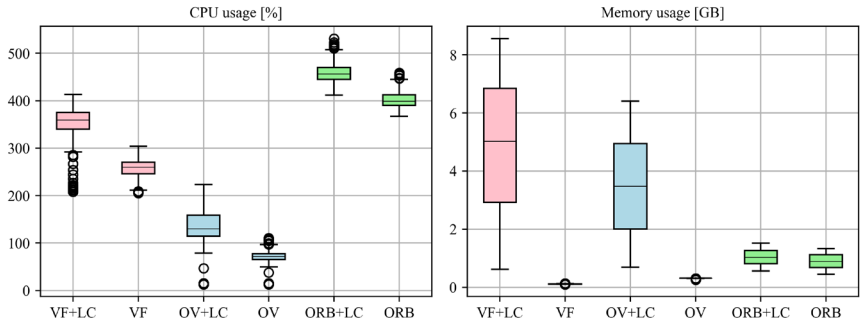


Fig. 2. CPU and memory usage of VINS-Fusion (VF) in red, OpenVINS (OV) in blue as well as ORB-SLAM3 (ORB) in green with and without Loop Closing (LC) running the algorithms on the sequence *Garden Medium Perimeter*.

ORB-SLAM3 displayed a moderate increase in computational resources when Loop Closing was enabled, with CPU usage growing from 399 % to 456 %. Similarly, its memory consumption saw an uptick from about 890 MB to 1036 MB. This increase is rather minor, considering the accuracy improvements provided by Loop Closing.

The impact of Loop Closing on VINS-Fusion was more pronounced, with CPU usage escalating from 260 % to 359 %. Memory usage exhibited a dramatic increase, from a mere 113 MB to 5031 MB with Loop Closing. This substantial jump in memory usage underscores the computational intensity of VINS-Fusion with Loop Closing, reflecting its potential limitation in resource-constrained environments.

OpenVINS with the Loop Closing module of VINS-Fusion showed a significant increase in CPU usage from 71 % without to 130 % with Loop Closing. Memory usage also increased from 309 MB to 3483 MB when Loop Closing was enabled. Although the increase in computational resources is notable, the efficiency of OpenVINS with Loop Closing, compared to VINS-Fusion, suggests a better balance between computational demand and localization accuracy. Note that there are no direct results for the combination of OpenVINS and Maplab in an online scenario, since the maps created by OpenVINS are optimized post-factum in an offline manner by Maplab.

4.3. Discussion

Regarding the impact of Loop Closing on SLAM algorithms, our study reveals that, while Loop Closing broadly enhances localization accuracy, its influence on computational demands significantly diverges across different systems. This divergence highlights an essential strategic balance that must be struck between accuracy enhancement and the management of computational resources. Further, the distinction in performance between driving scenarios, i.e., *Perimeter* and *Lane*, nuances our understanding of Loop Closing's effects, underlining the importance of context in evaluating SLAM system performance.

For ORB-SLAM3 augmented with Loop Closing, we observed a uniform improvement in localization accuracy across several scenarios, with only a moderate uptick in computational resource usage. This suggests ORB-SLAM3's viability in scenarios where enhanced precision is critical and some increase in computational resources is acceptable. In the *Perimeter* scenario, ORB-SLAM3 with Loop Closing managed to significantly reduce drift, underscoring its efficacy in environments with less structured paths. Conversely, in the *Lane* scenario, specifically in complex environments, ORB-SLAM3 faced challenges, indicating a need for refinement to handle such navigation tasks effectively.

VINS-Fusion, upon incorporating Loop Closing, displayed marked improvements in localization accuracy in smaller environments. Its performance, however, was markedly constrained in larger spaces, where it encountered significant limitations. These were particularly evident in the *Lane* scenario within expansive environments, where the computational demand surged dramatically, as reflected in the memory usage spikes. This pattern suggests that, while VINS-Fusion can be highly effective in smaller, more controlled settings, optimizing its use in larger or more complex environments requires addressing its computational inefficiencies.

OpenVINS, paired with the Loop Closing module from VINS-Fusion, showed a notable increase in localization accuracy, though at the cost of an increased computational load. This compromise may be strategic for certain applications, combining the benefits of Loop Closing with manageable computational demands. Remarkably, in both *Perimeter* and *Lane* scenarios, the integration of OpenVINS with Maplab executed in an offline manner exhibited the most considerable accuracy improvements. This suggests that for applications where delayed processing is acceptable, leveraging offline Loop Closing could achieve superior localization accuracy without the immediate computational burden.

The varied impact of Loop Closing on different SLAM systems, especially when considering the specific driving scenarios of *Perimeter* and *Lane*, underscores the critical need for tailored SLAM configurations. These adaptations should be conscientiously selected based on the operational requirements and the computational limitations at hand. The demonstrated variance in performance across different scenarios emphasizes the importance of context in deploying SLAM technologies, particularly in agricultural and other unstructured environments where navigation paths can significantly affect system performance. Our findings advocate for a nuanced approach to integrating Loop Closing in SLAM systems, one that not only considers the overall benefits in localization accuracy but also pays close attention to the unique demands of specific operations. This approach necessitates ongoing optimization efforts and strategic thinking about the use of computational resources to enhance the utility and applicability of SLAM technologies across a broad spectrum of settings.

5. Conclusion

In our exploration of the impact of Loop Closing on Visual SLAM systems within agricultural and unstructured settings, we critically evaluated ORB-SLAM3, VINS-Fusion, and OpenVINS enhanced with VINS-Fusion’s as well as Maplab's Loop Closing method. The study's results underscore the crucial role of Loop Closing in substantially improving localization accuracy, particularly in environments where unpredictability could lead to significant drift.

A significant observation from our analysis is the different computational demand imposed by Loop Closing across the evaluated SLAM systems. Notably, ORB-SLAM3, despite its generally high CPU usage, experiences minimal additional computational load upon integrating Loop Closing. This minimal increase in resource usage, in contrast to the significant decline

in ATE, firmly supports the integration of Loop Closing in ORB-SLAM3 as a worthwhile enhancement for applications requiring high precision without substantial additional computational costs. Conversely, integrating the Loop Closing module from VINS-Fusion into OpenVINS leads to a considerable uptick in memory usage, despite offering consistent improvements in localization accuracy. This increased demand for memory resources highlights a trade-off between achieving higher accuracy and the associated computational costs. Nevertheless, OpenVINS with Maplab's offline Loop Closing emerges as the standout in terms of accuracy enhancement, albeit without available data on computational performance due to its offline processing nature. Moreover, VINS-Fusion's performance is notably lower than the other two SLAM methods, encountering considerable difficulties in larger environments despite the benefits derived from Loop Closing. The computational demand for VINS-Fusion presents substantial challenges, with Loop Closing offering accuracy benefits but at the highest computational cost among the methods evaluated.

Future directions for research include assessing the computational efficiency of offline Loop Closing methods, such as those offered by Maplab, and their integration into real-time systems. Exploring deep learning-based Loop Closing techniques could also offer avenues for accuracy improvements with potentially reduced computational demands. Expanding these findings to broader applications and integrating multi-modal sensor data with SLAM systems could further enhance localization robustness and accuracy across various scenarios.

References

[1]. D. Sharafutdinov, M. Griguletskii, et al., Comparison of modern open-source visual SLAM approaches, *Journal of Intelligent & Robotic Systems*, Vol. 107, 2023, 43.

[2]. M. Servières, V. Renaudin, et al., Visual and visual-inertial SLAM: state of the art, classification, and experimental benchmarking, *Journal of Sensors*, Vol. 2021, 2021, 2054828.

[3]. A. Merzlyakov, S. Macenski, A comparison of modern general-purpose visual SLAM approaches, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'21)*, Prague, Czech Republic, 2021, pp. 9190-9197.

[4]. J. Delmerico, D. Scaramuzza, A benchmark comparison of monocular visual-inertial odometry algorithms for flying robots, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'18)*, Brisbane, QLD, Australia, 2018, pp. 2502-2509.

[5]. M. Burri, J. Nikolic, et al., The EuR oC micro aerial vehicle datasets, *The International Journal of Robotic Research*, Vol. 35, Issue 10, 2016, pp. 1157-1163.

[6]. A. Geiger, P. Lenz, et al., Are we ready for Autonomous Driving? The KITTI vision benchmark suite, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*, Providence, RI, USA 2012, pp. 3354-3361.

[7]. J. Sturm, N. Engelhard, et al., A benchmark for the evaluation of RGB-D SLAM systems, in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (*IROS'12)*, Vilamoura-Algarve, Portugal, 2012, pp. 573-580.

[8]. R. Li, Y. Lou, W. Song, et al., experimental evaluation of monocular visual-inertial SLAM methods for freight railways, *IEEE Sensors Journal*, Vol. 23, Issue 19, 2023, pp. 23282-23293.

[9]. B. Joshi, S. Rahman, et al., Experimental comparison of open source visual-inertial-based state estimation algorithms in the underwater domain, in *Proceedings of the International Conference on Intelligent Robots and Systems* (*IROS'19)*, Macau, China, 2019, pp. 7227-7233.

[10]. I. Hroob, R. Polvara, et al., Benchmark of visual and 3D lidar SLAM systems in simulation environment for vineyards, in *Proceedings of the 22nd Annual Conference Towards Autonomous Robotic Systems (TAROS'21)*, Lincoln, UK, 2021, pp. 168-177.

[11]. J. Cremona, R. Comelli, et al., Experimental evaluation of visual-inertial systems for arable farming, *Journal of Field Robotics*, Vol. 39, Issue 7, 2022, pp. 1121-1135.

[12]. C. Campos, R. Elvira, et al., ORB-SLAM3: an accurate open-source library for visual, visual-inertial and multi-map SLAM, *IEEE Transactions on Robotics,* Vol. 36, Issue 6, 2021, pp.1874-1890.

[13]. T. Qin, J. Pan, et al., A general optimization-based framework for local odometry estimation with multiple sensors, *ArXiv Preprint*, 2019, abs/1901.03638.

[14]. P. Geneva, K. Eckenhoff, et al., OpenVINS: A research platform for visual-inertial estimation, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'20)*, Paris, France, 2020, pp. 4666-4672.

[15]. A. Cramariuc, L. Bernreiter, et al., Maplab 2.0 – a modular and multi-modal mapping framework, *IEEE Robotics and Automation Letters,* Vol. 8, Issue 2, 2023, pp. 520-527.

[16]. S. Umeyama, Least-squares estimation of transformation parameters between two point patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI*), Vol. 13, Issue 4, 1991, pp. 376-380.

[17]. M. Grupp, Evo: Python package for the evaluation of odometry and SLAM. https://github.com/ MichaelGrupp/evo

(056)

Complex Wavelet-enhanced Convolutional Neural Networks for Electrocardiogram-based Detection of Paroxysmal Atrial Fibrillation

A. Al Fahoum
Biomedical Systems and Informatics Engineering Dept., Hijjawi Faculty for Engineering Technology,
Yarmouk University, Irbid, 21163, Jordan
E-mail: afahoum@yu.edu.jo

Summary: This research focuses on developing an automated classification system to distinguish between Paroxysmal Atrial Fibrillation (PAF) and Normal Sinus Rhythm (NSR) using ECG signals. The methodology leverages Continuous Wavelet Transform (CWT) for signal processing and Convolutional Neural Networks (CNN) for machine learning, achieving an impressive 94 % accuracy. This innovative approach highlights the potential of integrating DSP and ML for enhanced PAF detection. Key findings demonstrate a significant improvement in detection accuracy, showcasing the potential of this innovative approach. The study's significance lies in its contribution to automatic medical diagnostics, offering a promising direction for future research and potential clinical applications in the efficient and timely identification of PAF.

Keywords: Deep learning (DL), Continuous wavelet transform (CWT), Electrocardiogram (ECG) and Atrial disease classification (ADF).

1. Introduction

PAF is an intermittent type of arrhythmia that poses significant health risks and challenges in detection. Accurate and timely identification is crucial for effective management. Despite the availability of long-term ECG recording devices, the transient nature of PAF episodes makes detection labor-intensive and time-consuming. This study aims to leverage the abundance of ECG data and advanced digital signal processing (DSP) and machine learning (ML) techniques to improve the accuracy and efficiency of PAF detection [1].

PAF represents a significant challenge within the landscape of cardiovascular diseases, characterized by sudden, intermittent episodes of atrial fibrillation (AF) that can revert to normal sinus rhythm without intervention. This condition not only increases the risk of stroke and heart failure but also complicates the diagnostic process due to its transient nature [2]. Current detection methods largely rely on ECG, which, while effective in continuous monitoring, often require manual interpretation and may miss shorter episodes of PAF due to their sporadic occurrence [3].

The advent of DSP techniques has opened new avenues for the analysis of ECG signals, providing tools that can enhance the sensitivity and specificity of PAF detection [4]. CWT, a method known for its ability to decompose non-stationary signals into time-frequency representations, offers a nuanced approach to identifying the complex patterns characteristic of PAF episodes. This technique allows for the detailed analysis of ECG signals, highlighting variations in heart rhythm that are indicative of atrial fibrillation [5-9].

Moreover, the integration of deep learning methods, particularly CNN, with DSP techniques marks a significant advancement in the field [7, 9]. CNNs are adept at recognizing patterns within large datasets, making them an ideal tool for analyzing the intricate features extracted through CWT [5]. By training these networks on labeled datasets of ECG signals, it becomes possible to automate the detection process, significantly reducing the time and expertise required to diagnose PAF.

The combination of CWT and CNN not only leverages the strengths of both approaches but also addresses the limitations of traditional detection methods. This synergy enhances the accuracy of PAF detection, offering a promising solution to the challenges posed by its paroxysmal nature. The potential of this integrated approach to improve patient outcomes through early and accurate detection is significant, underscoring the importance of continued research and development in this area.

2. Materials and Methods

The Materials and Methods section is structured to detail the comprehensive approach taken in utilizing ECG data for the detection of PAF through advanced DSP techniques and ML models, particularly focusing on CWT and CNN.

The study leveraged the Physionet.org database, renowned for its extensive repository of high-quality, freely available physiological signals. The MIT-BIH Atrial Fibrillation Database is a collection of 25 long-term ECG recordings of human subjects with PAF. It offers a high-resolution temporal depiction of cardiac activity and is useful for the automated detection of AF and related arrhythmias. The database includes rhythm annotation files and beat annotation files, providing a foundation for exploring robust automated AF detection methods against common QRS errors. From this database, 25 PAF and 25 NSR

ECG segments of both 1-minute and 5-minute durations were extracted. ECG signals are sampled at a frequency of 250 samples per second, which provides a detailed and accurate representation of the timing of cardiac activity. This sampling rate and a 12-bit resolution within a ±10 millivolt range enable precise examination of the ECG waveforms. These segments were specifically chosen to represent a diverse range of PAF and Normal Sinus Rhythm (NSR) instances, ensuring a dataset comprehensive enough to train and validate the ML model effectively [10].

The utilization of CWT in MATLAB was employed to generate amplitude and phase imaging from ECG data, with a focus on classifying PAF and NSR. The ECG recordings were preprocessed by applying a finite impulse response filter (FIR) with a frequency range of 0.01 to 100 Hz to maintain linear phase and signal-time domain features. The signals were then normalized using a min-max approach. Afterwards, this method simplified the use of CWT to identify crucial features in both the time and frequency domains, which were essential for identifying unique patterns associated with PAF.

The CWT was performed using a complex Morlet wavelet, chosen for its excellent ability to balance temporal and frequency localization. The analysis scales varied from 1 to 100, which is equivalent to the maximum available bandwidth of the processed signal. This allowed for a thorough breakdown of the specified frequency range.

The red-green-blue (RGB) images were generated to display the magnitude and phase of the CWT coefficients. These images were generated without axes and empty white space, enhancing the visibility of the visual patterns in the data. The purpose of this amplitude-phase visualization method is to ease the comprehension of the varying characteristics of ECG signals. Furthermore, it provides a valuable means to observe and comprehend the evolving patterns associated with PAF, thereby enhancing the precision of diagnostic classification models. These images are the foundational data for further analysis through deep learning techniques [7]. Fig. 1 shows cropped image of the PAF segment.

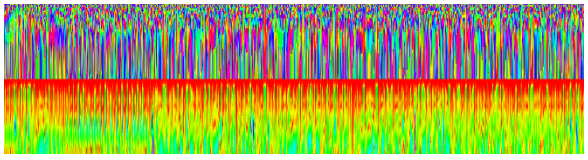


Fig. 1. Two subplots to visualize the amplitude and phase of the Continuous Wavelet Transform (CWT) on the dataset.

The CNN architecture was meticulously designed. The CNN model was structured to include several convolutional layers, pooling layers, and fully connected layers, each serving a specific purpose in feature extraction and classification. A MATLAB code outlines the development and training of the CNN designed for binary classification tasks, specifically aimed at detecting the presence of PAF from NSR data. This network is part of a broader workflow involving image preprocessing, data loading, and model training. Below is a description of the network and the workflow, followed by a flowchart description.

Image Preprocessing

The preprocessing step involves resizing images stored in a specified directory to a uniform dimension of 1024×1024 pixels using the bilinear resampling filter. This uniformity is crucial for ensuring consistent input sizes for the CNN model. The resized images are then saved to an output folder for subsequent processing.

Data loading and preparation

The data loading function retrieves images from the output folder, converting them into a 3-array format suitable for model input. It also assigns binary labels based on the presence of specific substrings in the filenames, indicative of the class to which each image belongs. The dataset is randomized and partitioned into training, validation, and testing sets with respective ratios, ensuring a comprehensive evaluation framework. Additionally, the pixel values of the RGB image data are normalized to the range [0, 255] to facilitate model convergence during training.

CNN Architecture

The CNN model is constructed using TensorFlow's Keras API, comprising an input layer tailored for 1024×1024 RGB images, followed by a series of convolutional and max-pooling layers. These layers are designed to extract and down-sample features from the input images progressively. The network architecture has convolutional layers with 32, 64, and 128 filters, and then max-pooling layers that reduce the spatial dimensions of the feature maps. This makes the computations simpler and lowers the risk of overfitting. Following the convolutional base, the network transitions to a flattening layer, converting the 2D feature maps into a 1D feature vector. This vector feeds into a sequence of densely connected layers (or fully connected layers), culminating in a binary classification output layer with a sigmoid activation function. The model employs the Adam optimizer and binary cross-entropy loss function, which are appropriate for binary classification tasks.

Training Process

The model undergoes training over a maximum range of 60 epochs using the prepared training dataset. The accuracy metric monitors the effectiveness of the training process and provides insights into the model's performance across epochs.

Flowchart Description

1. **Image Preprocessing:**
   - Resize images to 1024x1024 pixels;
   - Save resized images to the output folder.

2. **Data Loading:**
   - Convert amplitude and phase arrays of CWT into RGB images;
   - Assign binary labels based on classification requirements;
   - Normalize pixel values.
3. **Data Partitioning:**
   - Randomize the dataset;
   - Split into training, validation, and testing sets.
4. **Model Construction:**
   - Input layer for 1024x1024 RGB images;
   - Convolutional and max-pooling layers;
   - Flattening layer to convert 2D feature maps to 1D vectors;
   - Dense layers for classification;
   - Binary output layer with sigmoid activation.
5. **Model Training:**
   - Train the model over 32 epochs;
   - Monitor training accuracy.

The model training involved feeding the magnitude and phase images produced by CWT into the CNN. A dataset split strategy was employed, typically allocating 70 % of the data for training and 10 % for validation, and 20 % for testing purposes. This split ensured a robust training process while leaving an adequate portion of data for model validation. Additionally, techniques such as data augmentation, regularization, and dropout were applied to prevent overfitting and improve the model's generalization capability.

Validation of the CNN model's performance was conducted using the reserved segment of the dataset. Key performance metrics, including accuracy, sensitivity, specificity, and F1 score, and others were calculated to assess the model's effectiveness in correctly identifying PAF instances from NSR [11]. This step was critical in determining the model's clinical applicability and reliability in real-world scenarios.

3. Results and Discussion

The model achieved an overall 94 % accuracy rate after training with a split of 70 % for training, 10 % for validation, and 20 % for testing. The algorithm has an early stopping condition satisfied when 100 % training accuracy is obtained and maintained for at least three consecutive epochs to prevent overfitting. Various metrics are used to study the model's performance [11]. The model achieved an overall accuracy across all classes of 94 % of predictions being correct. This high accuracy indicates the model is generally reliable across various situations. The overall error rate across all classes of 6 % suggests that the model's predictions are incorrect. This complements the overall accuracy and is relatively low. Recall, or sensitivity, measures the proportion of actual positives correctly identified. A recall of 96 % indicates the model is highly effective at catching the relevant cases. With 92 % specificity, the model is very good at avoiding false alarms. A precision of 92.31 % shows the model's high reliability in its positive predictions. A false-positive rate of 8 % is relatively low but important in contexts where false alarms are costly. The F1 Score is the harmonic mean of precision and recall, providing a balance between the two in situations where one may be more important than the other. An F1 score of 94.12 % is excellent, indicating a strong balance between precision and recall. A very high MCC of 0.8807 indicates a strong correlation between the model's predictions and the actual classifications. Cohen's Kappa of 0.88 indicates almost perfect agreement beyond chance, suggesting the model's predictions align closely with the expected outcomes. Overall, the model shows excellent performance across various metrics, indicating it is highly effective in its classifications, with a strong balance between recognizing positive cases (high recall) and accurately identifying negative cases (high specificity). The low error rates and high MCC and Kappa scores underline the model's reliability and consistency in prediction accuracy. However, depending on the application, even a small number of false positives or negatives could be critical, so these areas may still require attention.

This high accuracy demonstrates the effectiveness of combining CWT for signal processing and CNN for classification. The study discusses the potential of this methodology to improve PAF detection and suggests future research directions, including expanding the dataset for further accuracy improvements. This method for predicting AF uses simple, clear ECG parsing algorithms along with complex wavelet analysis using CNN networks that were specially made for this purpose. It stands in comparison with other techniques that assess current AF and predict its future occurrences. Even though there have been improvements in measuring AF that make them more sensitive and specific, handheld devices are still not very good at catching PAF [1]. On the other hand, using machine learning models to combine demographic data, simple clinical assessments, and plasma biomarkers can give more in-depth information about diseases, but it is not as sensitive or specific as direct methods. Moreover, more refined electrophysiological strategies are under development, utilizing machine learning to analyze historical clinical ECG data or databases [12-13]. There is a wide range of specificity and sensitivity (82 % – 93 %) for these methods, which look for atrial premature beats and other ECG abnormalities or intervals of atrial or ventricular depolarizations [1]. However, they need longer recording times. Given the intricate nature of Paroxysmal Atrial Fibrillation (PAF) and the sporadic approach to monitoring, there is a possibility that certain individuals categorized as controls might have had undetected occurrences of PAF [14]. While all confirmed cases experienced at least one PAF episode within the study duration, it's conceivable that some controls underwent PAF episodes at more extended intervals. If there is too much noise, electrical disturbances from outside sources, like

electromyographic activity coming from a participant, could change the ECG signal and make the results less accurate. The pilot nature of this study constrained the inclusion of real patients at this stage to fine-tune the algorithm and ensure robust results. Nonetheless, ventricular arrhythmias or waveform irregularities like bigeminy or T-wave alterations could introduce confounding variables, leading to false positives [14]. Within the scope of this study, clinical assessment excluded such anomalous traces; however, this falls short of the mark for a wholly autonomous procedure. Anticipation is set on the subsequent phase to integrate an initial ECG segmentation step, perhaps employing an ECG segmentation algorithm complemented by autocorrelation analysis, reconstructed phase space, bispectrum and biocoherence to sieve out traces bearing these potential confounders at further stages, which could enhance the results [15-18].

4. Conclusions

The study successfully created an automated classification system using ECG signals to differentiate between PAF and NSR, achieving a high accuracy rate. This demonstrates the viability of combining DSP and ML techniques for medical diagnostics, offering promising directions for future research and potential clinical applications.

References

[1]. V. Alexeenko, J. A. Fraser, D. Abasolo, Ch. H. Fry, R. I. Jabr, Prediction of paroxysmal atrial fibrillation from complexity analysis of the sinus rhythm ECG: a retrospective case/control pilot study, *Frontiers in Physiology*, Vol. 12, 2021, 570705.

[2]. C. Ma, *et al*., A review on atrial fibrillation detection from ambulatory ECG, *IEEE Transactions on Biomedical Engineering*, Vol. 71, Issue 3, 2024, pp. 876-892.

[3]. Y. Zou, X. Yu, S. Li, et al. A generalizable and robust deep learning method for atrial fibrillation detection from long-term electrocardiogram, *Biomedical Signal Processing and Control*, Vol. 90, 2024, 105797.

[4]. I. Haq, K. Liu, et al., Artificial intelligence-enhanced electrocardiogram for arrhythmogenic right ventricular cardiomyopathy detection, *European Heart Journal – Digital Health*, Vol. 5, Issue 2, March 2024, pp. 192-194.

[5]. A. Al Fahoum, A. Zyout, Enhancing early detection of schizophrenia through multi-modal EEG analysis: a fusion of wavelet transform, reconstructed phase space, and deep learning neural networks, in *Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI'23)*, 2023, pp. 38-41.

[6]. A. Al Fahoum, A. Abu Al-Haija, H. A. Alshraideh, Identification of coronary artery diseases using photoplethysmography signals and practical feature selection process, *Bioengineering,* Vol. 10, Issue 2, 2023, 249.

[7]. A. Al-Fahoum, I. Howitt, Combined wavelet transformation and radial basis neural networks for classifying life-threatening cardiac arrhythmias, *Medical & Biological Engineering & Computing*, Vol. 37, 1999, pp. 566-573.

[8]. L. Khadra, A. Al-Fahoum, H. Al-Nashash, Detection of life-threatening cardiac arrhythmias using the wavelet transformation, *Medical and Biological Engineering and Computing*, Vol. 35, 1997, pp. 626-632.

[9]. A. Al Fahoum, A. Al Omari, G. Al Omari, A. Zyout, PPG signal-based classification of blood pressure stages using wavelet transformation and pre-trained deep learning models, *Computing in Cardiology*, Vol. 50, 2023, pp. 1-4.

[10]. PhysioNet, AFPD: An open-access database for the study of atrial fibrillation, https://physionet.org/content/afpdb/1.0.0/

[11]. A. Al Fahoum, enhanced cardiac arrhythmia detection utilizing deep learning architectures and multi-scale ECG Analysis, *Tuijin Jishu/Journal of Propulsion Technology*, Vol. 44, Issue 6, 2023, pp. 5539-5548.

[12]. A. Han, O. Kwon, et al., Evaluating the risk of paroxysmal atrial fibrillation in noncardioembolic ischemic stroke using artificial intelligence-enabled ECG algorithm, *Frontiers in Cardiovascular Medicine*, Vol. 9, 2022, 865852.

[13]. Y. Kim, G. Joo, K. Jeon, et al., Clinical applicability of an artificial intelligence prediction algorithm for early prediction of non-persistent atrial fibrillation, *Frontiers in Cardiovascular Medicine*, Vol. 10, 2023.

[14]. L. Khadra, A. Al-Fahoum, S. Binajjaj, A new quantitative analysis technique for cardiac arrhythmia using bispectrum and bicoherency, in *Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, San Francisco, CA, USA, 2004, pp. 13-16.

[15]. K. Lee, H. Ko, et al., Explainable paroxysmal atrial fibrillation diagnosis using electrocardiogram with artificial intelligence, *Europace*, Vol. 25, 2023, euad122-526.

[16]. A. Al-Fahoum, L. Khadra, Combined bispectral and bicoherency approach for catastrophic arrhythmia classification, in *Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2006, pp. 332-336.

[17]. A. Al-Fahoum, A. Al-Fraihat, A. Al-Araid, Detection of cardiac ischaemia using bispectral analysis approach, *Journal of Medical Engineering & Technology,* Vol. 386, 2014, pp. 311-316.

[18]. A. Al-Fahoum, A. Qasaimeh, ECG arrhythmia classification using simple reconstructed phase space approach, in 2006 Computers in Cardiology, 2006, *IEEE*, pp. 757-760.

(057)

Role of fMRI Denoising for Classification of Schizophrenia from Functional Brain Connectivity

J. Hlinka [1,2], D. Tomeček [1,3], M. Kolenič [1], B. Rehák Bučková [2,4], J. Tintěra [1,5], J. Horáček [1] and F. Španiel [1]

[1] National Institute of Mental Health, Topolová 748, 250 67 Klecany, Czech Republic
[2] Institute of Computer Science of the Czech Academy of Science, Pod vodarenskou vezi 271, 18200, Prague 8, Czech Republic
[3] Faculty of Electrical Engineering, Czech Technical University, Technická 2, 166 27 Prague 6, Czech Republic
[4] Donders Institute for Brain, Cognition and Behaviour, Radboud University, Kapittelweg 29, 6525 EN Nijmegen, Netherlands
[5] Institute of Clinical and Experimental Medicine, Vídeňská 1958/9, 140 21 Prague 4, Czech Republic
Tel.: + 420 266053808
E-mail: hlinka@cs.cas.cz

Summary: This study explores the impact of denoising strategies on classifying first-episode psychosis (FEP) patients from healthy controls using functional connectivity measures derived from fMRI data. Leveraging a dataset of 100 FEP patients and 90 healthy controls, the research evaluates how different preprocessing approaches—ranging from raw data to moderate and stringent denoising – affect the classification accuracy when applying logistic regression on dimension-reduced features via PCA. The findings reveal that both moderate and stringent denoising methods significantly enhance classification performance compared to using raw data, with moderate denoising reaching an 82 % accuracy with 24 principal components and stringent denoising achieving 81 % accuracy with 45 components. The study underscores the importance of denoising in improving the reliability of functional connectivity measures for schizophrenia classification. However, it also suggests that the choice between moderate and stringent denoising may not be critical, as combining multiple strategies did not substantially improve performance. This research highlights the potential of optimized fMRI data preprocessing in psychiatric diagnosis, providing insights into the neurodevelopmental and neurodegenerative processes underlying schizophrenia.

Keywords: Functional connectivity, Schizophrenia, fMRI, Classification, Denoising.

1. Introduction

1.1. Motivation

Although a neurodevelopmental hypothesis for schizophrenia complemented by neurodegenerative processes is well established, the link between these processes and the specific brain dysfunction underlying schizophrenia psychopathology is not clear [1]. However, in general schizophrenia is widely conceptualized as a disconnection disease, i.e. affecting and stemming from abnormal connectivity, although concurrent changes in other anatomical features such as decreased gray matter thickness are commonly observed. Indeed, various changes of brain connectivity have been reported. Apart from distributed changes of white matter integrity [2] affecting the structural substrate of brain connectivity, the reported effects include quite prominent changes in so called functional connectivity – the statistical dependence between the activity of remote brain regions. Changes in functional connectivity have thus been widely used as bio-markers for construction of classifiers distinguishing healthy subjects from patients with schizophrenia.

1.2. Key Challenges and State-of-art

However, it is far from understood how to optimally select functional connectivity descriptors, as well as to build a classifier, to obtain optimal performance. Apart from standard approaches such as applying logistic regression or linear Support Vector Machines (SVM) directly to the functional connectivity indices, a range of studies aims to improve the performance by either applying advanced machine learning approaches such as deep networks [3], or using sophisticated Persistent Homology features based on the Topological Data Analysis approaches [4]. However, despite the theoretical promises, robust and substantial improvements by these approaches have not yet been established, with the added value of many of these advanced tools being in the current data situation at most incremental, if any [3].

This might be attributed to the high levels of noise in functional magnetic resonance imaging (fMRI) data. Importantly, the noise is typically a structured noise that might potentially confound the results, such as the increased amount of head motion under some conditions [5], leading to systematic bias in the

functional connectivity matrices towards atypical multivariate functional connectivity patterns [6]. Another contributing factor is of course the problem of relatively small sample sizes (in terms of number of subjects), compared to relatively high number of features, typically available in neuroimaging studies [7].

1.3. Current Study Plan

In the current study, we use a comparably large dataset of first episode of psychosis data (100 patients and 90 healthy controls with MRI data acquired with the same procedures) to study the effect of choice of denoising strategies on the performance of classification. Functional connectivity is in line with the literature quantified by Pearson's correlation coefficient [8]. Given the relatively modest sample size, we use robust dimension reduction method, i.e. principal component analysis (PCA) and classical classifier choice (logistic regression), a combination that has been previously shown to perform on par with more sophisticated tools in this particular context [9, 10].

2. Data and Methods

2.1. Study Overview and Samples

In total, 190 subjects participated in the study; 100 First episode psychosis (FEP) patients (mean age = 28.75, SD = 6.83, 42 females/58 males) and 90 healthy volunteers serving as controls (mean age = 27.81, SD = 6.82, 50 females/40 males). There were no significant differences between the patient and control samples in age and sex. In the patient group, at the time of the MRI scan, the average duration of untreated psychosis was 3.23 months (S.D. = 4.82), and the average duration of antipsychotic treatment was 2.29 months (S.D. = 4.58). The study design was approved by the local Ethics Committee of the Institute of Clinical and Experimental Medicine and the Psychiatric Center Prague. All subjects provided written informed consent after receiving a complete description of the study.

The FEP patients were diagnosed according to ICD-10 criteria and structured MINI International Neuropsychiatric Interview. FEP subjects were investigated during their first hospitalization and were considered as FEP if they fulfilled these criteria: a) first hospitalization for schizophrenia, and b) clinical interview identified first psychotic and/or prodromal symptoms of psychosis not earlier than 24 months ago (mean = 5.90 months, SD = 6.16).

The resting fMRI was performed at the initial stage of second-generation antipsychotic therapy (mean 10 weeks of medication at the time of resting state fMRI. Ninety healthy control subjects (HC) were recruited via a local advertisement; they had a similar socio-demographic background as the FEP to whom they were matched by age and sex.

The healthy controls had a slightly higher number of years of education than the FEP (15.64, SD = 3.34 and 13.48, SD = 2.28, $t = 4.466$, $p < 0.001$). Healthy controls were evaluated with MINI and were excluded if they had a lifetime history of any psychiatric disorder or a family history of psychotic disorders. Other exclusion criteria for both groups included a history of seizures or significant head trauma, mental retardation, a history of substance dependence, and any MRI contraindications. The protocol was approved by the institutional review boards of the National Institute on Mental Health, Klecany. Written informed consent was obtained from all participants.

2.2. fMRI Data Acquisition

Scanning was performed with a 3T MRI scanner (Siemens Magnetom Trio) located at the Institute of Clinical and Experimental Medicine in Prague, Czech Republic. Functional images were obtained using T2-weighted echo-planar imaging (EPI) with blood oxygenation level-dependent (BOLD) contrast using SENSE imaging. GE-EPIs (TR/TE = 2000/30 ms, flip angle = 70°) consisted of 35 axial slices acquired continuously in sequential decreasing order covering the entire cerebrum (voxel size = 3×3×3 mm, slice dimensions 48x64 voxels). The next 400 functional volumes were used for the analysis. A three-dimensional high-resolution MPRAGE T1-weighted image (TR/TE = 2300/4.63 ms, flip angle 10°, voxel size = 1×1×1 mm) covering the entire brain was acquired at the beginning of the scanning session and used for anatomical reference.

2.3. Data Preprocessing, Brain Parcellation and FC Analysis

Functional MRI is a neuroimaging method that is based on measuring blood oxygen level-dependent signal. One of the typical features of the fMRI data is the noise which is present in the raw BOLD signal [11]. The presence of noise in the fMRI data significantly limits the reliability of functional connectivity measures [12]. Typical artifacts, such as subject movements, arterial pulsation, respiration, and also hardware of the MRI scanner itself, induce non-neural temporal correlations in the BOLD, and relatively sophisticated data preprocessing is thus warranted to obtain maximize the level to which the functional connectivity estimates reflect the underlying neuronal dynamics.

The resting state fMRI data were corrected for head movement (realignment and regression) and registered to MNI standard stereotactic space (Montreal Neurological Institute, MNI) with a voxel size of 2×2×2 mm by a 12 parameter affine transform maximizing normalized correlation with a customized EPI template image. This was followed by segmentation of the anatomical images in order to create subject-specific white-matter and CSF masks.

The resulting anatomical images and masks were spatially normalized to a standard stereotactic MNI space with a voxel size of 2×2×2 mm.

The denoising steps included regression of six head-motion parameters (acquired while performing the correction of head-motion) with their first-order temporal derivatives and five principal components of white matter and cerebrospinal fluid. The CONN toolbox has implemented a component-based noise correction method (CompCor) that in the default setting performs PCA dimensionality reduction of white matter and cerebrospinal fluid time series derived from particular regions [13]. The CompCor method uses noise regions of interest (ROIs) acquired while segmenting each subject's high-resolution anatomical images. Time series from defined regions of interest were additionally linearly detrended in order to remove possible signal drift and finally filtered by a band-pass filter with cutoff frequencies 0.004 – 0.1 Hz. We shall refer to this preprocessing setup as the *stringent* denoising scheme.

As an alternative denoising pipeline, closer to the practice in some studies, we used a more *moderate* denoising scheme in which we used six head-motion parameters without their first-order derivatives and only the mean time-series of white-matter and cerebrospinal fluid (instead of the 5 PCA components for each compartment as in default CompCor pipeline described above). This alternative denoising pipeline was performed without explicit linear detrending, however time series were also finally filtered by a band-pass filter with cutoff frequencies 0.004 – 0.1 Hz. As a benchmark, we also used data without the denoising steps described for the stringent or moderate scheme; this dataset version is further denoted as *raw* (albeit they naturally include the basic steps of motion correction and normalization to MNI template to allow meaningful extraction of region-based average activation time series).

2.4. Analysis

The functional connectivity analysis was carried out using the CONN toolbox (Gabrieli Lab. McGovern Institute for Brain Research Massachusetts Institute of Technology, USA; www.nitrc.org/projects/conn). CONN is a complex Matlab-based toolbox for the analysis of functional connectivity in resting-state or task-based fMRI data [14]. The toolbox uses standard SPM (Wellcome Department of Imaging Neuroscience, London, UK; www.fil.ion.ucl.ac.uk/spm) modules for data preprocessing.

The regional mean time series were estimated by averaging voxel time series within each of the 90 brain regions (excluding the cerebellar regions) comprising the Automated Anatomical Labeling (AAL) atlas [15]. To quantify the whole-brain pattern of functional connectivity, we performed a ROI-to-ROI connectivity analysis by computing, for each subject, the Pearson's correlation matrix among the regional mean time series.

The classification tasks were carried out in the MATLAB environment. For classification of healthy vs patients, we used logistic regression, applied to dimension reduced features of each of the three datasets (obtained by the raw, moderate, or stringent denoising). The dimension reduction was carried out by applying principal component analysis to the 4005 functional connectivity features. To avoid overfitting, the performance was evaluated in a leave-one-out evaluation scheme. The optimal threshold was selected within each fold using the Youden's J statistic. To map the effect of strictness of dimension reduction, we varied the number of principal components extracted and used for the analysis between 1 and 190.

3. Results

The results of the classifier on the three types of data preprocessing differed substantially, see Fig. 1 for visualization of the results. In particular, for the raw data, the maximum accuracy reached was 74 %, achieved for using 44 principal components, and then gradually decreasing due to overfitting with too many input features. Similar overall picture was observed for the moderate denoising, however the accuracy reached was 82 %, achieved already for using 24 principal components. Finally, for the stringent denoising, the accuracy reached was 81 %, achieved for using 45 principal components. Notably the results were relatively robust with respect to choose of number of components. Combining data from multiple denoising strategies achieved performance comparable with that of the well-performing moderate or stringent strategies.

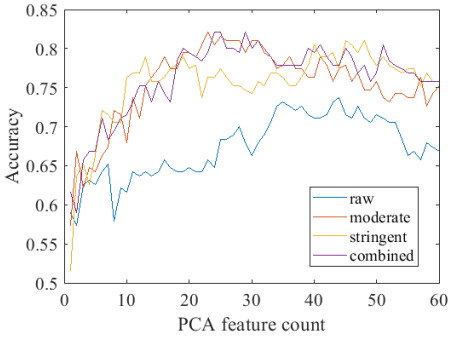


Fig. 1. Accuracy of classification of first episode psychosis patients versus healthy controls from resting state functional magnetic resonance imaging functional connectivity features using logistic regression, as a function of dimension reduction (number of PCA components) and preprocessing option. For visualization purposes only the results for the range of principal components of 1 to 60 are shown; the accuracy gradually decreased for even larger component count due to natural overfitting.

4. Discussion and Conclusions

Most of the applied settings provided clearly above chance performance in the classification task, however

the performance depended crucially on the analysis options. Interestingly, even the *raw*, minimally preprocessed data provided non-negligible classification performance. However, both evaluated standard processing approaches provided about ten percent improvement in accuracy.

Surprisingly, both the moderate and stringent denoising strategies obtained comparable performance, despite substantial change in the mean difference between the groups (results not shown). Our analysis has thus shown, that denoising has a beneficial effect on the performance of classification of schizophrenia from functional connectivity, while the selection among suitable strategies may not be so crucial.

A key role was played by the number of principle components of the functional connectivity feature set that were used in the classification. Despite some dimension reduction of the original 4005 features proved clearly beneficial, unlike to our previous study [10] in multiple sclerosis, reduction to very small number of components (below 10) was not competitive.

Finally, perhaps unfortunately, combining data with multiple denoising strategies did not substantially help the classification performance, yielding yet another unsuccessful attempt at substantially improving the classification of schizophrenia patients from healthy controls using resting state functional magnetic resonance imaging functional connectivity features.

Acknowledgments

This work was supported by the Czech Health Research Council Project No. NU21-08-00432, by the long-term strategic development financing of the Institute of Computer Science (RVO:67985807) of the Czech Academy of Sciences, and by the ERDF-Project Brain dynamics, No. CZ.02.01.01/00/22_008/ 0004643.

References

[1]. M. R. Dauvermann, G. Lee, et al., Glutamatergic regulation of cognition and functional brain connectivity: insights from pharmacological, genetic and translational schizophrenia research, *British Journal of Pharmacology*, Vol. 174, Issue 19, 2017, pp. 3136-3160.

[2]. T. Melicher, T. J. Horacek, et al., White matter changes in first episode psychosis and their relation to the size of sample studied: A DTI study, *Schizophrenia Research*, Vol. 162, Issues 1-3, 2015, pp. 22-28.

[3]. U. Mahmood, Z. Fu, et al., A deep learning model for data-driven discovery of functional connectivity, *Algorithms*, Vol. 14, Issue 3, 2021, a14030075.

[4]. L. Caputi, A. Pidnebesna, et al. Promises and pitfalls of topological data analysis for brain connectivity analysis. *NeuroImage*, Vol. 238, 2021, 118245.

[5]. J. Hlinka, C. Alexakis, et al., Is sedation-induced BOLD fMRI low-frequency fluctuation increase mediated by increased motion?, *Magnetic Resonance Materials in Physics, Biology and Medicine*, Vol. 23, Issues 5-6, 2010, pp. 367-374.

[6]. J. Kopal, A. Pidnebesna, et al., Typicality of functional connectivity robustly captures motion artifacts in rs-fMRI across datasets, atlases, and preprocessing pipelines, *Human Brain Mapping*, Vol. 41, Issue 18, 2020, pp. 5325-5340.

[7]. R. A. Poldrack, C. L. Baker, et al., Scanning the horizon: Towards transparent and reproducible neuroimaging research, *Nature Reviews Neuroscience*, Vol. 18, Issue 2, 2017, pp. 115-126.

[8]. J. Hlinka, M. Paluš, et al., Functional connectivity in resting-state fMRI: Is linear correlation sufficient?, *NeuroImage*, Vol. 54, Issue 3, 2011, pp. 2218-2225.

[9]. M. R. Arbabshirani, K. A. Kiehl, et al., Classification of schizophrenia patients based on resting-state functional network connectivity, *Frontiers in Neuroscience*, Vol. 7, 2013, pp. 1-16.

[10]. B. Rehák Bučková, J. Mareš, et al., Multimodal-neuroimaging machine-learning analysis of motor disability in multiple sclerosis, *Brain Imaging and Behavior*, Vol. 17, Issue 1, 2023, pp. 18-34.

[11]. M. E. Raichle, The restless brain: How intrinsic activity organizes brain function, *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 370, Issue 1668, 2015, 20140172.

[12]. W. R. Shirer, H. Jiang, et al., Optimization of rs-fMRI pre-processing for enhanced signal-noise separation, test-retest reliability, and group discrimination, *NeuroImage*, Vol. 117, 2015, pp. 67-79.

[13]. Y. Behzadi, K. Restom, et al., A component based noise correction method (CompCor) for BOLD and perfusion based fMRI, *NeuroImage*, Vol. 37, Issue 1, 2007, pp. 90-101.

[14]. S. Whitfield-Gabrieli, A. Nieto-Castanon, Conn: a functional connectivity toolbox for correlated and anticorrelated brain networks, *Brain Connectivity*, Vol. 2, Issue 3, 2012, pp. 125-141.

[15]. N. Tzourio-Mazoyer, B. Landeau, et al., Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain, *NeuroImage*, Vol. 15, Issue 1, 2002, pp. 273-289.

(058)

Approximate Entropy: An Algorithm for Quantifying Brain Complexity

J. Knociková
University of Chemistry and Technology, Faculty of Chemical Engineering,
Department of Mathematics, Informatics and Cybernetics
E-mail: Juliana.Alexandra.Knocikova@vscht.cz

Summary: Neurophysiological data are characterized by a high degree of nonstationary processes with varying time-frequency features. The data tools resulting from the concept of nonlinear dynamics, such as approximate entropy, proved to be predictive of subsequent clinical changes. This rate of information production describes the randomness or instability of a dynamical system. The statistical parameter of approximate entropy could be used to quantify regularity and predictability and assess the complexity of the analyzed biosignals. On the basis of correctly chosen parameters, the approximate entropy could serve as a reliable quantitative biomarker underlying the appropriate neurodynamics. In this study, we characterize the alterations of this nonlinear tool to characterize changes in the brain dynamics. The level of alterations in EEG signal complexity is a sensitive tool to distinguish pathological brain changes, with a prognostic and diagnostic value for different neuropsychiatric diseases.

Keywords: Approximate entropy, Signal complexity, Neurophysiology, EEG, Quantitative biomarker.

1. Introduction

Complex neurophysiological signals, such as EEG, have been studied using various linear and non-linear tools [1-3]. In fact, brain activity is considered to be a highly complex, non-linear, and mostly irregular system. Adult heathy subjects are characterized by rather higher levels of neurophysiological dynamics. It becomes more ordered during deep sleep, rigid thinking, under deep anesthesia, or during certain neurophysiatric disorders [3, 4]. The less ordered neurodynamic is typical for imaginations, REM sleep, early psychosis, or during psychedelic states [5]. Conventional linear statistics cannot analyze the dynamics of complex physiological signals. The hidden dynamical changes are often undetected by classical time-series EEG analyses. To better understand the implications of the variability present in physiological signals, it is important to use nonlinear tools, in addition to conventional linear approaches [3, 5].

Approximate entropy (ApEn), an algorithm derived from Kolmogorov-Sinai entropy, is a measure of long-term trends in neurophysiological time series, which increases when those long-term trends are disrupted [6]. Increasing this parameter means unpredictability and random behavior (Fig. 1). On the other hand, the presence of repetitive patterns in EEG time series results in more predictable (hence less complex) behavior. Therefore, ApEn alterations could be shown to be a reliable quantitative biomarker and provide useful information on the development of specific neuropsychiatric diseases and related therapeutic progress.

2. The concept of Approximate Entropy

The value of ApEn provides the capability to present quantitative information about the complexity of the EEG signal that exhibits a combination of deterministic and stochastic behaviors. Lower ApEn values are associated with a higher degree of regularity and predictability and reflect a more ordered system or a lower system complexity. High values of ApEn reflect the high disorder and irregularity.

The calculation of ApEn requires the specification of two unknown parameters – m, the embedding dimension, which determines the length of the sequences to be compared, and r, a tolerance threshold to accept similar patterns between two segments. The value of m can be estimated using the first minimum value of the non-linear correlation function called average mutual information and subsequent use of the false nearest neighbor approach. Properly assessed parameter r could work as an additive noise removal filter. The approximate entropy suggested for clinical use demonstrates that r should fall within the interval of 0.1 to 0.25 times the standard deviation of the signal and that m should be 1 or 2 for data length (N) ranging from 100 to 5000 data points.

2.1. Calculation of Approximate Entropy

If there is a data time series *x(n) = x(1), x(2),...,x(N),* where *N* is a number of data samples, the algorithm for calculating the ApEn should start with assessment of the standard deviation of analyzed signal (SD_x), the embedding dimension and the threshold level.

$$SD_x = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}[x(n) - \frac{1}{N}\sum_{n=1}^{N}x(n)]^2} \qquad (1)$$

1. The first step is to estimate vectors X(i) defined

$$X(i) = [x(i), x(i+1), ..., x(i+m-1)], \qquad (2)$$

for i = 1, N-m+1.

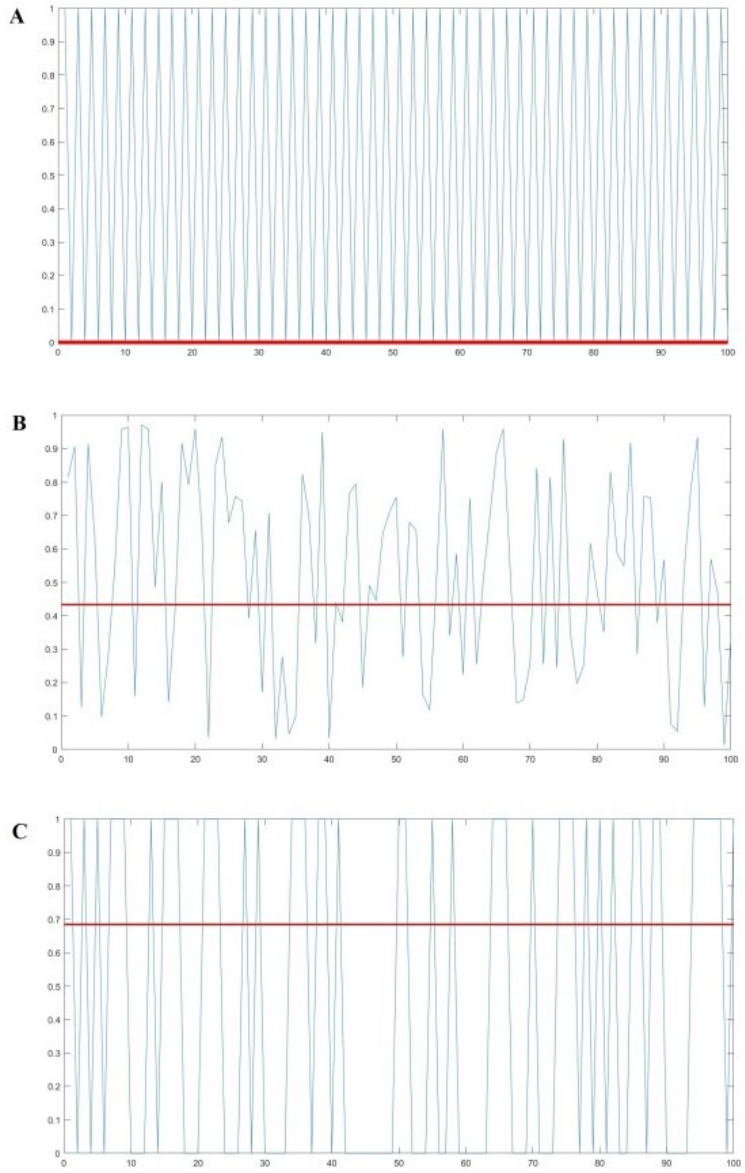


Fig. 1. Regularity and predictability of signals. Low ApEn value (0.00005) of a perfectly regulated signal (A) compared to the increase in ApEn value of the EEG signal (B; 0.4332) and a random signal (C; 0.6849).

2. The difference between X(i) and X(j), d[X(i),X(j)] is estimated as a maximum absolute difference between their related scalar components

$$d[X(i), X(j)] = \\ = \max_{k=0,m-1}[\left|x(i+k) - x(j+k)\right|] \le r \quad (3)$$

3. For a given X(i), the number of differences d[X(i),X(j)], for j = 1, N-m+1 that is smaller or equal than the threshold *r* and the ratio of this number to the total number of m-vectors (N-m+1) must be assessed. If $N_r^m(i)$ is number of

$$\text{d[X(i), X(j)]} \le r, \quad (4)$$

then

$$C_r^m(i) = N_r^m(i)/(N-m+1) \quad (5)$$

This step is repeated for any given i, where i = 1,…,N-m+1.

4. Averaged value of $C_r^m(i)$ natural logarithm is calculated

$$\Phi^m(r) = \sum_{i=1}^{N-m+1} \ln C_r^m(i) / (N-m+1) \quad (6)$$

5. Embedding dimension is increased to m+1 and all previous steps are repeated. The value of $C_r^m(i)$ and $\Phi^{m+1}(r)$ is estimated.

6. According to described algorithm, parameter approximate entropy (ApEn) is theoretically defined as a function of the embedding dimension *m* and a threshold *r*.

$$ApEn(m,r) = \lim_{N\to\infty}[\Phi^m(r) - \Phi^{m+1}(r)] \quad (7)$$

7. Practically, parameter ApEn is expressed as

$$ApEn(m,r,N) = [\Phi^m(r) - \Phi^{m+1}(r)] \quad (8)$$

As stated above, ApEn measures the likelihood that vectors that are close enough within r for m observations remain close within the same tolerance of r when m is increased.

3. ApEn as a Quantitative Biomarker

Due to the advantages of ApEn algorithm, such as the possibility to be applied to EEG signals of shorter length and being almost unaffected by noise, it has been effectively used to quantify the degree of the disorder of analyzed system and has proved to be very sensitive metrics to characterize specific diseases. Moreover, ApEn is highly resistant to short strong transients and outliers.

ApEn detects the appearance of episodic behavior that is often not present in peak amplitudes. Therefore, this algorithm is often considered to reflect the level of new generation of EEG signal patterns. Chaos-time variations in the stability production of healthy biological systems represent the ability of organisms to adapt to the environment and achieve homeostasis.

This variability in the level of complexity of the brain corresponds to the processes alternated with different pathophysiological / physiological stages (Table 1).

Major depressive disorder has been described as a dynamical disease that manifests itself through behavior symptoms, mainly the persistently reduced mood and loss of normal interests. This prevalent disease is characterized by a decrease in the complexity of the brain and higher predictability and regularity of EEGs compared to healthy subjects [7, 8]. This comparison also confirms the concept of right-hemisphere disorganization and the abnormal activity in the prefrontal cortex [7, 9].

Schizophrenia is a persistent and severe psychiatric disorder, manifesting itself as disturbances in cognition, affects, and perceptions. Usually, it is diagnosed by qualitative criteria. The complexity measures underlying this disease depend on different factors, like the disease development, treatment, and the clinical status or symptom severity, mainly the balance between positive and negative qualitative criteria. According to Taghavi et al., 2011, extracted complexity values for normal subjects were significantly higher than that of schizophrenic patients, especially in the limbic area of the brain [10]. Other authors also reported reduced EEG complexity in patients suffering from this disease [11].

However, recent findings reported an increase in ApEn values in schizophrenia patients compared to healthy subjects [12]. Increased neural complexity has been found to be a typical sign in patients suffering from schizophrenia with a more recent onset of the disease, premedicated, and with more positive symptoms [13].

Variations in neurophysiological complexity were also reported for different neurodevelopmental stages. EEG complexity in autistic spectrum disorder showed a lower complexity assessment. For example, children with attention deficit hyperactivity disorder have lower EEG entropy in rest compared to healthy subjects [14, 15]. ApEn of EEG in adolescents with attention deficit / hyperactivity disorder was also reported to manifest lower values than controls during a cognitive task, especially in the right frontal region [16].

Table 1. Alteration of approximate entropy during different neuropsychiatric conditions compared to healthy controls.

Neuro-psychiatric condition	Disease	ApEn	References
Mood and Anxiety	Major depressive disorder	↓	Pezard, et al.,1996 Faust, et al,, 2014
Schizophrenia, Psychosis	Schizophrenia (based on clinical status or symptom severity)	↑↓	Taghavi, et al., 2011 Akar, et al. 2016 Leei, et al., 2008 Thilakvathi, et al., 2017
Neurodevel. disease	Attention deficit hyperactivity disorder	↓	Chen, et al., 2019 Khoshnoud, et al., 2018 Sohn, et al., 2010

4. Conclusions

In the clinical context, early diagnosis and appropriate treatment are crucial to prevent disease progression. The measure of ApEn could serve as an effective quantitative biomarker with prognostic and diagnostic value to monitor the impact of pharmacological and rehabilitation treatments.

References

[1]. M. Akay, Hypoxia silences the neural activities in the early phase of the phrenic neurogram of eupnoea in the piglet, *Journal of NeuroEngineering and Rehabilitation*, Vol. 2, 2005, 32.

[2]. J. A. Knocikova, Wavelet analysis of electrical activities from respiratory muscles during coughing and sneezing in anaesthetized rabbits, *Acta Veterinaria Brunensis,* Vol. 78, Issue 3, 2009, pp. 387-397.

[3]. J. A. Knocikova, Quantitative electroencephalographic biomarkers behind major depressive disorder, *Biomedical Signal Processing and Control,* Vol. 68, 2021, 102596.

[4]. O. Faust, et al., Depression diagnosis support system based on EEG signal entropies. *J. Mech. Med. Biol*., Vol. 14, Issue 3, 2014, pp. 1-20.

[5]. R. L. Carhart-Harris, et al., The entropic brain: a theory of conscious states informed by neuroimaging research with psychedelic drugs, *Front. Hum. Neurosci.,* Vol. 8, Issue 20, 2014.

[6]. S. M. Pincus, Approximate entropy as a measure of system complexity, *Proc. Natl. Acad. Sci. U.S.A.,* Vol. 88, 1991, pp. 2297-2301.

[7]. J. Pezard, et al., Depression as a dynamical disease, *Biol. Psychiatry*, Vol. 39, Issue 12, 1996, pp. 991-999.

[8]. O. Faust, et al., Depression diagnosis support system based on EEG signal entropies, *J. Mech. Med. Biol*., Vol. 14, Issue 3, 2014, pp. 1-20.

[9]. T. Glenn, et al., Approximate entropy of self-reported mood prior to episodes in bipolar disorder, *Bipolar Disord*., Vol. 8, 2006, pp. 424-429.

[10]. M. Taghavi, et al., Usefulness of approximate entropy in diagnosis of schizophrenia, *Iran J. Psychiatry Behav. Sci.,* Vol. 5, Issue 3, 2011, pp. 62-70.

[11]. Akar, et al., Analysis of the complexity measures in the EEG of schizophrenia patients, *Intern. J. of Neural Systems,* Vol. 26, Issue 3, 2016, 1650008.

[12]. B. Thilakvathi, et al., EEG signal complexity analysis for schizophrenia during rest and mental activity, *Biomedical Research-India*, Vol. 28, Issue 1, 2017, pp. 1-9.

[13]. S. H. Lee, Nonlinear analysis of electroencephalogram in schizophrenia patients with persistent auditory hallucination, *Psychiatry Investigation*, Vol. 5, Issue 2, 2008, 115.

[14]. H. Chen, et al., EEG characteristics of children with attention-deficit/hyperactivity disorder, *Neuroscience*, Vol. 406, 2019, pp. 444-456.

[15]. S. Khoshnoud, et al., Functional brain dynamic analysis of ADHD and control children using nonlinear dynamical features of EEG signals, *Journal of Integr. Neuro*., Vol. 17, Issue 1, 2018, pp. 17-30.

[16]. H. Sohn, et al., Linear and non-linear EEG analysis of adolescents with attention-deficit/hyperactivity disorder during a cognitive task, *Clin. Neurophysiol.,* Vol. 121, Issue 11, 2010, pp. 1863-1870.

(059)

Detector with an RGB Sensor for Determining the Technical Condition of Motor Oil of Locomotive Diesel Engines

Denys Baranovskyi and Maryna Bulakh
Rzeszow University of Technology, Faculty of Mechanics and Technology, Kwiatkowskiego 4,
37-450 Stalowa Wola, Poland
Tel.: + 48736073748
E-mail: d.baranovsky@prz.edu.pl

Summary: This study introduces a novel approach leveraging an RGB sensor-based detector to monitor the technical condition of motor oil in real-time. The detector, designed to analyze the red, green, and blue spectral ranges, offers a cost-effective and simplified alternative for detecting wear particles in the motor oil. Utilizing Wavelet Transform for data analysis, the study demonstrates the detector's capability to reduce data complexity and enhance the precision of wear particle detection. The processed RGB data feeds into a Neural Network and Fuzzy Logic system, further analyzing the concentration of wear particles and providing a qualitative assessment of the oil's condition. Our findings indicate that the RGB sensor-based approach is not only viable but also advantageous in terms of cost, efficiency, and potential integration into onboard diagnostic systems for continuous diesel engine monitoring. The study's implications for predictive maintenance strategies could significantly impact the locomotive industry by improving the reliability and longevity of locomotive diesel engines.

Keywords: Detector, RGB sensor, Wavelet transform, Spectral ranges, Motor oil, Technical condition, Locomotive diesel engines.

1. Introduction

The relentless pursuit of operational efficiency and sustainability within the locomotive industry underscores the need for innovative diagnostic tools that can accurately assess the technical condition of locomotive diesel engines [1]. A pivotal component of engine maintenance is monitoring the condition of motor oil, as it plays a crucial role in the engine's performance and longevity [2]. Traditional methods for analyzing motor oil [3, 4], such as spectrographs and photometric instruments, while effective, often require stationary conditions, additional equipment, reagents, and time, thus limiting their applicability in real-time, on-board scenarios [5, 6].

This research introduces a novel approach leveraging a detector with an RGB sensor to determine the technical condition of motor oil in locomotive diesel engines. By utilizing red, green, and blue spectral ranges data, this method aims to provide a cost-effective, efficient, and accurate means of assessing wear particles in motor oil, ultimately enhancing engine maintenance strategies and operational reliability.

2. Methods

It is known that every metal has a reflectance in the RGB spectrum [3, 5]. Each metal has a unique absorption peak within the RGB spectrum, influencing the oil's overall RGB reflectance differently.

Fig. 1 illustrates how various metals (Al, Na, Fe, Mg, Ni, Cr, Cu, Si, Mn, Pb) potentially influence the reflectance of red, green, and blue spectral ranges in motor oil.

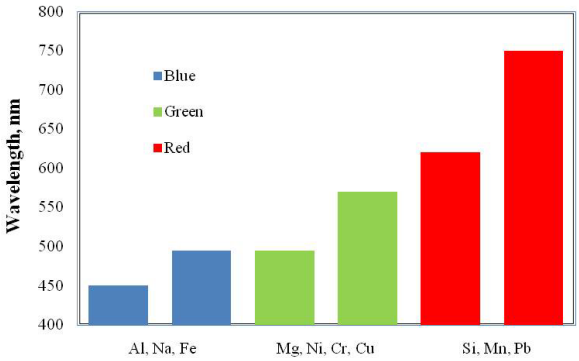


Fig. 1. Reflectivity of metals in the RGB spectrum.

Fig. 1 provides a visualization of the hypothetical impact of each metal on motor oil performance, which can be determined using RGB values. This graph can be used for diagnostic purposes in detecting and identifying engine wear particles through motor oil analysis.

In connection with the above, it is proposed to use a detector with an RGB sensor. The developed detector with an RGB sensor is designed to obtain RGB data for the subsequent determination of wear particles in motor oil of locomotive diesel engines. The detector consists of an RGB sensor capable of capturing red, green, and blue spectral ranges of light reflected from or transmitted through a motor oil sample. The design focuses on achieving high sensitivity and accuracy in differentiating subtle changes in oil coloration, attributed to varying concentrations of wear particles. A controlled light source illuminates the oil sample, ensuring consistent lighting conditions across all measurements.

The RGB sensor collects spectral data from the illuminated motor oil samples. The data acquisition system is calibrated to account for any systemic variations, ensuring the reliability of the spectral data obtained.

Data from the RGB sensor enters the Neural Network. The output of the Neural Network produces quantitative values of the concentrations of wear particles in the motor oil of locomotive diesel engines. The resulting values are fed to the Fuzzy logic block. At the Fuzzy logic stage, we obtain a conclusion about the technical condition of the motor oil of locomotive diesel engines. This finding will indicate the likelihood of continued use of the motor oil or the need to change the motor oil of locomotive diesel engines.

RGB data received from an RGB sensor in the first version is amenable to Wavelet Transform, with the help of which the data is decomposed into a set of wavelets. The RGB data undergoes Wavelet Transform to reduce data complexity and highlight significant features indicative of wear particle concentrations. This transformation facilitates a more efficient analysis by focusing on critical data elements. In the second option, data from the RGB sensor is transmitted directly, i.e. without Wavelet Transform.

This methodological approach is designed to monitor the condition of motor oil of locomotive diesel engines in real time to improve maintenance efficiency and extend engine durability.

3. Results

The operating principle of the proposed system for determining the technical condition of motor oil in locomotive diesel engines is as follows.

First, testing is carried out on fresh motor oil. Data from the RGB sensor is incoming, which is represented by input matrices $A_{in}^{r}, A_{in}^{g}, A_{in}^{b}$ of size $m \times k$. On the basis of which a conclusion will be drawn about the technical condition of the motor oil by the proposed system.

During engine operation, the motor oil of locomotive diesel engines undergoes degradation changes, and wear particles also enter the motor oil of locomotive diesel engines. RGB sensor measurements are taken every 4-8 hours of engine operation. Accordingly, data will be received from the RGB sensor in the form of matrices $A_{n}^{r}, A_{n}^{g}, A_{n}^{b}$ of size $m \times k$, where n is the measurement step, $n = 1,2,3, \ldots$

Next, the difference between the input and new matrix is calculated using the equations:

$$\Delta_n^r = A_{in}^r - A_n^r;\ \Delta_n^g = \\ = A_{in}^g - A_n^g;\ \Delta_n^b = A_{in}^b - A_n^b \quad (1)$$

The result of calculating the difference in the distributions of RGB signals for motor oil of locomotive diesel engines after operating 214 engine hours is presented in Fig. 2.

This graph (Fig. 2) illustrates the distinct differences in the blue spectral range, indicating a high concentration of wear particles (Al, Na, Fe) in the motor oil of locomotive diesel engines after operating 214 engine hours. The red and green spectral ranges show relatively minor changes, suggesting that these colours are less indicative of the technical condition of the motor oil of locomotive diesel engines after operating 214 engine hours.

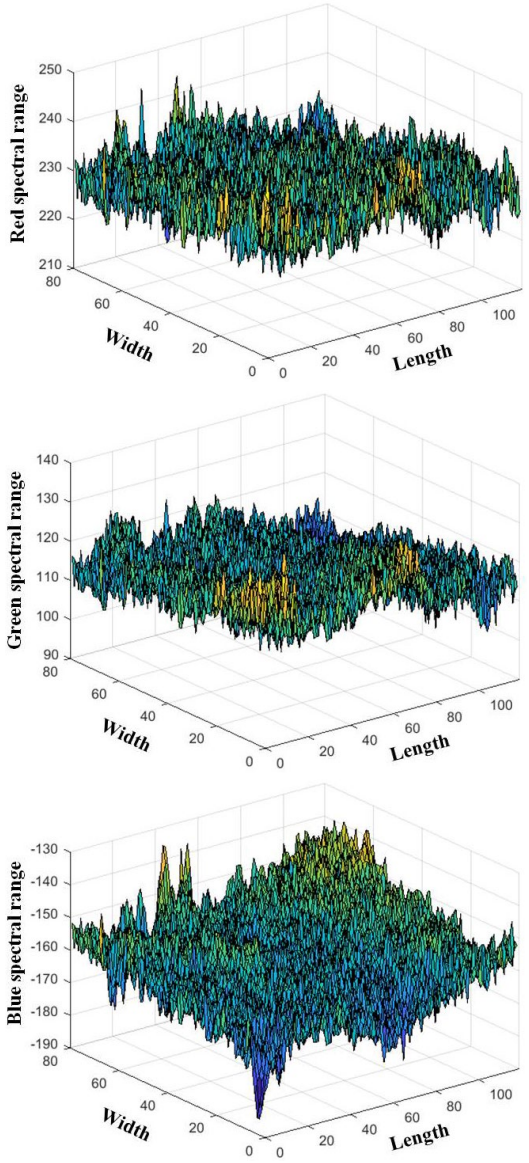


Fig. 2. Difference in RGB signal distributions of the motor oil of locomotive diesel engines after operating for 214 engine hours.

Also, a Wavelet Transform is performed for data from the RGB sensor. As a result of the Wavelet Transform, sets of wavelets are obtained for the red, green, and blue ranges. A set of wavelets is represented by input matrices $AA_{in}^{r}, AA_{in}^{g}, AA_{in}^{b}$ of size $p \times r$. Graphical representation of the Wavelets transformation for RGB spectral ranges in fresh motor oil of locomotive diesel engines is shown in Fig. 3.

Fig. 3 displays the initial wavelet transformation results for the red, green, and blue spectral ranges,

highlighting the baseline characteristics of fresh motor oil of locomotive diesel engines. The transformation enables the identification of specific patterns and features within each spectral range, which serve as reference points for subsequent comparisons with oil samples taken after various engine operation periods.

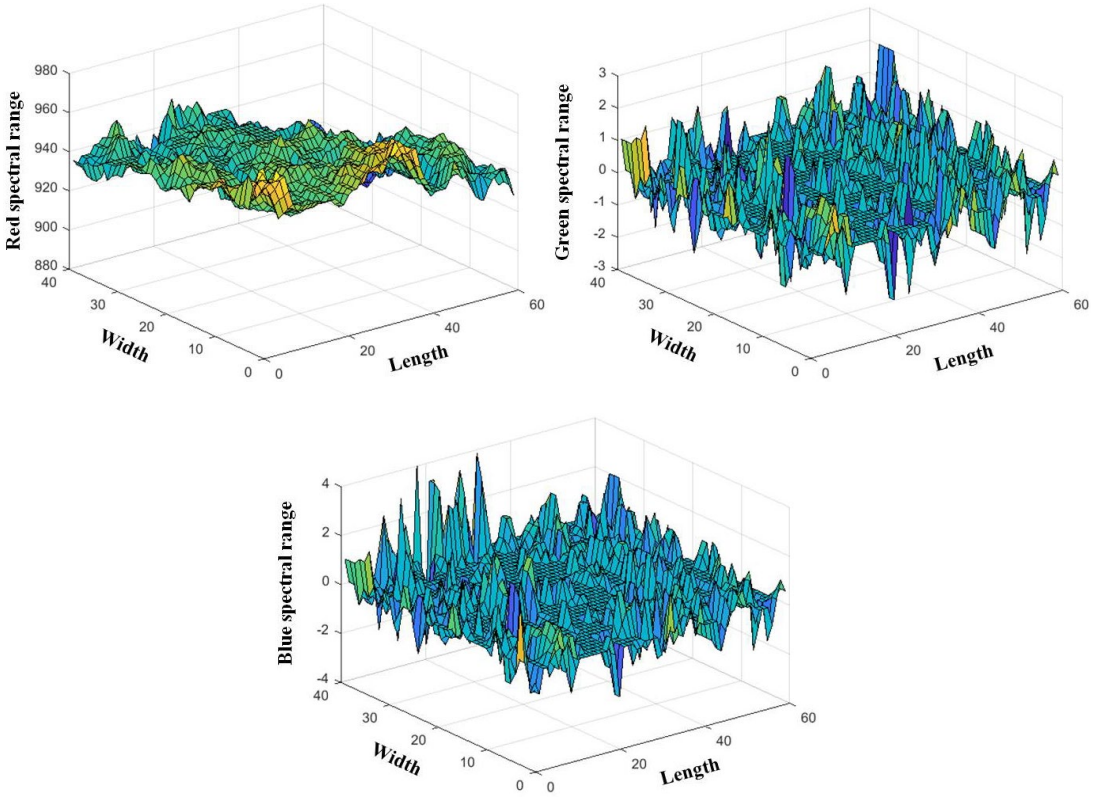


Fig. 3. Graphical representation of the Wavelets transformation for RGB spectral ranges in fresh motor oil of locomotive diesel engines.

During engine operation, the motor oil undergoes degradation changes, and wear particles also enter the motor oil of locomotive diesel engines. RGB sensor measurements are taken every 4-8 hours of locomotive diesel engines operation. The result is the following set of wavelets for the red, green, and blue spectral ranges. The set of wavelets for each new dimension is represented by matrices AA_n^r, AA_n^g, AA_n^b, size $p \times r$. Next, the difference $\Delta_{an}^r, \Delta_{an}^g, \Delta_{an}^b$ of the input and new matrices is calculated similarly to equations (1).

The result of calculating the difference of the Wavelet Transform for RGB spectral ranges of the motor oil of locomotive diesel engines after operating 214 engine hours is presented in Fig. 4.

Fig. 4 illustrates the significant changes in the Wavelet Transform outputs across the red, green, and blue spectral ranges, compared to the baseline established with fresh motor oil of locomotive diesel. The variations captured here underscore the presence of wear particles and the degradation of oil quality over time. The graphical representation effectively demonstrates the increased sensitivity and specificity of the Wavelet Transform in detecting subtle changes in the motor oil's condition.

The data in Fig. 4 (green and blue spectral ranges) indicate an increased concentration of wear particles in the motor oil of locomotive diesel engines after operating 214 engine hours of such metals as: Al, Na, Fe, Mg, Ni, Cr, and Cu. Red spectral range also indicates the presence of Si, Mn, Pb. The concentrations of these metals in motor oil can be determined through laboratory research.

The resulting differences between the data sets Δ_n^r, $\Delta_n^g, \Delta_n^b, \Delta_{an}^r, \Delta_{an}^g, \Delta_{an}^b$ are fed to the input of the Neural Network.

Every Neural Network has a training layer. To train the Neural Network, spectral studies of motor oil of locomotive diesel engines are carried out in laboratory conditions. Motor oil samples are collected from locomotive diesel engines at various stages of operation, ranging from fresh oil to oil that has been in use for extended periods. These samples represent a spectrum of technical conditions, from optimal to significantly degraded.

The system's accuracy and reliability will be validated through comparative analysis with traditional oil analysis methods. This involves cross-referencing the detector's assessments with results obtained from spectrographic and photometric analyses conducted under laboratory conditions.

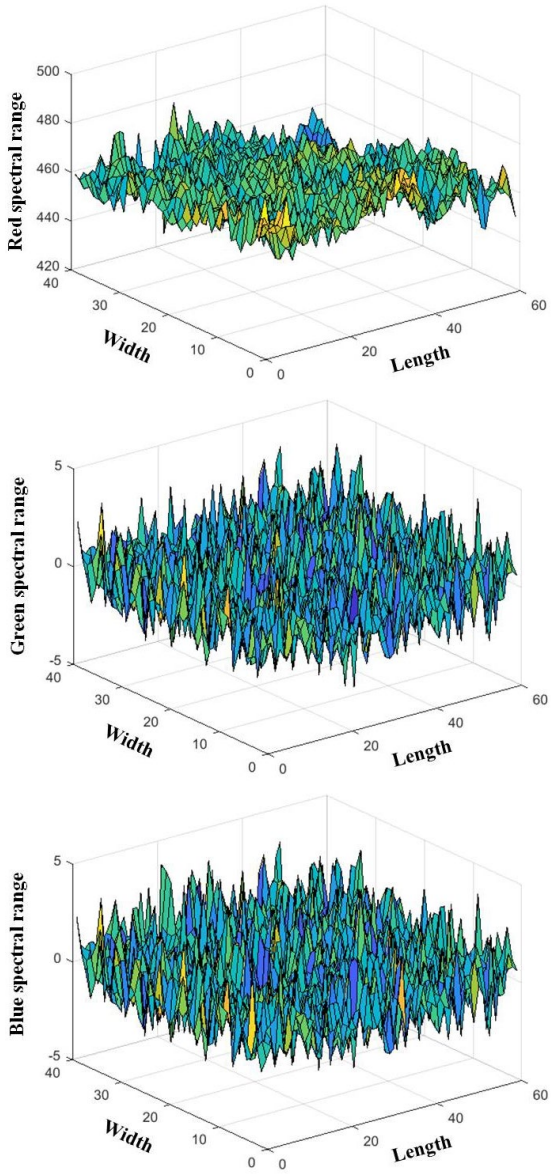


Fig. 4. Difference in Wavelet Transform outputs for RGB spectral ranges of the motor oil of locomotive diesel engines after operating 214 engine hours.

At the corresponding engine operating time, the quantitative values of the concentrations of wear particles in the motor oil of locomotive diesel engines are compared and assigned to the data sets Δ_n^r, Δ_n^g, Δ_n^b, Δ_{an}^r, Δ_{an}^g, Δ_{an}^b. As a result of training the Neural Network, at its output we obtain a quantitative result of the concentrations of wear particles in the motor oil of locomotive diesel engines for a certain time of engine operation.

As a result of the work of Neural Networks, the output contains two data arrays of wear product concentrations in motor oil of locomotive diesel engines for a certain engine operating time. Next, a comparison of these two arrays is performed. If the difference in these data does not exceed 15 %, then a conclusion will be made about the advisability of using only that part of the system in which the Wavelet transform is used. This will reduce the computational energy at each stage of the system's operation to determine the technical condition of the motor oil of locomotive diesel engines.

The proposed Neural Network architecture is as follows:

- *Input Layer*. Input Size: 3 neurons, corresponding to the RGB data obtained from the sensor. Each neuron represents the intensity of red, green, and blue spectral ranges from or transmitted through the motor oil sample.

- *Hidden Layers*. To capture the complex relationships between RGB values and wear product concentrations, the network should have multiple hidden layers: *Layer 1*: 64 neurons, using ReLU activation for non-linear processing; *Layer 2*: 32 neurons, also with ReLU activation. This layer further processes the information to help identify patterns specific to different wear particles. Optionally, additional layers or adjustments to the number of neurons could be explored based on the complexity of the data and the required accuracy.

- *Output Layer*. Output Size: Depending on the approach, this could be a single neuron if the goal is to predict a single concentration value of wear particles or multiple neurons if predicting concentrations of multiple types of wear particles separately.

Activation Function: for classification (e.g., low, medium, high concentration levels), a SoftMax activation function might be more appropriate.

Training the Neural Network. Dataset: A collection of RGB data from motor oil samples with known concentrations of wear particles. This dataset is used to train and validate the model. Preprocessing: Normalize the RGB values to a 0-1 range to facilitate training. If predicting specific concentrations, ensure the target values are scaled appropriately.

Loss Function and Optimizer. For classification, categorical cross entropy might be more suitable. An optimizer like SGD (Stochastic Gradient Descent) can be used to minimize the loss function during training.

Training Process. Divide the dataset into training, validation, and test sets. Train the model using the training set, while monitoring its performance on the validation set to prevent overfitting. Adjust the model's architecture, number of epochs, and learning rate based on performance.

The next block is Fuzzy logic. At the Fuzzy logic stage, a conclusion will be obtained about the technical condition of the motor oil of locomotive diesel engines. This output will verbally indicate whether the motor oil can be continued to be used or whether the motor oil of locomotive diesel engines needs to be replaced.

4. Conclusions

The research presented an approach to determining the technical condition of motor oil of locomotive diesel engines through the use of a detector equipped with an RGB sensor. This method offers a significant advancement over traditional motor oil analysis

techniques by providing a real-time, cost-effective solution that can be integrated into on-board systems. The utilization of RGB spectral data, analyzed through Wavelet Transform, demonstrates a high potential for accurately identifying wear particles in motor oil of locomotive diesel engines. Utilizing Wavelet Transform on the RGB data facilitated a reduction in data volume and computational energy requirements. This process enabled the efficient handling of the spectral data, enhancing the system's performance in real-time applications.

The findings underscore the efficacy of the RGB sensor in capturing critical data that reflects the motor oil's condition, thereby enabling timely decisions regarding engine maintenance. Furthermore, the comparative analysis between direct RGB data and Wavelet Transformed data provides valuable insights into optimizing the data processing for enhanced efficiency and reduced computational load.

The study established a strong correlation between the blue spectral range data and the concentration of wear particles in the oil after 214 engine hours, indicating significant degradation. In contrast, the red and green spectral ranges did not show notable changes, highlighting the importance of the blue spectrum in assessing motor oil condition.

The work also proposes Neural Network architecture for determining the technical condition of motor oil of locomotive diesel engines. A comparison of the operation of Neural Networks, as well as the operation of the Fuzzy logic block in the proposed system for determining the technical condition of motor oil of locomotive diesel engines, will be given in future publications and studies.

Future research will delve deeper into refining the system's accuracy and exploring the integration of more advanced machine learning algorithms. This work lays a foundational step towards diesel engine maintenance protocols, with the potential to significantly impact the locomotive industry's approach to diesel engine condition monitoring and maintenance practices.

References

[1]. L. P. Lingaitis, S. Mjamlin, D. Baranovsky, V. Jastremskas, Experimental investigations on operational reliability of diesel locomotives engines, *Eksploatacja i Niezawodnosc – Maintenance and Reliability,* Vol. 14, Issue 1, 2012, pp. 6-11.

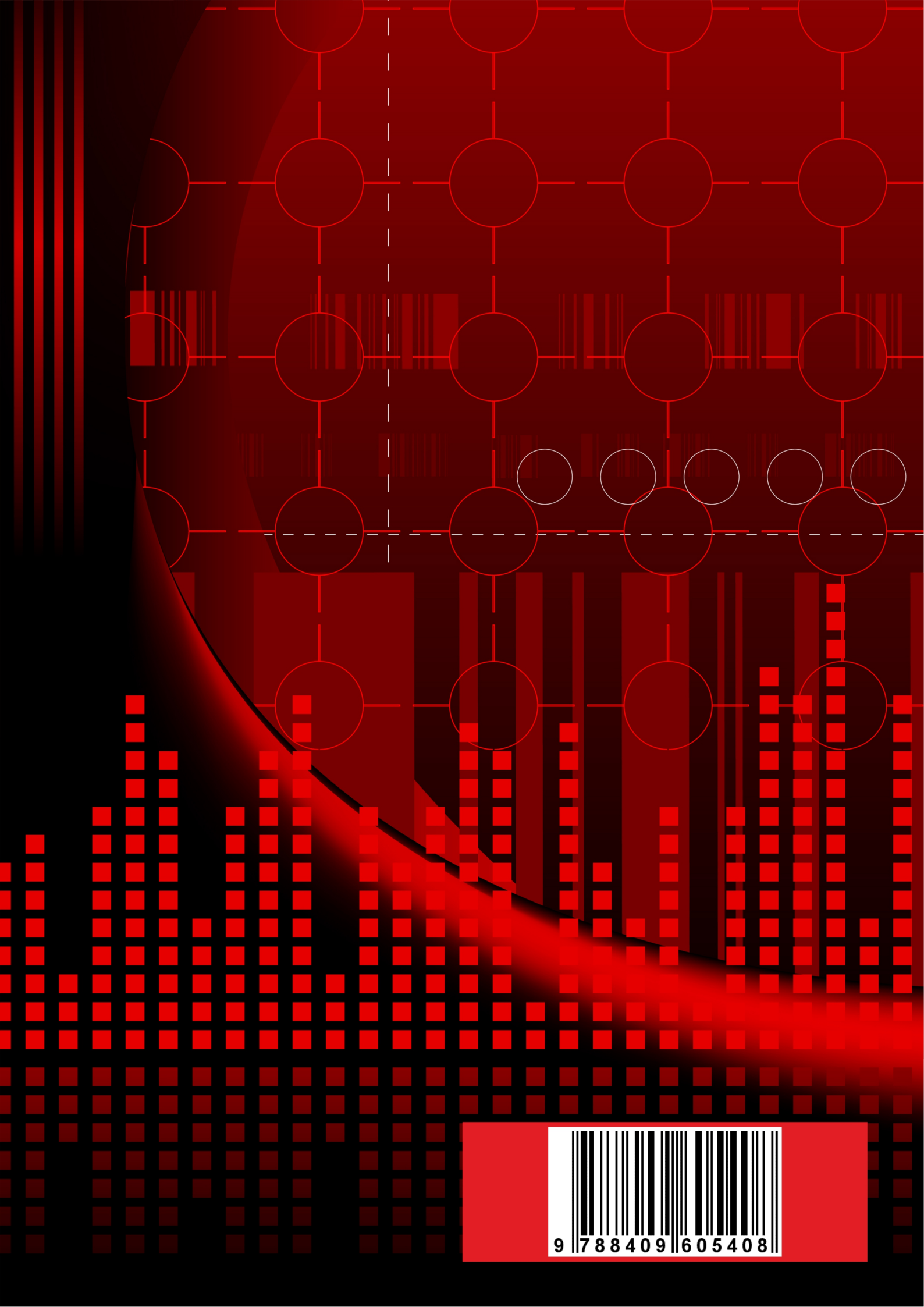
[2]. L. P. Lingaitis, S. Mjamlin, D. Baranovsky, V. Jastremskas, Prediction methodology of durability of locomotives diesel engines, *Eksploatacja i Niezawodnosc – Maintenance and Reliability*, Vol. 14, Issue 2, 2012, pp. 154-159.

[3]. C. V. Ossia, H. Kong, L. V. Markova, N. K. Myshkin, On the use of intrinsic fluorescence emission ratio in the characterization of hydraulic oil degradation, *Tribol. Int.*, Vol. 41, 2008, pp. 103-110.

[4]. C. Tao, H. Zhu, X. Wang, S. Zheng, Q. Xie, C. Wang, R. Wu, Z. Zheng, Compressive single-pixel hyperspectral imaging using RGB sensors, *Optics Express*, Vol. 29, Issue 7, 2021, pp. 11207-11220.

[5]. M. Bulakh, L. Klich, O. Baranovska, A. Baida, S. Myamlin, Reducing traction energy consumption with a decrease in the weight of an all-metal gondola car, *Energies*, Vol. 16, Issue 18, 2023, 6733.

[6]. M. Benavides, J. Mailier, A. Hantson, G. Muñoz, A. Vargas, J. Van Impe, A. V. Wouwer, Design and test of a low-cost RGB sensor for online measurement of microalgae concentration within a photo-bioreactor, *Sensors*, Vol. 15, Issue 3, 2015, pp. 4766-4780.


9 788409 605408