A comparison of different support vector machine kernels for artificial speech detection

Choon Beng Tan[1], Mohd Hanafi Ahmad Hijazi[1], Puteri Nor Ellyza Nohuddin[2]
[1]Data Technology and Applications Research Group, Faculty of Computing and Informatics, Universiti Malaysia Sabah, Kota Kinabalu, Malaysia
[2]Institute of IR4.0, Universiti Kebangsaan Malaysia, Bangi, Malaysia

Article Info

Article history:

Received Aug 31, 2021
Revised Jul 28, 2022
Accepted Aug 15, 2022

Keywords:

Artificial speech
Artificial speech detection
Support vector machine

ABSTRACT

As the emergence of the voice biometric provides enhanced security and convenience, voice biometric-based applications such as speaker verification were gradually replacing the authentication techniques that were less secure. However, the automatic speaker verification (ASV) systems were exposed to spoofing attacks, especially artificial speech attacks that can be generated with a large amount in a short period of time using state-of-the-art speech synthesis and voice conversion algorithms. Despite the extensively used support vector machine (SVM) in recent works, there were none of the studies shown to investigate the performance of different SVM settings against artificial speech detection. In this paper, the performance of different SVM settings in artificial speech detection will be investigated. The objective is to identify the appropriate SVM kernels for artificial speech detection. An experiment was conducted to find the appropriate combination of the proposed features and SVM kernels. Experimental results showed that the polynomial kernel was able to detect artificial speech effectively, with an equal error rate (EER) of 1.42% when applied to the presented handcrafted features.

Corresponding Author:

Mohd Hanafi Ahmad Hijazi
Data Technology and Applications Research Group, Faculty of Computing and Informatics
Universiti Malaysia Sabah, Kota Kinabalu, Malaysia
Email: hanafi@ums.edu.my

1. INTRODUCTION

Speaker recognition is the process of identification or verification of a speaker from the speech signal. Speaker identification is the process of determining the speech owner from the speech, whereas speaker verification is the process of accepting or rejecting the claimed identity of a speaker. Recently, automatic speaker verification (ASV) systems were introduced to provide better security, replacing the traditional authentication methods that were less efficient and secure. Applications of ASV systems include but are not limited to access control and banking transactions [1].

In spite of the security and comfort brought by ASV systems, spoofing attacks from security foes is unavoidable. To bypass the ASV systems, malicious entities attempted to launch a spoofing attack to get access to the system illegally. Various countermeasures named voice presentation attack detection (PAD) were introduced to secure the ASV systems. Generally, voice PAD can be categorized into replayed and artificial speech detection. Artificial speech refers to the speech signal generated by speech synthesis and voice conversion techniques, whereas replayed speech refers to the speech signal generated by replaying the recorded speech.

To secure the ASV systems, numerous voice PADs were introduced to secure the ASV systems. There were many classifiers used in recent works to detect artificial speech. One of the extensively used classifiers in recent works was the support vector machine (SVM), as it was found to excel in various classification tasks [2], [3]. From the findings, the recent work [4] showed that SVM with radial basis function kernel (RBF) outperformed classifiers such as k-nearest neighbour (KNN), decision tree, and Naive Bayes with 1% equal error rate (ERR) on the ASVspoof 2019 replay evaluation set. The performances of different SVM kernels were experimented and found that the RBF kernel performed the best in replay detection.

Nonetheless, to the best of our knowledge, there were none of the studies shown to investigate the performance of different kernels of SVM against artificial speech detection. It is necessary to investigate the appropriate kernel used in SVM as the performance of the model varies depending on the classification tasks. The selection of kernel is dependent on the features input as some features are linear separable by the SVM hyperplane, and some are not. Hence, in this work, the performance of different kernels of SVM on artificial speech detection is presented. On the other hand, handcrafted features such as hexadecimal-based features, image-based features, and the conventional mel-frequency cepstral coefficient (MFCC) features are used for artificial speech detection.

The key contribution of this paper is the empirical comparison performance of SVM kernels in detecting artificial speech when applied to the presented handcrafted features. The remaining sections of the paper are arranged as: section 2 describes the proposed features and classifiers for artificial speech detection; section 3 presents the experimental setup, results, and discussion; and lastly, section 4 concludes the paper.

2. METHOD

2.1. MFCC

MFCC coefficients, which are typically optimal for speech analysis, were also used as features for the work described in this paper. The process of extracting MFCC is presented in Figure 1. First, the input signal was windowed into short frames. Then, discrete fourier transform (DFT) was applied to the signal (waveform) to obtain the power spectrum. Logarithm was applied to the amplitude to obtain the log-amplitude spectrum. Then, mel-scaling was conducted in which mel filterbank was applied to the log-amplitude spectrum to produce the mel spectrum. Lastly, discrete cosine transform (DCT) was applied on the mel spectrum to produce a number of coefficients known as MFCC. It is shown that the first 13 coefficients of MFCC were most informative about formants and spectral envelope [5]. Hence, 13 MFCC coefficients were used in this paper as conventional speech features.

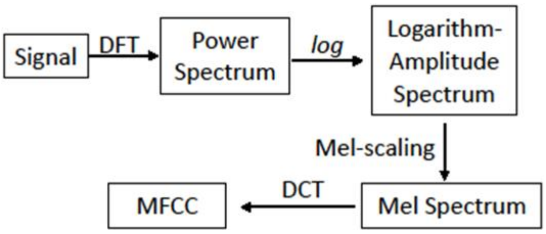


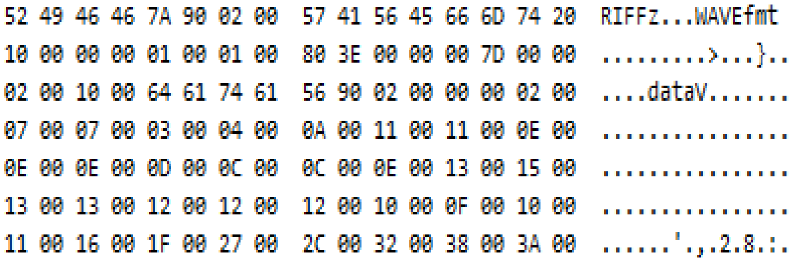
Figure 1. The process of MFCC feature extraction

2.2. Hexadecimal frequencies

Figure 2 is presented to show the hexadecimal representation of voice data. Similar to image representation, voice data can also be represented in text and numeric formats such as binary and hexadecimal. Hexadecimal representation provides a more human-friendly representation in numeric compared to binary. To the best of our knowledge, there was no related work done by applying features engineered from a hexadecimal representation of speech signal for spoof detection. In this paper, text-based features were extracted from the hexadecimal representation of audio data to form a feature space.

A work that utilizes features extracted from hexadecimal represented data for classification problems were found in [6], [7]. In the works [6], [7] the occurrences of each opcode in the executable file were counted and used as features to classify malicious software (malware). The approach used by [6], [7] produced high accuracy in malware classification. The approach was able to achieve good performance because the different classes of malware usually have a higher frequency of certain opcodes.

As the approach [6], [7] produced good result in malware classification, it was adapted to the domain of artificial speech detection in this paper. In this paper, a hexadecimal representation of speech was used to extract features to classify between genuine and spoof speech. The artificial speech data may contain an abnormal number of certain hexadecimal, ranged from 00 to FF, which may be used as an indicator to distinguish between genuine and spoof voices.



```
52 49 46 46 7A 90 02 00  57 41 56 45 66 6D 74 20  RIFFz...WAVEfmt 
10 00 00 00 01 00 01 00  80 3E 00 00 00 7D 00 00  .........>...}..
02 00 10 00 64 61 74 61  56 90 02 00 00 00 02 00  ....dataV.......
07 00 07 00 03 00 04 00  0A 00 11 00 11 00 0E 00  ................
0E 00 0E 00 0D 00 0C 00  0C 00 0E 00 13 00 15 00  ................
13 00 13 00 12 00 12 00  12 00 10 00 0F 00 10 00  ................
11 00 16 00 1F 00 27 00  2C 00 32 00 38 00 3A 00  ......'.,.2.8.:.
```

Figure 2. The hexadecimal representation of a voice recording

To the best of our knowledge, this is the first work that used hexadecimal-based features in detecting artificial speech. For each of the text-represented speech data, the occurrences of each of the 256 hexadecimal, from 00 to FF, were counted. Then, a histogram of hexadecimal frequencies consisting of 256 feature sets was computed. In addition, min-max normalized hexadecimal frequencies were derived from the hexadecimal frequencies by applying min-max normalization [8] on the hexadecimal frequencies. In total, 512 features were extracted from hexadecimal. The (1) shows the formula for min-max normalization used in this paper.

$$Normalization\ min - max = (xi - xmin)\ /\ (xmax - xmin) \tag{1}$$

Where x_i is the occurrence of hexadecimal value i, x_{max} and x_{min} are the maximum and the minimum number of occurrences of hexadecimal values in a speech, respectively.

2.3. Image-based features

The images used in this works were the spectrogram and MFCC for artificial speech detection. Although both spectrogram and MFCC are commonly used to represent speech signals, little interest has been paid to applying both as images [9], [10]. In this paper, the spectrogram and MFCC images were generated from the audio using pyplot and librosa libraries in python, respectively and saved as a 640×480 pixels PNG image. The examples of the generated spectrogram and MFCC images are Figure 3 and Figure 4, respectively.

Two types of image-based features were extracted from both of the spectrogram and MFCC images in this paper, namely color layout filter (CLF) and local binary patterns (LBP) features. Weka's implementation of the CLF features was used in this paper, resulted in 33 CLF features [11]. This paper applied the setting used in the original LBP [12], with a neighborhood radius r = 1, resulting in 8 neighboring pixels in a 3×3 pixels window. Then, the generated frequency histogram of the LBP operation was used to generate the 256 LBP features.

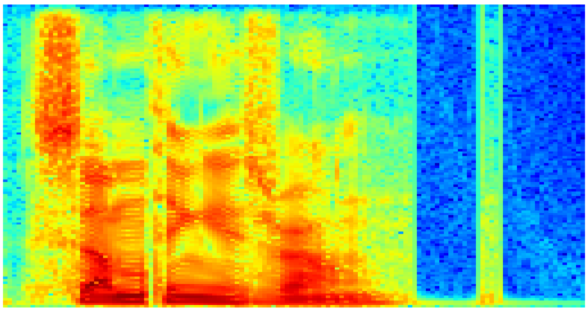


Figure 3. An example of a spectrogram image generated from speech recording for artificial speech detection

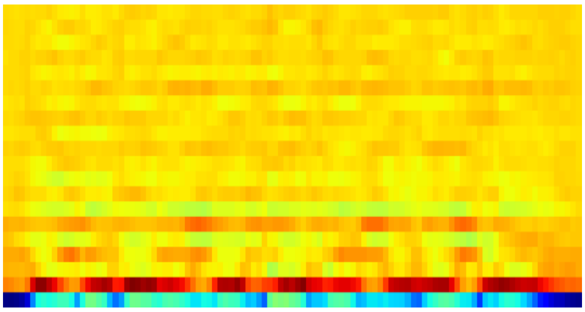


Figure 4. An example of an MFCC image generated from speech recording for artificial speech detection

2.4. Support vector machine (SVM)

SVM is one of the supervised machine learning models which mostly used in binary classification tasks [13], [14]. There were also several recent works introduced which used SVM, for example [15], [16]. In this paper, various SVM settings were tested to identify the appropriate settings for artificial speech detection. The Weka implementation of SVM, known as libsvm library, was used in this paper. Four SVM kernels were tested, namely radial basis function, linear, polynomial, and sigmoid.

The RBF kernel is usually the default kernel used in most of the machine learning tools and libraries such as Weka and sklearn. The RBF is a real-valued function often used to build function estimates. The of linear kernel is (2).

$$k\gamma\ (x,y)\ =\ exp\ (\ -\gamma\ ||\ x\ -\ y\ ||2\ ) \tag{2}$$

Where γ parameter defines the influence of a training sample selected as support vector while $||\ x\ -\ y\ ||$ is the euclidean distance between two points x and y. As for linear kernel, it is used for linearly separable data. Linear separable means that the data can be separated by a straight line if the data is graphed into two dimensions. The (3) shows the formula of the linear kernel.

$$k(x,y)\ =\ x \cdot y \tag{3}$$

Where $x,\ y$ is the dot product of two points x and y. Polynomial kernel portrays the resemblance of feature vectors in feature space over the original variables polynomials to allow the non-linearity of the model. The polynomial kernel is often used in image processing tasks. The (4) shows the formula of the polynomial kernel.

$$k(x,y)\ =\ (\ x \cdot y\ +\ 1\ )q \tag{4}$$

Where $x,\ y$ is the dot product of two points x and y while q is the degree of the polynomial. The sigmoid kernel is equivalent to a two-layer perceptron model and is often used in a neural network as an activation function. The (5) shows the formula of the sigmoid kernel.

$$k(x,y)\ =\ tanh\ (\alpha\ xT\ y\ +\ r\ ) \tag{5}$$

Where α is the scaling parameter of the sample while r is the shifting parameter for threshold mapping of the transpose T of the two points x and y. More information on the kernels can be found in [17]–[19].

3. RESULT AND DISCUSSION

3.1. Experimental setup

An experiment was conducted to identify the appropriate settings for SVM in artificial speech detection. In this paper, the ASVspoof 2019 logical access (LA) dataset was used for the experiment, which was made up of speech synthesis and voice conversion attacks [20], [21]. The ASVspoof 2019 LA dataset consists of three partitions, namely training, development, and evaluation sets. The corpus was built from speech samples of 107 speakers, of which 46 were male and 61 females. There were six spoof algorithms, namely A01-A06, in the training and development partition.

As the training and partition consist of 5,128 bona fide utterances and 45,096 utterances, resampling was conducted as a preventive measure in this paper to reduce the chances of model overfit during training. Both under-sampling and over-sampling were used to ensure both bonafide and spoof samples were in the same number. Both bonafide and spoof samples were resampled to 7,200 samples, a total of up to 14,400 samples used in the experiments.

The resampled training and development partitions were used to train the SVM models. The evaluation partition was used for testing. The experiment was conducted using Weka, whereby default SVM settings other than the kernel, was used. To further improvise the detection performance, feature fusion was conducted. The fused features include MFCC, image-based features, and hexadecimal-based features. Feature fusion may cause the model generated by a classifier to overfitting, as the feature set with large numbers will often be biasedly assigned a larger weight. To mitigate this issue, feature normalization should be applied [22]. A min-max normalization described in section 2.2 is used in this experiment. Concerning the classification output, we also investigate the classification performances when probability estimates [23] are used. The probability estimates in general are used to calculate the number of times an event happened divided by the number of trials. The mostly used approach to produce probability estimates in SVM is the platt scaling. The platt scaling is often applied to a binary class problem using logistic regression model to output a probability estimates in the range of 0–1. The machine used in the experiment conducted is with the specification as: Intel i5-3210 M processor, 2.50 GHz, 8 GB of RAM, Windows 10 (64-bit) OS.

3.2. Analysis of results

EER is the primary metric to assess the performance of a biometric system, especially speaker verification. The EER is a threshold point of a biometric system at which the false acceptance rate (FAR) and false rejection rate (FRR) are equals. Hence, in this work, the EER metric was used to measure the performances of SVM models where a lower EER indicates a better performance. Table 1 showed the performances of the SVM models with different settings as described in the foregoing section.

From Table 1, there are two SVM models with different settings performed with less than 5% EER in the ASV spoof 2019 LA evaluation set. The two best settings, as Table 1, are the polynomial SVM and linear SVM, both with feature normalization, which produced 1.42% and 3.55% EER, respectively. This observation can be seen as the fused features were effective in artificial speech detection when using the appropriate SVM settings.

An interesting observation is that all SVM kernels performed the best when feature normalization is applied. This indicates that the feature normalization can produce a better result. When no normalization is applied, the features with larger values are likely to influence the prediction result intrinsically. This is because the SVM models may tend to give more weight to the features with larger values, and overfitting occurs. Therefore, normalized features produced better results by bringing all the features to the same range to reduce the probability of overfitting, but it may not always be the case.

Another observation is that the commonly best-performing kernel, the RBF was not performed well, as Table 1, although it was shown to perform well in most cases [24]. Nonetheless, it can be observed that the normalization improved the performance of the SVM with RBF kernel to 10.84% from 50% EER when no normalization was applied. Polynomial kernel SVM performed the best among the compared kernels may be due to most features included were the image-based features. Note that polynomial kernel was often used for image processing and shown to produce decent performance [25].

Table 1. Performance of SVM models with different settings in the ASVspoof 2019 logical access (LA) dataset

Setting/kernel	EER (%)			
	Radial basis function	Linear	Polynomial	Sigmoid
None (default)	50.00	17.43	8.01	50.00
Normalized	10.84	3.55	1.42	14.00
Probability estimates	50.00	18.29	16.97	50.00
Normalized + probability estimates	16.41	13.85	49.14	16.70

Some recent works that applied SVM for artificial speech detection were used in this paper for comparison. Table 2 compared the performance between the proposed approach in this paper and recent works. For better representation, the linear SVM with feature normalization and polynomial SVM with feature normalization as shown in the Table 1 were labeled as model 1 and model 2 in Table 2. A performance comparison was conducted to compare model 1 and model 2 against the recent works, namely model 3 – model 7 which used SVM as classifier on the evaluation set.

From Table 2, the model 2 performed the best among the compared recent works which used SVM as classifier. It can be observed in Table 2 that the mostly used kernel in the recent works was the polynomial. In addition, models which use polynomial SVM, namely model 2, model 3, model 5, and model 6 produced below 3% EER when detecting artificial speech. The RBF kernel often outperformed polynomial kernel especially in the case of replay attack detection [4]. However, in the case of artificial attack detection, the best kernel for SVM is the Polynomial kernel as shown in Table 2.

Table 2. Performance comparison of the proposed approach against the recent works

Model	EER (%)
Model_1: Fused features + linear SVM with feature normalization	3.55
Model_2: Fused features + polynomial SVM with feature normalization	1.42
Model_3: LFCC-ResNet18 + polynomial one class SVM [16]	2.19
Model_4: CQCC-LCNN features + linear SVM [26]	9.08
Model_5: LFCC + polynomial SVM [27]	2.92
Model_6: LFCC-GMM + GAT-S + GAT-T + RawNet2 + polynomial SVM [28]	1.68
Model_7: X-vectors + linear SVM [29]	7.12

4. CONCLUSION

In this paper, various SVM kernels were experimented to identify the best kernel for artificial speech detection when applied to the presented handcrafted features. Resampling was conducted to reduce the implication of an unbalanced dataset towards overfitting. Three categories of features were used in the experiment, namely MFCC, hexadecimal-based, and image-based features. Feature fusion was applied to improvise the performance of artificial speech detection using SVM. The ASVspoof 2019 logical access dataset was used in the experiment. Results showed that the polynomial SVM with feature normalization performed the best. Besides, it was found that feature normalization improvised the result of artificial speech detection. Future works are directed at the extraction of deep learning-based features and ensemble classification, as well as the integration of voice PAD and ASV systems.

REFERENCES

[1] C. B. Tan *et al.*, "A survey on presentation attack detection for automatic speaker verification systems: State-of-the-art, taxonomy, issues and future direction," *Multimedia Tools and Applications*, vol. 80, no. 21–23, pp. 32725–32762, 2021, doi: 10.1007/s11042-021-11235-x.

[2] C. Aroef, Y. Rivan, and Z. Rustam, "Comparing random forest and support vector machines for breast cancer classification," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 815-821, 2020, doi: 10.12928/telkomnika.v18i2.14785.

[3] I. M. Murwantara, P. Yugopuspito, and R. Hermawan, "Comparison of machine learning performance for earthquake prediction in Indonesia using 30 years historical data," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 3, pp. 1331-1342, 2020, doi: 10.12928/telkomnika.v18i3.14756.

[4] K. M. Malik, A. Javed, H. Malik, and A. Irtaza, "A light-weight replay detection framework for voice controlled IoT devices," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 982–996, 2020, doi: 10.1109/JSTSP.2020.2999828.

[5] C. K. On, P. M. Pandiyan, S. Yaacob, and A. Saudi, "Mel-frequency cepstral coefficient analysis in speech recognition," in *2006 International Conference on Computing & Informatics*, 2006, pp. 1–5, doi: 10.1109/ICOCI.2006.5276486.

[6] M. H. A. Hijazi, T. C. Beng, J. Mountstephens, Y. Lim, and K. Nisar, "Malware classification using ensemble classifiers," *Advanced Science Letters*, vol. 24, no. 2, pp. 1172–1176, 2018, doi: 10.1166/asl.2018.10710.

[7] L. Li, Y. Ding, B. Li, M. Qiao, and B. Ye, "Malware classification based on double byte feature encoding," *Alexandria Engineering Journal*, vol. 61, no. 1, pp. 91–99, 2022, doi: 10.1016/j.aej.2021.04.076.

[8] M. M. S. -Alvarez, D. -T. Pham, M. Y. Prostov, and Y. I. Prostov, "Statistical approach to normalization of feature vectors and clustering of mixed datasets," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 468, no. 2145, pp. 2630–2651, 2012, doi: 10.1098/rspa.2011.0704.

[9] Y. Jia *et al.*, "Speaker recognition based on characteristic spectrograms and an improved self-organizing feature map neural network," *Complex & Intelligent Systems*, vol. 7, no. 4, pp. 1749–1757, 2021, doi: 10.1007/s40747-020-00172-1.

[10] K. S. Rao, V. R. Reddy, and S. Maity, "Language identification using prosodic features," 2015, pp. 55–81, doi: 10.1007/978-3-319-17163-0_4.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009, doi: 10.1145/1656274.1656278.

[12] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996, doi: 10.1016/0031-3203(95)00067-4.

[13] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998, doi: 10.1109/5254.708428.

[14] M. H. A. Hijazi, C. Jiang, F. Coenen, and Y. Zheng, "Image classification for age-related macular degeneration screening using hierarchical image decompositions and graph mining," 2011, vol. 6912, pp. 65–80, doi: 10.1007/978-3-642-23783-6_5.

[15] J. Villalba, A. Miguel, A. Ortega, and E. Lleida, "Spoofing detection with DNN and one-class SVM for the ASVspoof 2015 challenge," in *Interspeech 2015*, 2015, pp. 2067–2071, doi: 10.21437/Interspeech.2015-468.

[16] Y. Zhang, F. Jiang, and Z. Duan, "One-class learning towards synthetic voice spoofing detection," *IEEE Signal Processing Letters*, vol. 28, pp. 937–941, 2021, doi: 10.1109/LSP.2021.3076358.

[17] N. Kalcheva, M. Karova, and I. Penev, "Comparison of the accuracy of SVM kemel functions in text classification," in *2020 International Conference on Biomedical Innovations and Applications (BIA)*, 2020, pp. 141–145, doi: 10.1109/BIA50171.2020.9244278.

[18] H. -T. Lin and C. -J. Lin, "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods," *Neural Computation*, pp. 1–32, 2003. [Online]. Available: https://www.researchgate.net/profile/Hsuan-Tien-Lin/publication/2478380_A_Study_on_Sigmoid_Kernels_for_SVM_and_the_Training_of_non-PSD_Kernels_by_SMO-type_Methods/links/00b7d5141eff00b686000000/A-Study-on-Sigmoid-Kernels-for-SVM-and-the-Training-of-non-PSD-Kernels-by-SMO-type-Methods.pdf

[19] G. F. Smits and E. M. Jordaan, "Improved SVM regression using mixtures of kernels," in *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, 2022, pp. 2785–2790, vol. 3, doi: 10.1109/IJCNN.2002.1007589.

[20] X. Wang *et al.*, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, 2020, doi: 10.1016/j.csl.2020.101114.

[21] M. Todisco *et al.*, "ASVspoof 2019: future horizons in spoofed and fake audio detection," in *Proc. Interspeech 2019*, 2019, pp. 1008–1012, doi: 10.21437/Interspeech.2019-2249.

[22] A. -B. Al-Ghamdi, S. Kamel, and M. Khayyat, "Evaluation of artificial neural networks performance using various normalization methods for water demand forecasting," in *2021 National Computing Colleges Conference (NCCC)*, 2021, pp. 1–6, doi: 10.1109/NCCC49330.2021.9428856.

[23] T. Leathart, E. Frank, G. Holmes, and B. Pfahringer, "Probability calibration trees," *Proceedings of the 9th Asian Conference on Machine Learning Research*, 2018, vol. 77, pp. 145–160, doi: 10.48550/arXiv.1808.00111.

[24] F. F. Tampinongkol, Y. Herdiyeni, and E. N. Herliyana, "Feature extraction of Jabon (Anthocephalus sp) leaf disease using discrete wavelet transform," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, pp. 740-751, 2020, doi: 10.12928/telkomnika.v18i2.10714.

[25] N. Arizumi, "Piecewise polynomial approximation method for convolution with large kernel," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 3080–3083, doi: 10.1109/ICIP40778.2020.9191304.

[26] S. -Y. Chang, K. -C. Wu, and C. -P. Chen, "Transfer-representation learning for detecting spoofing attacks with converted and synthesized speech in automatic speaker verification system," in *Proc. Interspeech 2019*, 2019, pp. 1063–1067, doi: 10.21437/Interspeech.2019-2014.

[27] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing attack detection using the non-linear fusion of sub-band classifiers," in *Proc. Interspeech 2020*, 2020, pp. 1106–1110, doi: 10.21437/Interspeech.2020-1844.

[28] H. Tak, J. -W. Jung, J. Patino, M. Todisco, and N. Evans, "Graph attention networks for anti-spoofing," in *Proc. Interspeech 2021*, 2021, pp. 2356–2360, doi: 10.21437/Interspeech.2021-993.

[29] A. G. -Alanis, J. A. G. -Lopez, S. P. Dubagunta, A. M. Peinado, and M. M. -Doss, "On joint optimization of automatic speaker verification and anti-spoofing in the embedding space," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1579–1593, 2021, doi: 10.1109/TIFS.2020.3039045.

BIOGRAPHIES OF AUTHORS

Choon Beng Tan received his BCompSc. and MSc. degrees from Universiti Malaysia Sabah (UMS) in 2016 and 2018. He is now doing his PhD study in Computer Science in Universiti Malaysia Sabah (UMS). His recent work includes malware classification using machine learning and ensemble techniques, and cloud data integrity scheme; he is now working on voice presentation attack detection. His research interests include information security, cloud computing, and voice biometric security. He can be contacted at email: tanchoonbeng@ums.edu.my.

Mohd Hanafi Ahmad Hijazi is an Associate Professor of Computer Science at the Faculty of Computing and Informatics, Universiti Malaysia Sabah in Malaysia. His research work addresses the challenges in knowledge discovery and data mining to identify patterns for prediction on structured and/ or unstructured data; his particular application domains are medical image analysis and understanding and sentiment analysis on social media data. He has authored/ co-authored more than 50 journals/ book chapters and conference papers, most of which are indexed by Scopus and ISI Web of Science. He also served on the program and organizing committees of numerous national and international conferences. He is the leader of the Data Technologies and Applications research group at the faculty. He can be contacted at email: hanafi@ums.edu.my.

Puteri Nor Ellyza Nohuddin received her BSc. in Computer Science from University of Missouri-Columbia, USA and her MSc IT from Universiti Teknologi MARA. In 2012, she was awarded her Ph.D. in Computer Science from the University of Liverpool, UK. Puteri joins Institute of IR4.0 (IIR4.0), Universiti Kebangsaan Malaysia as a Research Fellow in July 2015. Prior to coming to IIR4.0, she was lecturer at the Universiti Pertahanan Nasional Malaysia, Kuala Lumpur. Prior to her academic career, she worked with several conglomerates such as ExxonMobil, Sime Darby, Shell IT and Malaysian Resources Corporation Berhad as System Analyst. Puteri’s teaching interests include Programming, Database systems and Data mining. Her primary research interests are in the field of Big Data, Data Mining and Knowledge Engineering. Specifically, she is interested in Time Series Clustering, Trend mining, Tacit Knowledge, and Social Network Analysis. She can be contacted at email: puteri.ivi@ukm.edu.my.