International Journal of Electrical and Computer Engineering (IJECE)
Vol. 12, No. 4, August 2022, pp. 4391~4399
ISSN: 2088-8708, DOI: 10.11591/ijece.v12i4.pp4391-4399

Principal coefficient encoding for subject-independent human activity analysis

Pang Ying Han[1], Sarmela Anak Perempuan Raja Sekaran[1], Ooi Shih Yin[1], Tan Teck Guang[2]
[1]Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia
[2]DRSoft Sdn Bhd, Melaka, Malaysia

Article Info

Article history:

Received Apr 30, 2021
Revised Jan 19, 2022
Accepted Feb 3 2022

Keywords:

1D inertial motion data
Autoencoder
Deep analytic model
Principal coefficient encoding
Subject-independent

ABSTRACT

Tracking human physical activity using smartphones is an emerging trend in healthcare monitoring and healthy lifestyle management. Neural networks are broadly used to analyze the inertial data of activity recognition. Inspired by the autoencoder neural networks, we propose a layer-wise network, namely principal coefficient encoder model (PCEM). Unlike the vanilla neural networks which apply random weight initialization and back-propagation for parameter updating, an optimized weight initialization is implemented in PCEM via principal coefficient learning. This principal coefficient encoding allows rapid data learning with no back-propagation intervention and no gigantic hyperparameter tuning. In PCEM, the most principal coefficients of the training data are determined to be the network weights. Two hidden layers with principal coefficient encoding are stacked in PCEM for the sake of deep architecture design. The performance of PCEM is evaluated based on a subject-independent protocol where training and testing samples are from different users, with no overlapping subjects in between the training and testing sets. This subject-independent protocol can better assess the generalization of the model to new data. Experimental results exhibit that PCEM outperforms certain state-of-the-art machine learning and deep learning models, including convolutional neural network, and deep belief network. PCEM can achieve ~97% accuracy in subject-independent human activity analysis.

Corresponding Author:

Pang Ying Han
Faculty of Information Science and Technology, Multimedia University
75450 Ayer Keroh, Melaka, Malaysia
Email: yhpang@mmu.edu.my

1. INTRODUCTION

Human activity recognition (HAR) becomes more prominent with the increasing and advancement of the smart home concept, healthcare monitoring and healthy lifestyle management [1]. Using remote healthcare monitoring tools on smartphones is very common and prevalent in remote mobile health monitoring (RMHM) systems. The adoption of smartphones in HAR is made competent with its ability to capture motion data through its multiple inertial sensors as well as its significant attachment in our daily life [2]. With a simple and straightforward installation process, HAR app can be activated on the smartphone and running in the background to track our physical activity.

Smartphone-based HAR is usually developed in four phases: data acquisition, data segmentation, feature extraction and classification. During data acquisition, various factors such as the position of the smartphone, and collection frequency are considered. In literature, different positions' placements of the smartphone have been investigated, e.g.: in front pocket, back pocket, in hand, on waist, typing, and phoning

[1]–[4]. Multiple sampling frequencies are also explored and used in collecting data [4]–[6]. There is a certain degree of influence of the data segmentation size on the recognition performance. Thus, various segmentation sizes have been examined with different sliding window timings of 2.56 s, 1 s, 6.7 s, 7.5 s and 10 s [1], [3], [7] during the data segmentation phase.

Upon the completion of data segmentation phase, the salient features shall be selected from the samples. This phase can be known as feature extraction. Variant feature extraction techniques have been explored in the past, these include those from hand-crafted to the adoption of deep learning approaches [8]–[10]. Group-based context-aware classification method for human activity recognition on smartphones, namely group-based context-aware human activity recognition (GCHAR), was proposed by Cao *et al.* [9]. In GCHAR, a hierarchical group-based scheme is adopted to lessen the misclassification through context-awareness in lieu of intensive computation. Inter-group and intra-group hierarchical classification schemes are constructed to diminish the prediction load for each classification level. The context-awareness is engaged to detect the transitions among activities. Compared with RandomTree, Bagging, J48, and BayesNet, the supremacy of GCHAR is proven in terms of model training efficiency and classification performance. Ahmed *et al.* [11] proposed a hybrid approach, utilizing filter and wrapper practices, for feature selection in HAR. In this approach, a sequential floating forward search is implemented to obtain optimal features. These chosen features are further analyzed by using support vector machine (SVM) specifically in the task of data classification. Experimental results showed that this feature selection-based system was able to achieve 6% higher accuracy compared to those without feature selection. The only downside of these hand-crafted techniques is prior expert knowledge or rigorous empirical study is extensively needed for feature engineering. In recent years, numerous deep learning methods were proposed [6], [12]–[14]. Deep learning methods perform automated feature extraction and produce a more detailed abstraction for data representation when the network grows deeper. For instances, convolutional neural network (CNN) [1], [15], autoencoder [16], deep belief network (DBN) [17], recurrent neural network (RNN) [5], long short term memory (LSTM) [18], [19] based techniques have been rigorously explored to extract the deep features of motion inertial signal for HAR. The rightful selection of features is definitely useful in determining the recognition rate of the entire system later.

The last step of recognizing HAR is through the usage of classifiers. The classification model is built based on the extracted feature set from the previous step. Various machine learning approaches, either single standalone classifier or fusion of multiple classifiers, have been widely studied by the field experts. Notable classifiers such as decision tree (DT), multilayer perceptron (MLP), SVM, random forest (RF), logistic regression, and extreme gradient boost (XGB) were examined in the smartphone-based HAR [2], [3], [20], [21]. The good performance of deep learning methods in pattern recognition is undeniable. However, the exceptional accomplishment of these approaches is contingent upon enormous training samples for model generalization and high-performance hardware requirements to support expensive computational loads [22]. Besides, some parameters need to be initialized or even tuned in the deep neural networks, such as random weights initialization and back-propagation weight tuning. Contrary to these classic deep learning networks, we propose a deep analytic model to analyze the human motion data with fast weight initialization without the usage of back-propagation. Inspired by the autoencoder architecture, this proposed system is a layer-wise network, namely principal coefficient encoding model (PCEM). The main contributions of this paper are:

a. A subject-independent human activity recognition with a deep analytic model that can generalize well to new data without a massive training set. In PCEM, model training is performed based on data samples of a group of subjects, while the system efficacy is tested on the samples of another group of subjects.
b. An optimized principal coefficient weight initialization in the neural layers allows quick data learning with no back-propagation intervention and no gigantic hyperparameter tuning. Contrary to the classic deep learning systems which require graphics processing unit (GPU) for computation, the model training of principal coefficient encoder model (PCEM) applies only a central processing unit (CPU) due to its light computation.
c. An extensive performance analysis with two different classification modes. The effectiveness of PCEM is examined in two-class classification (i.e. distinguishing active and passive physical activities) and multiclass classification (i.e. distinguishing the types of each activity).

2. THE PROPOSED METHOD

In this work, smartphone embedded sensors are utilized in performing human activity recognition. The overall architecture of the proposed PCEM is depicted in Figure 1. From the figure, we can notice that the proposed architecture comprises four stages: data acquisition, data preprocessing, feature extraction via principal coefficient encoding and data classification.

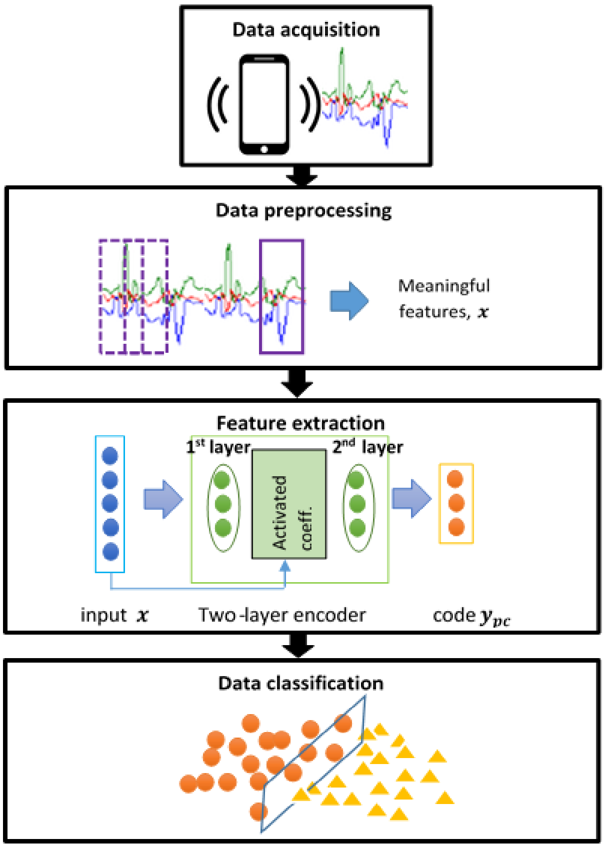

Figure 1. The architecture of the proposed PCEM

2.1. Data preprocessing

The acceleration and angular velocity signals are preprocessed via a low-pass filter to reduce noise. A sliding window of 2.56 s and fifty percent overlap is implemented to segment the waveform signals. The segmented data is further processed to generate meaningful feature variables as illustrated in Figure 2.

Average value	Signal entropy
Standard deviation	Auto-regression coefficients
Median absolute value	Correlation coefficient
Largest value in array	Largest frequency component
Smallest value in array	Frequency signal weighted average
Signal magnitude area	Frequency signal skewness
Average sum of squares	Frequency signal kurtosis
	Energy of a frequency interval
Interquartile range	Angle between two vectors

Figure 2. Process data into feature variables

2.2. Feature extraction: principal coefficient encoding

In autoencoder, data input x is multiplied with a weight matrix W with a bias vector b. The weights and biases are initialized randomly and then updated iteratively through backpropagation. Then, a nonlinear activation function f is applied to obtain the encoder's output, called code y.

$$y = f(W.x + b) \tag{1}$$

In the principal coefficient encoder, normalized data is firstly computed by subtracting the mean vector m from each of the data dimensions from the input data x. Next, the resultant data is multiplied by an orthogonal matrix (i.e. weight matrix) V to obtain the result $\check{y}_{pc}$.

$$\breve{y}_{pc} = V(x - m) = V.x - V.m \quad (2)$$

To produce a nonlinear model to better cater the complex real-world data, a nonlinear input-output mapping function f is applied to $\breve{y}_{pc}$ to obtain code y_{pc}.

$$y_{pc} = f(V.x - V.m) \quad (3)$$

Let $x_0 = \vec{0}$, we will have

$$f\left(V.\vec{0} - V.m\right) = f\left(W.\vec{0} + b\right) \quad (4)$$

Let the activation function f is a strictly monotonic function, $b = -V.m$ is obtained. The result is injected in (3) and gets an equivalent expression in (1).

$$f(V.x + b) = f(W.x + b) \quad (5)$$

Hence, we achieve $W = V$. In this study, we apply parametric rectified linear unit as the nonlinear activation function, $f(z) = max(\propto z, z)$ where $\propto = [0, 1]$. In this work, two hidden layers with principal coefficient encoding are developed for multi-layer feature extraction to learn data representation with multiple levels of abstraction, see Figure 1. The intermediate activated coefficients are combined with x and further analyzed in the second hidden layer to encode deeper features.

2.3. Classification

Owing to the flexibility characteristic of SVM, varieties of classification problems can be resolved with minimal tuning. On top of that, the automatic complexity control in SVM is able to solve overfitting concerns. The real-world data which is randomly distributed in a nonlinear way can be tackled via an adequate kernel trick in SVM to bridge linearity to nonlinearity. In this work, a nonlinear SVM is employed to generate a decision boundary for classifying the extracted code. The idea of the nonlinear classifier is fairly analogous to the linear SVM, but a kernel function is applied to represent the similarity of vectors (i.e. codes) in a kernelized feature space over polynomials of the original variables. This allows the learning of nonlinear modelling. The kernel implementation allocates the data in a higher-dimensional space so that a decision hyperplane can be structured in this new kernel feature space, as illustrated in Figure 3.

The mapping from an original input space into a kernelized feature space is formulated:

$$y \longmapsto \Phi(y) \quad (6)$$

In other words, the function g is computed for the mapping:

$$g(\boldsymbol{y}) = w.\Phi(\boldsymbol{y}) + d \quad (7)$$

Computing Φ for each sample is rather inefficient. The kernelized feature space is in a very high dimensionality or even with infinite dimensions, resulting in the hardness representation of w in memory. Hence, kernel function K, $K(i,j) = \Phi(i).\Phi(j)$, is implemented to avoid explicit computation of each Φ, refer [23].

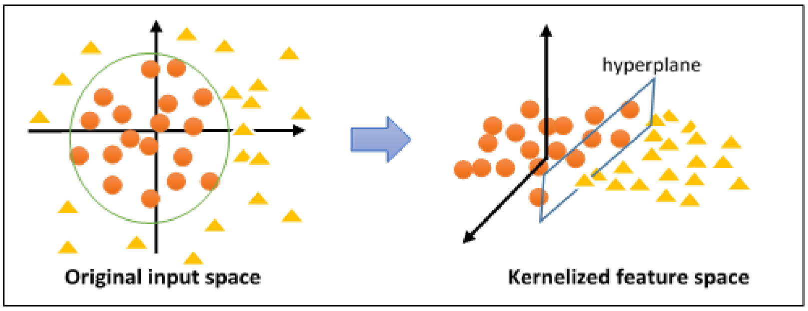

Figure 3. Nonlinear SVM mapping

3. RESULTS AND DISCUSSION

To evaluate the classification performance, as well as the generalization ability of the proposed PCEM model, we test it on the human static-postures and dynamic-activities (HSD) database. In this database, there are three static postures (sitting, standing and laying) and three dynamic activities (walking, walking upstairs and walking downstairs). During the data collection process, thirty volunteers (19-48 years old) were carrying a Samsung Galaxy S II smartphone (bundled with accelerometer and gyroscope sensors) on the waist. The total accelerometer, estimated body accelerometer and gyroscope data were captured at a 50 Hz rate. The process of data collection was disclosed in detail in [2]. In this work, four performance evaluation metrics are employed, which are classification accuracy, precision, recall and F1-score. The experiments are conducted using CPU with Intel (R) Core i7-7700K 4.2 GHz and RAM 48 GB on Matlab R2018a platform.

3.1. Parameter performance analysis

Parametric rectified linear unit function is introduced in the principal coefficient encoder for a nonlinear input-output mapping to better model the complex real-world data. In this experiment, the influence of the activation function parameter ∝ is examined. Figure 4 illustrates the performance measures of different ∝ values. It is noticed that the system performance is quite stable across different ∝ values, except ∝=0.1 with about 1% lower accuracy. We adopt ∝=0.5 for the subsequent experiments.

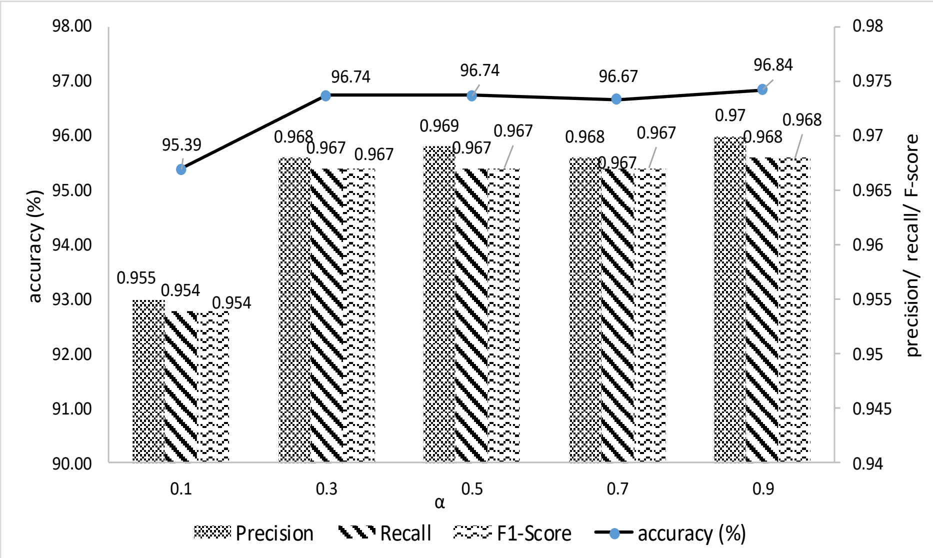

Figure 4. Performance of different ∝

Next, the performance analysis with different dimension reductions in the hidden layers of PCEM is examined, as illustrated in Figure 5. From the empirical results, we observe that the system performance is improving when the reduced or preserved dimension is getting higher. There is an impressive performance improvement from the dimension of 20% (i.e. only 20% dimensional features are preserved) to 50%, from 94.4% to 96.8% accuracy score. However, when more and more dimensions are preserved (i.e. higher-dimensional code), PCEM exhibits a slight performance degradation. As observed in Figure 5, when excessive dimensional features are preserved (>60%), there is a slight accuracy degradation. This could be attributed to the presence of the uninformative components (i.e. noise) in the code, rising the misclassification. Real-world data is embedded with correlated features. The redundant information is treated as noise which could pessimistically affect the classification learning model. A reasonable dimensionality reduction in PCEM (50%-dimension reduction in this case) helps eliminate the irrelevant and redundant data, producing effective code which is useful for data classification. Hence, 50%-dimension reduction is employed.

3.2. System performance analysis

We examine the efficiency of PCEM in two modes: i) two-class classification: human activities are classified based on intensity level-active or passive activity and ii) multiclass classification: human activities are classified into one of C possible classes. Walking, climbing stairs and jogging exhibit periodic behavior patterns generated by the repetitive movements of the activities. We categorize those activities with periodic movement patterns as an active class in the two-class classification mode; whist, activities of sitting, laying and standing are grouped as a passive class. As aforementioned, the efficiency of PCEM is assessed as a subject-independent solution. In other words, PCEM is trained using training samples from a group of users. Then, the model is applied to new users without the necessity of collecting additional samples of these new users to retrain the model. In this experiment, HSD dataset is partitioned into two sets: 70% of the volunteers are selected to generate the training samples and the remaining 30% of the volunteers' data is used for testing. PCEM needs 10.0275 s to train the model with 7,352 samples and 1.0724 s to test 2,947 samples. Table 1 summarizes the performances of PCEM in two-class and multiclass classification respectively. From the results, we notice that the proposed model excels in classifying active and passive activities where it can achieve 100% accuracy. PCEM also achieves an excellent score in F-score, precision and recall. This indicates that the model generates zero false positives and false negatives in distinguishing active and passive activities. Figure 6 illustrates the confusion matrices of the two modes. There is no misclassification between the active and passive activities as shown in Figure 6(a) and low misclassification among activity classes as shown in Figure 6(b).

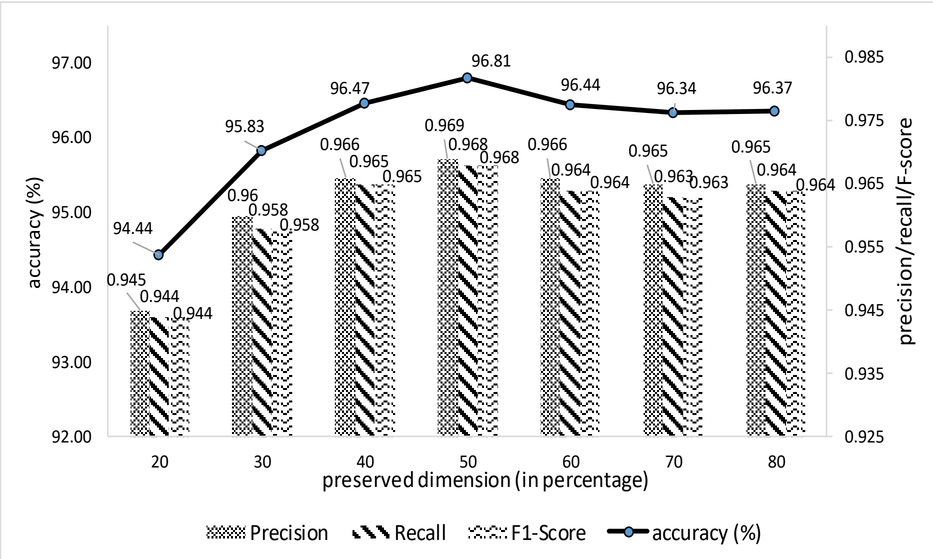

Figure 5. Performance of different dimensions in the hidden layers

Table 1. Performance of PCEM in two-class and multiclass classifications

		Precision	Recall	F-score	Overall Accuracy (%)
Two-class classification	Passive	1.000	1.000	1.000	100
	Active	1.000	1.000	1.000	
	Average	1.000	1.000	1.000	
Multiclass classification	Stand	0.904	0.976	0.939	96.8103
	Sit	0.971	0.884	0.925	
	Lay	1.000	1.000	1.000	
	Walk	0.965	0.998	0.981	
	Downstair	1.000	0.974	0.987	
	Upstair	0.983	0.975	0.979	
	Average	0.969	0.968	0.968	

The performance of PCEM is slightly dropped in multiclass classification. The model obtains approximately 97% in classifying different types of activities. Though it is not as perfect as the two-class model, it is still an encouraging observation. From the empirical results, we notice that PCEM records lower F-score in stand and sit classes, indicating more false positives and negatives in distinguishing stand and sit classes as illustrated in the confusion matrix. The resemble almost-constant patterns of these stationary activities may be the reason for the misclassification. But, PCEM is still able to extract the underlying distinct patterns of the data with minor false positives and negatives.

```
=== Confusion Matrix ===

    a    b   <-- classified as
 1560    0 |    a = passive
    0 1387 |    b = active
```

(a)

```
=== Confusion Matrix ===

   a   b   c   d   e   f   <-- classified as
 519  13   0   0   0   0 |   a = standing
  55 434   0   0   0   2 |   b = sitting
   0   0 537   0   0   0 |   c = laying
   0   0   0 495   0   1 |   d = walking
   0   0   0   6 409   5 |   e = downstair
   0   0   0  12   0 459 |   f = upstair
```

(b)

Figure 6. Confusion matrix of PCEM in (a) two-class and (b) multiclass activity classifications

3.3. Comparison and discussion

In this experiment, we compare the performance of other models with the proposed PCEM. For a fair comparison, the same database is used, and the subject-independent protocol is implemented. Table 2 shows the classification accuracy of various approaches in HAR on HSD database.

From the empirical results, we can notice that: i) The proposed PCEM demonstrates a superior performance than other approaches. This is because the optimized principal coefficient learning and discriminant classifier in PCEM competently analyze the motion features from lower to deeper level via its stacking layer-wise architecture; ii) The performances of CNN and deep network (DBN) are slightly poorer than that of PCEM, achieving ~95% accuracy. This could be due to the insufficient training sample to generalize and optimize the model learning. Besides, CNN and DBN suffer from training efficiency [14], which grow in computational complexity with the number of layers. On the other hand, PCEM is a fast-analytic solution owing to no back-propagation and no puzzling parameter tuning; iii) In this HAR application, the stacked autoencoder is not able to perform well. This is because the number of samples is insufficient, affecting its learning ability. With limited training samples, it is difficult for the model to obtain full features learned through encoding learning; and iv) PCEM is shown to be an auspicious solution in HAR with minimal training efforts (no iterative learning and back-propagation). This analytic solution is trainable without GPU, but only CPU.

Table 2. System comparison of the existing approaches in HAR

Approach	Accuracy (%)
CNN* [24]	95.75
ANN* (reported in [24])	91.08
GCHAR* [9]	94.16
Deep Belief Network* (reported in [17])	95.80
Hierarchical Continuous Hidden Markov Model* [25]	93.18
Stacked autoencoder* [16]	89.64
PCEM	96.81

* Results are extracted from the respective papers

4. CONCLUSION

Inspired by the autoencoder neural network, a layer-wise network, called as PCEM, is presented. The core difference between the autoencoder and PCEM is the former initializes the weights with random values and applies back-propagation for parameter/weight update; but the later implements a principal coefficient weight initialization that allows rapid data learning with no back-propagation intervention. Unlike other classic deep neural networks (e.g. CNN), there is no gigantic hyperparameter tuning in PCEM. In this work, a subject-independent testing protocol is implemented to evaluate the performance of PCEM. This subject-independent protocol can better assess the generalization of the model in recognizing new data. Empirical results demonstrate the superiority of PCEM to the other machine learning and deep learning models, with recognition accuracy ~97% in the activity recognition. Furthermore, the model training time for PCEM is lesser than one minute, whereas most classic deep networks require hours to train the model due to the complex computation and enormous parameter tuning.

ACKNOWLEDGEMENTS

This research is supported by Fundamental Research Grant Scheme (FRGS), FRGS/1/2020/ICT02/MMU/02/7. Authors acknowledged Ministry of Higher Education Malaysia (MOHE).

REFERENCES

[1] G. Wang, Q. Li, L. Wang, W. Wang, M. Wu, and T. Liu, "Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors," *Sensors*, vol. 18, no. 6, Jun. 2018, doi: 10.3390/s18061965.

[2] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the 21th international European symposium on artificial neural networks, computational intelligence and machine learning*, 2013, pp. 437--442.

[3] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, Mar. 2011, doi: 10.1145/1964897.1964918.

[4] D. Micucci, M. Mobilio, and P. Napoletano, "UniMiB SHAR: A dataset for human activity recognition using acceleration data from smartphones," *Applied Sciences*, vol. 7, no. 10, Oct. 2017, doi: 10.3390/app7101101.

[5] M. Inoue, S. Inoue, and T. Nishida, "Deep recurrent neural network for mobile human activity recognition with high throughput," *Artificial Life and Robotics*, vol. 23, no. 2, pp. 173–185, Jun. 2018, doi: 10.1007/s10015-017-0422-x.

[6] B. Almaslukh, A. Artoli, and J. Al-Muhtadi, "A robust deep learning approach for position-independent smartphone-based human activity recognition," *Sensors*, vol. 18, no. 11, Nov. 2018, doi: 10.3390/s18113726.

[7] P. Siirtola and J. Röning, "User-independent human activity recognition using a mobile phone: offline recognition vs. real-time on device recognition," in *Advances in Intelligent and Soft Computing*, vol. 151 AISC, 2012, pp. 617–627.

[8] S. Xu, Q. Tang, L. Jin, and Z. Pan, "A cascade ensemble learning model for human activity recognition with smartphones," *Sensors*, vol. 19, no. 10, May 2019, doi: 10.3390/s19102307.

[9] L. Cao, Y. Wang, B. Zhang, Q. Jin, and A. V. Vasilakos, "GCHAR: an efficient group-based context-aware human activity recognition on smartphone," *Journal of Parallel and Distributed Computing*, vol. 118, pp. 67–80, Aug. 2018, doi: 10.1016/j.jpdc.2017.05.007.

[10] W. Sousa Lima, E. Souto, K. El-Khatib, R. Jalali, and J. Gama, "Human activity recognition using inertial sensors in a smartphone: an overview," *Sensors*, vol. 19, no. 14, Jul. 2019, doi: 10.3390/s19143213.

[11] N. Ahmed, J. I. Rafiq, and M. R. Islam, "Enhanced human activity recognition based on smartphone sensor data using hybrid feature selection model," *Sensors*, vol. 20, no. 1, Jan. 2020, doi: 10.3390/s20010317.

[12] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, vol. 119, no. 1, pp. 3–11, Mar. 2019, doi: 10.1016/j.patrec.2018.02.010.

[13] C. Chen *et al.*, "Deep learning on computational-resource-limited platforms: a survey," *Mobile Information Systems*, vol. 2020, pp. 1–19, Mar. 2020, doi: 10.1155/2020/8454327.

[14] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: a brief review," *Computational Intelligence and Neuroscience*, vol. 2018, pp. 1–13, 2018, doi: 10.1155/2018/7068349.

[15] A. Ignatov, "Real-time human activity recognition from accelerometer data using convolutional neural networks," *Applied Soft Computing*, vol. 62, pp. 915–922, Jan. 2018, doi: 10.1016/j.asoc.2017.09.027.

[16] H. Badem, A. Caliskan, A. Basturk, and M. E. Yuksel, "Classification of human activity by using a stacked autoencoder," in *2016 Medical Technologies National Congress (TIPTEKNO)*, Oct. 2016, pp. 1–4, doi: 10.1109/TIPTEKNO.2016.7863135.

[17] H. Li and M. Trocan, "Deep learning of smartphone sensor data for personal health assistance," *Microelectronics Journal*, vol. 88, pp. 164–172, Jun. 2019, doi: 10.1016/j.mejo.2018.01.015.

[18] F. Hernandez, L. F. Suarez, J. Villamizar, and M. Altuve, "Human activity recognition on smartphones using a bidirectional LSTM network," in *2019 XXII Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, Apr. 2019, pp. 1–5, doi: 10.1109/STSIVA.2019.8730249.

[19] R. Wainwright and A. Shenfield, "Human activity recognition making use of long short-term memory techniques," *Athens Journal of Sciences*, vol. 6, no. 1, pp. 19–34, Feb. 2019, doi: 10.30958/ajs.6-1-2.

[20] B. Kolosnjaji and C. Eckert, "Neural network-based user-independent physical activity recognition for mobile devices," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9375, 2015, pp. 378–386.

[21] D. N. Tran and D. D. Phan, "Human activities recognition in android smartphone using support vector machine," in *2016 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS)*, 2016, pp. 64–68, doi: 10.1109/ISMS.2016.51.

[22] P. Jawandhiya, "Hardware design for machine learning," *International Journal of Artificial Intelligence and Applications*, vol. 9, no. 1, pp. 63–84, Jan. 2018, doi: 10.5121/ijaia.2018.9105.

[23] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[24] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, vol. 59, pp. 235–244, Oct. 2016, doi: 10.1016/j.eswa.2016.04.032.

[25] C. A. Ronao and S.-B. Cho, "Recognizing human activities from smartphone sensors using hierarchical continuous hidden Markov models," *International Journal of Distributed Sensor Networks*, vol. 13, no. 1, Jan. 2017, doi: 10.1177/1550147716683687.

BIOGRAPHIES OF AUTHORS

Pang Ying Han received her B.E. (Hons) degree in Electronic Engineering in year 2002, Master of Science degree in year 2005 and PhD degree in year 2013 from Multimedia University. She is currently a lecturer in Faculty of Information Science and Technology, Multimedia University. Her research interests include biometrics, human activity recognition, machine learning, inertial data processing, image processing, and pattern recognition. She can be contacted at email: yhpang@mmu.edu.my.

Sarmela received her B.I.T (Hons) (Artificial Intelligence) in 2019 from Multimedia University. She is currently pursuing PhD. (I.T.) in Faculty of Information Science and Technology, Multimedia University. Her research interests include human activity recognition, deep learning, machine learning, inertial data processing, image processing, and pattern recognition. She can be contacted at email: 1161303922@student.mmu.edu.my.

Ooi Shih Yin received the Bachelor of Information Technology (Hons), Master of Science (Information Technology), and PhD (Information Technology) from the Multimedia University, Malaysia, in 2017. From 2018 to 2019, she was a research fellow with the School of Electrical and Electronic Engineering, College of Engineering, Yonsei University, South Korea. Since 2019, she has been a senior lecturer with the Faculty of Information Science and Technology, Multimedia University, Malaysia. She is the author of more than 50 articles, and more than 10 inventions. Her research interests include temporal classification, tree-based algorithms, and machine learning applications in the field of biometrics and cybersecurity. She can be contacted at email: syooi@mmu.edu.my.

Tan Teck Guang received his degree in Artificial Intelligence from Multimedia University in year 2014. Currently, he is working at DRSoft Sdn Bhd as a software developer. He has been working in IT research and development industry, as well as Fintech solution for over 6 years. His research interests include pattern recognition, software engineering, architecture design, and financial technology. He can be contacted at email: ray.tan8139@gmail.com.