Development of Sensor Data Fusion and Optimized Elman Neural Model-based Sign Language Recognition System

*Yijuan Liang, Chaiyan Jettanasen**

School of Engineering, King Mongkut's Institute of Technology Ladkrabang, Thailand
yijuan_liang@126.com, chaiyan.je@kmitl.ac.th

Abstract

The sign language recognition system has placed an important role in disabled people's lives. The researchers utilize various methods to fulfill disabled people's requirements. However, the methods fail to access their sign at a reasonable cost with minimum computational difficulties. The improper sign access causes a reduction of the sign language recognition accuracy. Therefore, this study uses the sensor of wearable devices to capture people's motions and actions to identify sign language. The collected information is fused into a competition level that understands disabled people's requirements. After that, fused information is processed by the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM). In addition, the data augmentation process is incorporated to train the data, which helps to recognize the large volume of sentences. Here, the fused information is split into words processed by the neural model that recognizes the sign language features. The extracted features are analyzed by an optimized method that recognizes the language with maximum accuracy. During the analysis, network parameters are hyper-tuned with the help of the Pareto model, which reduces the misrecognition error rate. Then, the created system efficiency is evaluated using the experimental analysis. The POHDENM approach achieves a high accuracy of 97.86% and an F1-score of 98.08%. The model's high performance is achieved by fine-tuning its hyperparameters using the Pareto optimization algorithm, which balances precision of 98.06% and recall of 98.12%.

Keywords: Competition fusion level, Pareto Optimized Hyper-Tuned Deep Elman Neural Model, Sign language, Wearable devices

1 Introduction

Sign language recognition using wearable devices is a technology that can potentially change how individuals who are deaf or hard of hearing communicate with others [1]. This technology has been developed to facilitate communication by detecting and interpreting the movements of the signer's hands and fingers [2]. The technique relies on machine learning algorithms that have been trained to identify and understand the unique movements of the hands and postures associated with various sign languages [3]. The portable devices used for deciphering signs may be worn on the signer's wrist, arm, or hand and are compact and light [4]. These devices can capture and decode the signer's finger and thumb movements by incorporating sensors, cameras, and other hardware. Afterward, the collected data is sent to a computer or other device, processed by machine learning algorithms, and converted into text or voice.

Machine learning is essential for sign language identification using wearable devices [5]. The system is trained by machine learning algorithms to identify and decode the unique movements and gestures associated with various signs [6]. Sign language recognition is a way to bridge the communication gap between hearing-impaired people and others, using a convolutional graph neural network (GCN) architecture with spatial attention mechanism. The proposed architecture shows outstanding results on different datasets [7]. Because the technology can convert sign language into written or spoken English, it can reach a broader audience. Furthermore, machine learning techniques continuously improve the platform's efficacy and reliability. Algorithms are updated and fine-tuned to increase recognition precision when additional data and feedback are sent into the system [8]. The system's ability to learn and change over time is key to providing dependable sign language recognition.

Many possibilities exist for improving the lives of the deaf and hard of hearing via sign language recognition technology [9]. One of the key benefits is that it gives a mode of communication that is easier to use and more widely available than conventional techniques [10]. The devices used to recognize sign language are compact and lightweight, making them convenient to take anywhere. In addition, the necessity for an expensive and often transient sign language interpreter is mitigated by this technological advancement [11]. Increased social acceptance and accessibility are other benefits of sign language recognition technology. Communication difficulties are a major cause of social exclusion and isolation for those who are deaf or hard of hearing. Wearable technologies that understand sign language may be an important tool in removing these obstacles for people who are deaf or hard of hearing, allowing them to take their rightful place as contributing members of society [12]. The use of sign language recognition technology has several advantages, yet it comes with some drawbacks. The fact that the technology is only now beginning to take effect is an important obstacle. The input data quality and the

*Corresponding Author: Chaiyan Jettanasen; E-mail: chaiyan.je@kmitl.ac.th
DOI: https://doi.org/10.70003/160792642024092505003

device's environment may affect accuracy and dependability. The system's accuracy is vulnerable to environmental factors like illumination and noise level and user inputs like signing speed and fluidity. [13]. Another issue with sign language identification systems is that they may not be able to recognize all regional sign language variants.

A machine-learning system might find it difficult to correctly interpret all signals because of numerous variations and dialects within sign languages. Some users may find their experience with the technology limited due to this. Access to information and expression for the deaf and dumb might be greatly enhanced by wearable technology that can understand sign language. The technology has many potential benefits, including increased accessibility, convenience, and inclusivity. However, some challenges need to be addressed, including accuracy and reliability issues and the need to account for regional variations in sign language [14]. As the technology continues to evolve, it will likely become even more useful and widely adopted, helping to break down barriers and promote greater inclusion and accessibility for individuals who are deaf or hard of hearing. Therefore, the main objective of this research is listed as follows.

- The optimized neural model maximizes sign language recognition accuracy by handling the lighting and background noise-related inputs.
- Minimizing the misclassification and optimization problems while recognizing the sign language from the input data from smart glove sensors.
- To address the reliability and accuracy issues for regional variations in sign language

The remaining sections of the paper are structured as follows: In Section 2, the sign language detection procedure will be presented. Section 3 analyzes the working process of Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM) based sign language detection. Section 4 evaluates the introduced POHDENM system, and the conclusion is described in Section 5.

2 Sign Language Detection Procedure

This section describes and analyzes the various researcher's works, frameworks, and ideas regarding the sign language detection process. Deriche et al., 2019 [15] introduced the Gaussian Mixer Model (GMM) to identify ASL (Arabic Sign Language) with the Leap Motion controller's corresponding gestures. The controller was utilized to remove the occlusions in the finger image. Then, the signs were analyzed to get the optimum geometric features. The derived features were processed with the help of the Linear Discriminate Analysis (LDA), Bayesian Approach (BA), and GMM approach to classifying the sign language. During the analysis, the Dempster-Shafer theory was applied to fuse the information from the controller. The fusion-based classification process improved overall sign language detection. This system used native adult signers, and 100 isolated signs were detected with 92% accuracy. The GMM-based sign detection process successfully and effectively handled the missing values.

Lee et al., 2021 [16] recommended Recurrent Neural Network (RNN) to recognize American Sign Language. The author intended to create sign language detection applications by incorporating the whack-mole game model. During the analysis, static and dynamic signs were collected and processed using the K-Nearest Neighboring (KNN) with the Long-Short Term Recurrent Model. The algorithm processed the inputs sequentially, reducing the difficulties in sign language detection. The collected images and features, such as finger angles, radius, and position distance, were processed. The extracted features were analyzed by a classifier that recognized the sign language effectively. The model was trained using 2600 samples, and 100 samples were considered the testing image. The RNN approach recognized the American sign language up to 91.82% in five-fold cross-validations.

Mittal et al., 2019 [17] developed a continuous sign language detection system using the leap motion by applying the Modified Long-Short Term Memory Network (LSTM). The system is intended to detect the sequence of interconnected gestures. The leap motion-based collected images were processed with the help of neural modeling techniques that predicted the sign language. The system used the 942 signs that were processed, and 35 sign words were detected with up to 72.3% accuracy. In addition, the modified LSTM isolated the sign words with up to 89.5% accuracy.

Aly et al., 2020 [18] introduced the Signer Independent Deep Learning approach (DeepArSLR) to recognize the Arabic sign language. The sign language images were collected and processed with the help of the DeepLabV3+ tool. The tool trained the data by extracting the hand shape and semantic information. The derived information was processed using the Convolution Self-Organizing Map (CSOM) that extracted the various features. The derived features were analyzed using deep CNN, Bidirectional LSTM, and recurrent network. These networks recognized the sign language gestures with maximum accuracy and isolated the 23 words from the users.

Al-Samarraay et al., 2022 [19] developed a sign language detection system using the Fuzzy Decision with Opinion Score Model (FDOSM). This system is intended to reduce the multi-criteria decision-making issues while classifying the sign language gestures. Images of signs were collected and evaluated using an algorithm called interactively arithmetic mean, which extracted information from the gestures themselves. The extracted features were processed using the fuzzy approach to make effective language decisions. During the analysis, Sign Language Recognition (SLR) dataset information was utilized to analyze the system's efficiency.

Katoch et al., 2022 [20] developed an Indian Sign Language detection system using the Support Vector Machine (SVM) and Convolution Neural Networks (CNN). The sign symbols were collected that were processed with the help of the Bag of Visual Words (BOVW). The BOVM approach derived the region from the A to Z alphabets and 0 to 9 digits. The background subtraction method eliminated the background details during the analysis. Then, Speeded Up Robust Features were derived and processed using the SVM and CNN. The classification approach recognized the sign language effectively and created an Interactive Graphical Interface device to make it easy to access.

Xue et al., 2022 [21] applied the Multi-Modal Perception Information Fusion process to recognize the in-hand motion. Ten human in-hand motions were initially developed with the help of the finger trajectory, electromyographic, and contact force details. Then, the motion segmentation was performed in the multi-modal data analysis platform. Empirical Mode Decomposition (EMD) was applied to decompose the images during the analysis. Then Maximum Lyapunov Exponent (MLE) was utilized to derive the non-linear features. The extracted features were analyzed using a Random Forest (RF) approach that recognized the motions with 93.72% accuracy.

Nandi et al., 2022 [22] recommended Convolution Neural Networks (CNN) with different gradient optimization techniques to create the sign language recognition system. The system intended to reduce the gap between the deaf, dumb, and normal people. The system collected the 62400 images from 26 users and processed them with the help of the CNN approach. Initially, a data augmentation process was applied to reduce the irrelevant information and rescale the details. Then batch processing and dropout layer were applied to minimize the redundant information and features from the images. Then, the diffGrad optimization technique was applied to fine-tune the network parameter, improving the sign language detection system by up to 99% compared to other methods. According to various researchers' analyses, the sign language recognition system was created with the help of machine learning, image processing, and optimization techniques. These techniques analyzed each gesture in different directions, which helped to maximize the overall detection accuracy.

Nevertheless, the approaches cannot access their sign at an affordable price and with few computational problems. The accuracy of sign language recognition is hampered by inadequate sign access. So, to record participants' behaviors, this research uses wearable technology. The patient's motion is recorded by the sensor and used to identify needs accurately. Then, the research objective is attained with the help of the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM). The POHDENM-based sign language detection process's detailed working process is explained below.

3 Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM) based Sign Language Detection

This attempt aims to enhance the accuracy of gesture recognition detection methods. The detected sign language is utilized in various applications that consist of communication aids, speech and hearing impairments related people. The existing sign language detection system has several challenges, including signing variability, limited datasets, limited vocabulary, background noise, and camera placement. Addressing these challenges will require continued research and development in computer vision, machine learning, and natural language processing. Additionally, collaborations between sign language experts and computer scientists will be crucial to improving the accuracy and usability of sign language recognition systems. Therefore, this research uses wearable devices to collect people's requirements and perform data fusion. The fused information is analyzed with the help of the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM). The introduced POHDENM-based sign language detection process has several steps: data collection, pre-processing, data fusion, classification using a neural model, training, and validation of the introduced sign language detection process. Then, the overall working process of the POHDENM-based sign language prediction process is illustrated in Figure 1.

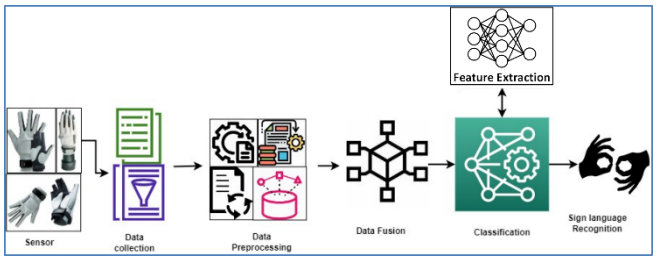


Figure 1. Overall working process of POHDENM-based sign language recognition

Figure 1 illustrates the overall working architecture of the sign language recognition process. The system uses wearable devices to collect data from people. The wearable devices are incorporated with sensors to capture gestures and hand movements. The wearable devices comprise several sensors, such as magnetometers, accelerometers, and gyroscopes, that capture hand movements and orientation in different aspects. During this process, WULALA data glove hardware space is utilized, and the configuration is presented in Table 1.

Table 1. Wulala data glove hardware space

Item	Attribute
Wireless communication	Bluetooth
Signal Latency	<5ms
DoF	11 Dof (5 fingers and six palms)
Finger curvature sensor types	5* WULALA-CS1G
Finger curvature sensor resolution	0.08
Finger curvature sensor repeatability	> 8 million
Finger curvature sensor measurement range	-30- 180
Palm IMU Sensor	1*6DoF IMU
Sensor sampling rate	Finger: 200Hz and IMU: 1000Hz
Battery duration	>8hours
Charging	USB (about 90 minutes)
Weight	70g (total with battery)

According to Table 1, the collected information is processed in the sign language recognition environment. Then, the sample glove sensor-based collected sign language gesture is depicted in Figure 2.

Figure 2. Sample sign language gesture

Various sensor-based collected information is fused to improve the overall sign language recognition process. The fusion process reduces the data availability, and limited data minimizes overall data analysis efficiency. By combining information from many sensors or data streams, sensor data fusion provides an improved understanding of the phenomena or system under observation. Fusion of data from numerous motion sensors may improve sign language identification by capturing the signer's motions and gestures more accurately. This may be useful for getting around the fact that certain sensors, although effective, may only be able to collect part of the needed data. Data fusion methods enable combining disparate data sources into a unified model that better represents the whole. Pre-processing the data from each sensor might consist of standardizing the formats, scaling to a common range, and measuring the data for consistency. Data representation is changed from 0 to 1 limit during the normalization process. This process helps to minimize the computation difficulties. After normalizing the information, data fusion is performed to create a more complete and accurate representation of the sign language gesture that can be used as input to sign language recognition. It is particularly useful when specific sensors are susceptible to noise, disturbance, or other causes of inaccuracy; this may increase the recognition state's precision and resilience.

3.1 Process of Data Fusion

Bayesian inference is a statistical approach that can be used to combine data from multiple sensors in sign language recognition systems [23]. In Bayesian inference, first construct a prior probabilistic model for the data, representing our prior information or views about the information before witnessing it. Finally, using Bayes' theorem, update the prior distribution in light of fresh observations to produce a subsequent posterior distribution for the data. In sign language recognition, Bayesian inference combines data from multiple sensors by creating a joint probability distribution for the data from all the sensors. This joint distribution represents the probability of observing the sensors' data given the underlying sign language gesture. Estimating prior probability distributions for data from each sensor provides the combined probability distribution. Our existing knowledge and hypotheses about the information provided by each sensor allow us to make educated guesses about these previous distributions. Assign a greater prior probability, for instance, to information gathered by a more reliable sensor. The combined probability distribution for all the sensors' data may be obtained by combining the prior distributions estimated for each sensor using the product rule of probability. This joint distribution represents the probability of witnessing sensor data, given the underlying sign language gesture. Estimating the most probable sign language gesture from the available data requires calculating the joint probability distribution. Given the observed data and the joint probability distribution, finding the sign language gesture with the greatest probability is possible using maximum a posteriori (MAP) estimation. Bayesian inference-based data fusion can be particularly useful in sign language recognition systems because it allows to explicitly model the uncertainty in the data from each sensor and combine this uncertainty in a principled way. This can help to improve the recognition system's robustness and accuracy, especially in cases where individual sensors may be prone to noise, interference, or other sources of error. After fusing the sensor information from multiple sensors, it has been processed with the help of the classification process. The classification process derives the features from the data and recognizes the signs with maximum recognition accuracy.

3.2 Fusion Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM) based Data Classification

The next step is Sign language recognition, which uses the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM). The POHDENM approach is a machine learning approach that combines several techniques to recognize sign language gestures with high accuracy. The smart glove sensor-based collected data is .CSV file format, which consists of information like Flexion and extension of fingers, grip strength, hand movement, and orientation information. The collected information is stored in the automatic sign language detection system database. The fused data is processed using the median filter that removes the noise from the captured sensor information. Before analysis, motion data from smart glove sensors often require pre-processing to remove noise, outliers, and other unwanted artifacts. The median filter examines every information in the sensor data, compared with the threshold value. If the collected data has any inconsistent information, like missing or improper values, that is removed with the help of the median filter. During this process, the filter determines the window size, applying the median filter to each data point in the sequence using a sliding window approach and smoothing the data using a low-pass or moving average filter. The median filter helps to remove noise and outliers from the data, while the additional smoothing helps to reduce high-frequency noise and makes the data easier to analyze. After pre-processing the data, it can be evaluated to ensure that relevant information has been preserved while unwanted artifacts have been removed and can then be used for further analysis or visualization. After removing the inconsistencies, the sensor information is analyzed to extract the features. During the feature extraction, arm motion, hand position, fingers bending angle, and relevant information are extracted using Recurrent Neural Networks (RNN).

3.2.1 Feature Extraction

Recurrent neurons are a kind of neural net that is excellent for analyzing sequence information such as time series in sign language recognition; RNN-based feature is utilized to extract relevant features from the captured time-series sensors data [24]. The RNN approach uses noise-removed information as input and normalizes the data because the data have been collected at different sampling rates. The next step is sequence segmentation, which involves segmenting the sequential data into fixed-length sequences or time windows. This enables the RNN to take individual data sequences as input rather than the complete sequence. The duration of each sequence is variable, depending on the system's needs for recognizing sign languages, and may be tweaked to strike a

perfect balance between precision and speed. Once the data is segmented into fixed-length sequences, the RNN extracts relevant features from each data sequence. The RNN learns a set of weights and biases that allow it to process the data sequence and output a set of feature vectors that summarize the important information in the sequence. The features extracted by the RNN can capture local and global temporal dependencies in the data, making them particularly powerful for gesture recognition.

Geometric and behavioral elements are continually extracted from the gesture recognition sensor data. The RNN approach is applied to extract features in a sequence and is performed for every gesture in the sensor data. The neural model derives the spatial and temporal features utilized to recognize sign language. The spatial features are derived with the help of the convolution inception model that labels every gesture, improving the overall recognition accuracy. The RNN training model is utilized to predict the feature of the gesture. The fused information is divided into training and testing. The modeling approach goes through every sensor data and extracts characteristics to boost the effectiveness of the tests. This process is repeated for every glove sensor data to extract the temporal and spatial features in different directions and positions. The extracted features are fed into the classifier to recognize the sign language.

3.2.2 Sign Language Recognition

The last step of this work is sign language recognition, done with the help of the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM). The approach utilizes a combination of several techniques to achieve optimal results. The extracted features are utilized to perform the recognition process. Some common features used for sign language recognition include hand shape, hand orientation, hand movement, and facial expressions. The extracted features are fed as input to the POHDENM. The POHDENM architecture is based on the Elman neural network, a recurrent neural network (RNN) type that can learn temporal patterns in sequential data. The network architecture is designed to have multiple layers and hidden units, which allows it to learn complex relationships between the input features and the sign language gestures. Pre-processed data from the extraction of features are sent to the multilayer network's input neurons. Characteristics such as hand form, hand orientation, hand movement, and facial emotions may be used for sign language identification. The input layer has one neuron for each input feature, fed into the next hidden layer. Hidden layers: The POHDENM system has multiple hidden layers, which allows it to learn complex relationships between the input features and the sign language gestures. Each successively concealed layer is built from several neurons interconnected with their predecessors. Signals are used by the neurons in the hidden layers to create output values by transforming the inputs. The network can learn temporal relationships using the result of the hidden units at time step t as input to the hidden neurons at time step t+1. The equations for the Elman neural network are defined in equation (1).

$$h(t) = g(W_{xh} * x_t + W_{hh} * h_{t-1} + b_h) \quad \textbf{(1)}$$

$$y_t = f(W_{hy} * h_t + b_y) \quad \textbf{(2)}$$

In equations (1 and 2), the input vector at t time is denoted as x_t , the hidden state at time t is represented as h_t , the output at time t is denoted as y_t , Specifically, W_{xh} represents the weight vector between the input and the hidden layer, W_{hh} represents the weights and biases between the hidden layer and itself, W_{hy} represents the weight matrix between the hidden and the output layer, and b_h and b_y represents the bias vectors for the hidden and output layers respectively. Elman layers are recurrent layer used in the outcome processing network that facilitates learning temporal information from sequential input. Elman layer neurons are recurrent and remember previous inputs. The network can simulate the evolving nature of sign language motions with stored information. The following step involves looping the Elman layer's output back into the layer's input. Then, it extends the Elman neural network by adding multiple hidden layers, allowing it to learn more complex representations of temporal data. The equations for the HDENM are defined below in equations (3) to (5).

$$h_t(i) = g(W_{xh}(i) * x_t + W_{hh}(i) * h_t(i-1) + b_h(i)) \quad \textbf{(3)}$$

$$h_t(N) = g(W_{xh}(N) * h_t(N-1) + b_h(N)) \quad \textbf{(4)}$$

$$y_t = f(W_{hy} * h_t(N) + b_y) \quad \textbf{(5)}$$

In the above computations, N is symbolized as the quantity of the hidden neurons, i is depicted as the hidden state measure, $W_{xh}(i)$ is the weighted sum between the input layers and the hidden layer i, $W_{hh}(i)$ is the weight array between hidden units i and itself, $b_h(i)$ is the prior probability for the hidden state i, $ht(N)$ is the outcome of the last hidden neurons, Why is the scale parameter between the last concealed layer and the final output layer, by is the error signal for the output nodes. After all input characteristics have been processed, the POHDENM system's activation function will categorize the sign language gesture. A biological neuron stands in for one gesture category at the output layer. Whether or not a non-linear stimulation is used in the output layer is determined by the nature of the issue being solved. The output unit might utilize a sigmoid-activated function for a binary classification issue or a softmax output function for a multi-class classification task. The dropout layer helps the network avoid overfitting while processing data. During learning, the dropout layer randomly eliminates a subset of the channel's neurons, pushing the surviving neurons to acquire more accurate depictions of the data input; the POHDENM system adds a batch normalization layer to standardize the results of the concealed layers. Normalizing the output helps prevent the vanishing or exploding gradient problem in deep neural networks. During the recognition, the network parameter requires updating to minimize the deviation between the outputs. This work uses the Pareto optimization technique to fine-tune the network parameters to improve the overall recognition accuracy.

3.2.3 Pareto Optimization Algorithm

Pareto optimization is a multi-objective optimization technique that seeks to find the best trade-off between conflicting objectives [25]. In the context of neural network parameter training and updating, Pareto optimization can be used to find the best set of hyperparameters that simultaneously optimize multiple performance metrics of the neural network. The optimization algorithm is used for data training and updating processes. The first step in Pareto optimization is defining the objective functions representing the different performance metrics to optimize the parameters. In neural network training, the objective functions could be the training accuracy, validation accuracy, and computational cost of the neural network. The objective function is defined as $f(x) = \{f_1(x), f_2(x), \ldots, f_k(x)\}$. Here, x is the decision variable, and the k-value represents the target quantity. The $f(x)$ is formulated to achieve the equation (6).

$$J_1 = \frac{1}{N}\sum (y - \hat{y})^2 \tag{6}$$

$$J_2 = \frac{(N_{weight} + N_{bias})}{N_{Params}} \tag{7}$$

In equation (6) J_1 is defined as the average squared deviation of (y) from the calculated value $\hat{y}$, N is denoted as the number of samples, and the number of weights in the model is denoted as N_{weight}. N_{bias} is the channel's quantity of distortions, whereas N_{Params} is the total number of design variables. According to the parameter, the model complexity (J_2) is computed using equation (7). Estimating intricacy by dividing the total number of variables by the total number of samples. Therefore, the Pareto optimization algorithm is utilized to identify the best solution while recognizing the sign language in the search space. Then, the search space is the range of values the hyperparameters can take. The search space may be continuous, discontinuous, or hybrid in a search. Some variables that might be incorporated into the search space include the learning algorithm, the number of hidden layers, the number of neurons within each layer, and the regularization parameters. The next phase is to provide potential solutions for a different set of hyperparameters. Random selection or panel search are methods often used to build a pool of potential answers from the search space. During the candidate, solution dominance is utilized in which two solutions are compared with the dominance characteristics. If the solution x_1 dominates another solution x_2, and if the solution is better than the x_2 in at least one objective function and is worse in any other objective function, then the dominant solution is defined as in equation (8) & (9):

x_1 is dominates x_2 and only if,

$$f_i(x_1) \le f_i(x_2) \text{ for all } i = 1, 2, ..., k \tag{8}$$

$$f_j(x_1) < f_j(x_2) \text{ for at least one } j = 1, 2, ..., k \tag{9}$$

Pareto optimality is when no alternative solution in the search space outperforms a given solution, denoted by x. Pareto optimality may be stated mathematically: a response x is optimum if and only if there is no alternative solution x' in the search area such that x' dominates x. The Pareto front collects all the best possible results from a search. It represents the best trade-off between the conflicting objective functions defined in equation (10).

$$PF = \{x \mid x \text{ is Pareto optimal}\} \tag{10}$$

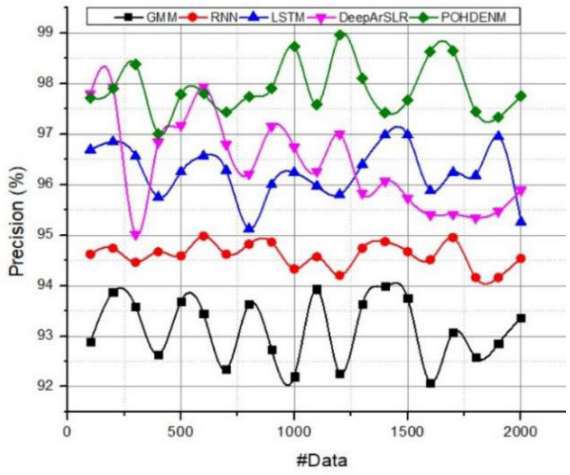
The neural network is trained using the objective functions for each candidate solution. The hyperparameters of the neural network are updated during training using an optimization algorithm such as stochastic gradient descent. The effectiveness of the neural network on each performance indicator is determined once training has been completed for each potential solution by evaluating the objective functions. After gathering performance data, a multi-objective optimization problem is built. The next step is applying a Pareto optimization method to find the optimal combination of hyperparameters for maximizing all relevant performance indicators. The Pareto optimization technique takes an evolutionary approach, merging and altering previous solutions to develop new candidates for optimization. The goal of the method is to locate a collection of non-dominated solutions that can't be made better in any one metric without negatively impacting the performance of the others. The optimal hyperparameters are chosen from among the non-dominated options at the end of the Pareto optimization process. The optimal approach may be determined by weighing the benefits and drawbacks of various performance measurements. Therefore, the POHDENM is an effective recurrent neural network model for sign language recognition since it combines the strengths of the HDENM architecture with Pareto optimization. By optimizing accuracy and complexity simultaneously, the POHDENM can achieve better generalization performance and be more efficient regarding computational resources.

4 Evaluation of POHDENM System, Results and Discussions

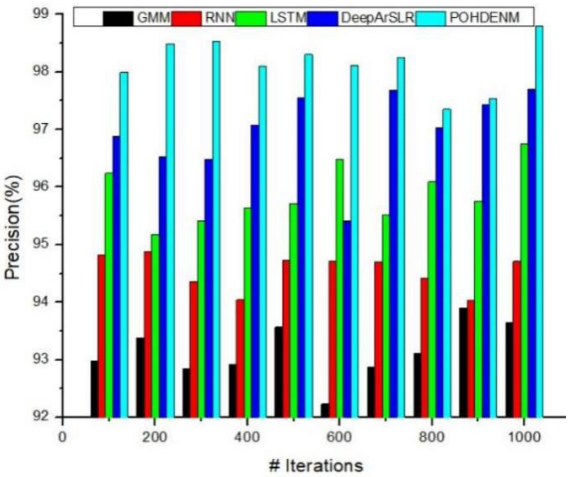
This section discusses the efficiency of the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM) based sign language recognition system. The system uses different sensors to capture people's information. This work uses the Table 1 setup information while collecting the details. According to the setup, the smart glove sensor detects the motion of the fingers, finger bending angle, and palm attitude angles and outputs data showing the position of each finger in real-time. For example, the data output might show that the index finger is extended at a 45-degree angle while the middle finger is slightly curled. The glove sensors continuously capture the information and perform a fusing process. After fusing the information sensor-based collected data, noise has been eliminated to improve the overall sign language recognition accuracy. The collected details are

divided into 80% of training data and 20% of testing data. The comprehensive program extracts the temporal and spatial properties using the RNN network feature. An improved neural network that corrects sign language recognition is then applied to the generated characteristics. Reliability, accuracy, recall, and F1-score are the only criteria for assessing the new POHDENM system's efficacy. Gaussian Mixer Model (GMM) [15], Recurrent Neural Network (RNN) [16], and Modified Long-Short Term Memory Network (MLSTM) [17] are some of the existing systems against which the findings are evaluated, and Signer Independent Deep Learning approach (DeepArSLR) [18]. These methods effectively process the sign language with minimum computation difficulties. The effective results and input handling procedures are the main reasons to select these methods to compare with the POHDENM.

4.1 Precision Analysis



(a) Precision vs. glove sensor data



(b) Precision vs. number of iterations

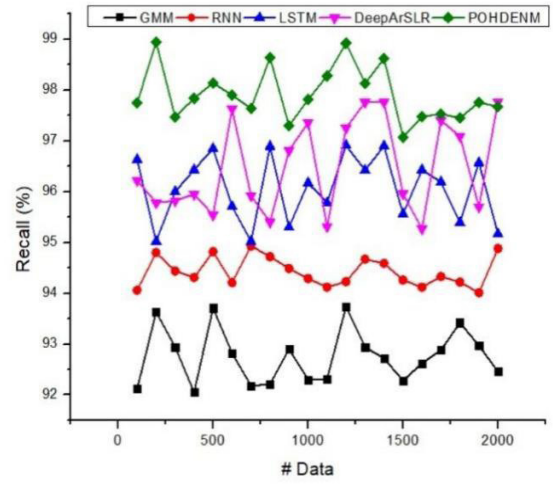
Figure 3. Precision analysis

Pareto optimization allowed for simultaneous optimization of precision and complexity, resulting in a more efficient and precise model. The temporal dynamics of sign language are also captured by the deep Elman recurrent layer, which improves generalization performance and accuracy. POHDENM's combination of these features has enabled it to achieve its superiority over previous methods on precision measures such as the GMM, RNN, LSTM, and DeepArSLR. The POHDENM can capture the temporal dynamics of sign language because of its use of Pareto optimization and a deep Elman recurrent layer, which enable it to maximize precision and complexity. Figure 3(a) illustrates the precision analysis of collected sensor data through gloves and indicates the various positions of finger information with various angles, generating nearly 2000 information pieces related to sign language motions. Figure 3(b) depicts the precision in terms of (%) with the number of iterations up to 1000. The results include improved generalization performance and quicker convergence compared to other approaches that need more training data and may be subject to overfitting. In addition to its robust precision and generalization capabilities, the POHDENM also has several additional benefits. It is practical for use in the real world since it can interpret various sign language motions. As an added bonus, it can be learned through data from several signers and used with any sign language. In addition, it is computationally efficient, allowing sign language sensor data to be processed in real time. With the POHDENM, the deaf and hard-of-hearing populations may have better access to information and communication (see Figure 3).

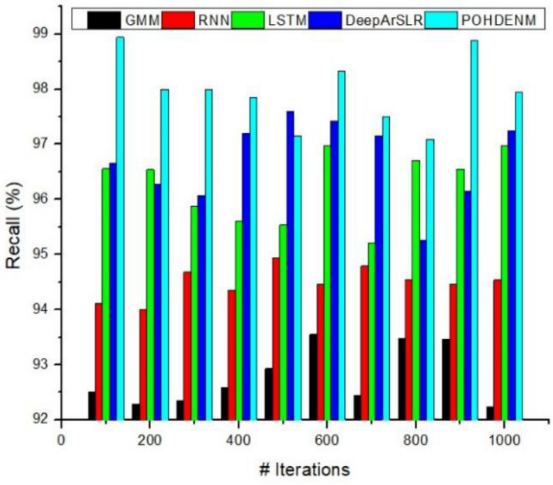
4.2 Recall Analysis

A model's ability to correctly identify all positive occurrences in a dataset is quantified by its recall. Figure 4 shows a graphical representation of the acquired recall value. Figure 4(a) shows the recall analysis of glove sensor data indicates dynamic positions of sign language motions up to 2000 numbers. Figure 4(b) depicts the recall ratio (%) with the number of iterations up to 1000. High recall indicates that the model can accurately identify many signs in sign language recognition. Due to its well-tuned architecture and parameters, the POHDENM model achieved excellent recall performance. Through the use of Pareto optimization, both the efficiency and accuracy of the model may be maximized at the same time, yielding better results. Note that when the data batch was too large, the model exhibited local convergence, leading to a decrease in recall from around 98% to 97%.

The deep Elman recurrent layer captured the temporal dynamics of sign language, enhancing the model's ability to detect signs in real-time. The POHDENM outperformed other approaches regarding recall performance, which is essential for precise sign language recognition, including GMM, RNN, LSTM, and DeepArSLR. The POHDENM is a highly accurate and efficient sign language recognition model because of its mix of optimal design, parameter adjustment, and deep recurrent layers.



(a) Recall vs. glove sensor data



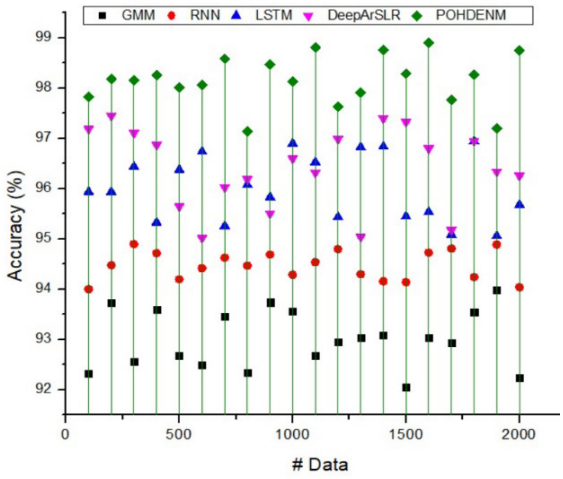
(b) Recall vs. number of iterations

Figure 4. Recall analysis

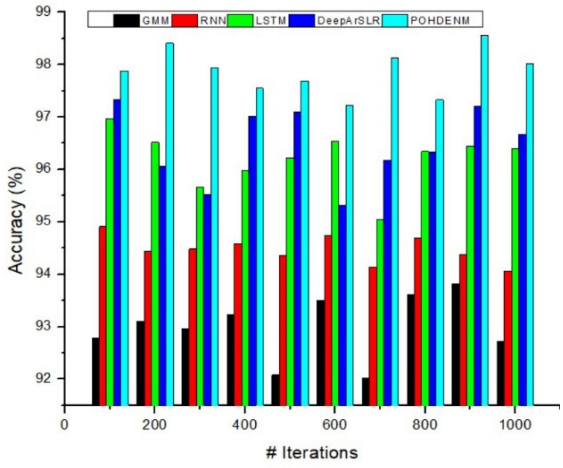
4.3 Accuracy Analysis

POHDENM can achieve high accuracy in sign language recognition by using gesture features and training processes. The experiment indicates that compared with other models, the accuracy (97.86%) is at least 2% higher. Directions, hand form, and hand motion are some examples of the types of gesture features that may be retrieved from glove sensor data and utilized in a model. These features are then pre-processed to eliminate variability and ensure consistency in the data.

The POHDENM is then trained using both supervised and unsupervised methods. The difference between supervised and unsupervised learning is that data is labeled with the relevant sign language gesture in the earlier case, while in the latter, data is clustered to discover patterns and structures. By using backpropagation through time (BPTT) during training, the model can acquire the temporal dependencies of the gestures. The model's parameters are optimized during training using gradient descent or other optimization algorithms to minimize the training loss. The optimized model is then evaluated on a test dataset to measure its accuracy and generalization performance. Combining optimized architecture, parameter tuning, deep Elman recurrent layers, gesture features, and training steps enables the POHDENM approach to achieve high sign language recognition accuracy (Refer: Figure 5).



(a) Accuracy vs. glove sensor data



(b) Accuracy vs. number of iterations

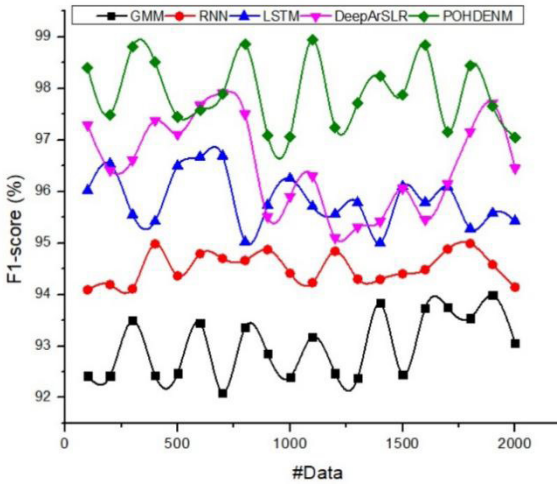
Figure 5. Accuracy analysis

4.4 F1-Score Analysis

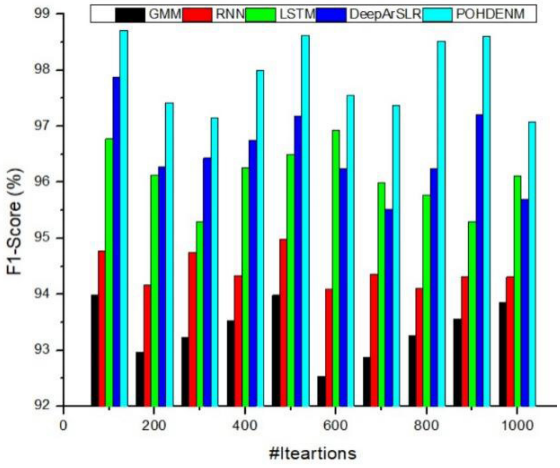
The POHDENM approach achieves high sign language recognition F1-score by combining data augmentation, regularization, ensemble method, and hyperparameter optimization. Compared to other methods, the average F1-score (98.08%) is at least 2% higher. The model uses gesture features extracted from glove sensor data and trains the Deep Elman Neural Network using backpropagation through time. The Pareto optimization algorithm is used to optimize the model's hyperparameters and achieve a balance between precision and recall.

The POHDENM approach outperforms GMM, RNN, Modified LSTM, and Signer Independent Deep Learning approach (DeepArSLR) regarding reliability, accuracy, recall, and F1-score. Using multiple evaluation metrics ensures that the model performs well on all aspects of sign language recognition. Data augmentation and regularization techniques prevent overfitting and improve the model's generalization ability to new and unseen data. Ensemble methods and hyperparameter optimization fine-tune the model and

improve its accuracy and F1 score. The POHDENM approach combines various technical points to achieve a high sign language recognition F1 score (Refer to Figure 6).



(a) F1-Score vs. glove sensor data



(b) F1-Score vs. number of iterations

Figure 6. F1-score analysis

5 Conclusion

This research discussed the Pareto Optimized Hypertuned Deep Elman Neural Model (POHDENM) based sign language recognition. Using gesture features, data augmentation, regularization, ensemble methods, hyperparameter optimization, and multiple evaluation metrics ensures that the model performs well on all aspects of sign language recognition. The POHDENM approach outperforms existing methods such as GMM, RNN, Modified LSTM, and the Signer Independent Deep Learning approach (DeepArSLR) in terms of accuracy (97.86%) and F1-score (98.08%). The model's high performance is achieved by fine-tuning its hyperparameters using the Pareto optimization algorithm, which balances precision (98.06%) and recall (98.12%). Overall, the POHDENM approach is an effective and efficient method for sign language recognition that can benefit individuals with hearing and speech impairments. However, the introduced system relies on glove sensor data as input, which may not be available in all situations. Additionally, the dataset used for training and testing the model may not represent all sign languages or signers, affecting the model's generalization capability. To address these limitations, future work could focus on developing big data sets to improve sign language detection accuracy. Future research might also look at transferring the concept to accommodate a variety of sign languages and signers.

Acknowledgements

This research was funded by 2023 Guangxi colleges and universities young and middle-aged teachers research basic ability improvement project. Project name: Research on application of gesture recognition technology based on wearable sensors (Project number: 2023KY1370).

References

[1] M. J. Hussain, A. Shaoor, S. A. Alsuhibany, Y. Y. Ghadi, T. al Shloull, A. Jalal, J. Park, Intelligent sign language recognition system for E-learning context, *Computers, Materials & Continua*, Vol. 72, No. 3, pp. 5327-5343, April, 2022. https://doi.org/10.32604/cmc.2022.025953

[2] P. K. Athira, C. J. Sruthi, A. Lijiya, A signer independent sign language recognition with co-articulation elimination from live videos: an Indian scenario, *Journal of King Saud University-Computer and Information Sciences*, Vol. 34, No. 3, pp.771-781, March, 2022. https://doi.org/10.1016/j.jksuci.2019.05.002

[3] R. El Rwelli, O. R. Shahin, A. I. Taloba, Gesture based Arabic Sign Language Recognition for Impaired People based on Convolution Neural Network, *International Journal of Advanced Computer Science and Applications*, Vol. 12, No. 12, pp. 574-582, 2021. https://dx.doi.org/10.14569/IJACSA.2021.0121273

[4] Q. Zhang, J. Z. Jing, D. Wang, R. Zhao, Wearsign: Pushing the limit of sign language translation using inertial and emg wearables, *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 6, No. 1, pp. 1-27, March, 2022. https://doi.org/10.1145/3517257

[5] D. Kothadiya, C. Bhatt, K. Sapariya, K. Patel, A. B. Gil-González, J. M. Corchado, Deepsign: Sign language detection and recognition using deep learning, *Electronics*, Vol. 11, No. 11, Article No. 1780, June, 2022. https://doi.org/10.3390/electronics11111780

[6] M. Zakariah, Y. A. Alotaibi, D. Koundal, Y. Guo, M. M. Elahi, Sign language recognition for Arabic alphabets using transfer learning technique, *Computational Intelligence and Neuroscience*, Vol. 2022, Article No. 4567989, 2022. https://doi.org/10.1155/2022/4567989

[7] M. Al-Hammadi, M. A. Bencherif, M. Alsulaiman, G. Muhammad, M. A. Mekhtiche, W. Abdul, Y. A. Alohali, T. S. Alrayes, H. Mathkour, M. Faisal, M. Algabri, H. Altaheri, T. Alfakih, H. Ghaleb, Spatial Attention-Based 3D Graph Convolutional Neural Network for Sign Language Recognition, *Sensors*, Vol. 22, No. 12, Article No. 4558, June, 2022. https://doi.org/10.3390/

s22124558

[8] K. M. H. Rawf, A. A. Mohammed, A. O. Abdulrahman, P. A. Abdalla, K. J. Ghafoor, A Comparative Study using 2D CNN and Transfer Learning to Detect and Classify Arabic-Script-Based Sign Language, *Acta Informatica Malaysia*, Vol. 7, No. 1, pp. 8-14, 2023.

[9] S. Faltaous, T. Winkler, C. Schneegass, U. Gruenefeld, S. Schneegass, Understanding Challenges and Opportunities of Technology-Supported Sign Language Learning, *Proceedings of the Augmented Humans International Conference*, Kashiwa, Chiba Japan, pp. 15-25, March, 2022. https://doi.org/10.1145/3519391.3519396

[10] B. Joksimoski, E. Zdravevski, P. Lameski, I. M. Pires, F. J. Melero, T. P. Martinez, N. M. Garcia, M. Mihajlov, I. Chorbev, V. Trajkovik, Technological solutions for sign language recognition: a scoping review of research trends, challenges, and opportunities, *IEEE Access*, Vol. 10, pp. 40979-40998, March, 2022. http://doi.org/10.1109/ACCESS.2022.3161440

[11] R.Wu, S.Seo, L.Ma, J.Bae, T. Kim, Full-fiber auxetic-interlaced yarn sensor for sign-language translation glove assisted by artificial neural network, *Nano-Micro Letters*, Vol. 14, Article No.139, July, 2022. https://doi.org/10.1007/s40820-022-00887-5

[12] D. S. Battina, L. Surya, Innovative study of an AI voice based smart Device to assist deaf people in understanding and responding to their body language, *International Journal of Creative Research Thoughts*, Vol. 9, No. 10, pp. 816-822, October, 2021.

[13] T. N. Bridgett, *American Sign Language Translation of the VCI from WISC-V*, Ph. D. Thesis, Gallaudet University, Washington, D.C., USA, 2022.

[14] M. S. Amin, S. T. H. Rizvi, M. M. Hossain, A Comparative Review on Applications of Different Sensors for Sign Language Recognition, *Journal of Imaging*, Vol. 8, No. 4, Article No. 98, April, 2022. https://doi.org/10.3390/jimaging8040098

[15] M. Deriche, S. O. Aliyu, M. Mohandes, An intelligent arabic sign language recognition system using a pair of LMCs with GMM based classification, *IEEE Sensors Journal*, Vol. 19, No. 18, pp. 8067-8078, September, 2019. https://doi.org/10.1109/JSEN.2019.2917525

[16] C. K. Lee, K. K. Ng, C. H. Chen, H. C. Lau, S. Y. Chung, T. Tsoi, American sign language recognition and training method with recurrent neural network, *Expert Systems with Applications*, Vol. 167, Article No. 114403, April, 2021. https://doi.org/10.1016/j.eswa.2020.114403

[17] A. Mittal, P. Kumar, P. P. Roy, R. Balasubramanian, B. B. Chaudhuri, A modified LSTM model for continuous sign language recognition using leap motion, *IEEE Sensors Journal*, Vol. 19, No. 16, pp. 7056-7063, April, 2019. https://doi.org/10.1109/JSEN.2019.2909837

[18] S. Aly, W. Aly, DeepArSLR: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition, *IEEE Access*, Vol. 8, pp. 83199-83212, April, 2020. https://doi.org/10.1109/ACCESS.2020.2990699

[19] M. S. Al-Samarraay, M. M. Salih, M. A. Ahmed, A. A. Zaidan, O. S. Albahri, D. Pamucar, H. A. Alsattar, A. H. Alamoodi, B. B. Zaidan, K. Dawood, A. S. Albahri, A new extension of FDOSM based on Pythagorean fuzzy environment for evaluating and benchmarking sign language recognition systems, *Neural Computing and Applications*, Vol. 34, No. 6, pp. 1-19, March, 2022. https://doi.org/10.1007/s00521-021-06683-3

[20] S. Katoch, V. Singh, U. S. Tiwary, Indian Sign Language recognition system using SURF with SVM and CNN, *Array*, Vol. 14, Article No. 100141, July, 2022. https://doi.org/10.1016/j.array.2022.100141

[21] Y. Xue, Y. Yu, K. Yin, P. Li, S. Xie, Z. Ju, Human in-hand motion recognition based on multi-modal perception information fusion, *IEEE Sensors journal*, Vol. 22, No. 7, pp. 6793-6805, April, 2022. https://doi.org/10.1109/JSEN.2022.3148992

[22] U. Nandi, A. Ghorai, M. M. Singh, C. Changdar, S. Bhakta, R. K. Pal, Indian sign language alphabet recognition system using CNN with diffGrad optimizer and stochastic pooling, *Multimedia Tools and Applications*, Vol. 82, No. 7, pp. 9627-9648, March, 2023. https://doi.org/10.1007/s11042-021-11595-4

[23] Z. Zhang, A. Nishimura, N. S. Trovão, L. L. Cherry, A. J. Holbrook, X. Ji, P. Lemey, M. A. Suchard, Accelerating Bayesian inference of dependency between mixed-type biological traits, *PLOS Computational Biology*, Vol. 19, No. 8, Article No. e1011419, August, 2023. https://doi.org/10.1371/journal.pcbi.1011419

[24] U. Farooq, M. S. Mohd Rahim, A. Abid, A multi-stack RNN-based neural machine translation model for English to Pakistan sign language translation, *Neural Computing and Applications*, Vol. 35, No. 18, pp. 13225-13238, June, 2023. https://doi.org/10.1007/s00521-023-08424-0

[25] X. Fan, S. Zhang, T. Gemmeke, Approximation of transcendental functions with guaranteed algorithmic QoS by multilayer Pareto optimization, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 28, No.12, pp. 2495-2508, December, 2020. https://doi.org/10.1109/TVLSI.2020.3012008

Biographies

Yijuan Liang received the bachelor's degree in computer science and technology. In 2008, she received the master's degree in electronics and communications engineering from Wuhan University of Technology, China. She is currently pursuing a PhD in Electrical Engineering at King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand. She is currently the deputy dean and associate professor at the School of Artificial Intelligence and Information Engineering, Guangxi Electrical Polytechnic Institute, China. Her current research interests are electrical engineering and artificial intelligence, machine learning.

Chaiyan Jettanasen received the B.Eng. and M.Eng. degrees in Electrical Engineering from Institut National des Sciences Appliquées (INSA) de Lyon, France in 2005, and the Doctoral degree in Electrical Engineering from Ecole Centrale de Lyon (ECL), France in 2008. He is presently an assistant professor at Department of Electrical Engineering, School of Engineering, King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand. His research interests include EMI/EMC in power electronic systems, ESD in electrical/ electronic system and application, conversion of electrical energy, piezoelectricity, and programming.