

## Research Article

# Deep Neural Learning Adaptive Sequential Monte Carlo for Automatic Image and Speech Recognition

Patcharin Kamsing <sup>1</sup>, Peerapong Torteeka,<sup>2</sup> Wuttichai Boonpook,<sup>3</sup> and Chunxiang Cao<sup>4,5</sup>

<sup>1</sup>*Air-Space Control, Optimization and Management Laboratory, Department of Aeronautical Engineering, International Academy of Aviation Industry, King Mongkut's Institute of Technology, Ladkrabang, Bangkok 10520, Thailand*

<sup>2</sup>*National Astronomical Research Institute of Thailand, ChiangMai 50180, Thailand*

<sup>3</sup>*Department of Geography, Faculty of Social Sciences, Srinakharinwirot University, Bangkok 10110, Thailand*

<sup>4</sup>*State Key Laboratory of Remote Sensing Science, Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100101, China*

<sup>5</sup>*University of Chinese Academy of Sciences, Beijing 100094, China*

Correspondence should be addressed to Patcharin Kamsing; [patcharin.ka@kmitl.ac.th](mailto:patcharin.ka@kmitl.ac.th)

Received 22 May 2020; Revised 18 August 2020; Accepted 22 September 2020; Published 7 October 2020

Academic Editor: Jun He

Copyright © 2020 Patcharin Kamsing et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To enhance the performance of image classification and speech recognition, the optimizer is considered an important factor for achieving high accuracy. The state-of-the-art optimizer can perform to serve in applications that may not require very high accuracy, yet the demand for high-precision image classification and speech recognition is increasing. This study implements an adaptive method for applying the particle filter technique with a gradient descent optimizer to improve model learning performance. Using a pretrained model helps reduce the computational time to deploy an image classification model and uses a simple deep convolutional neural network for speech recognition. The applied method results in a higher speech recognition accuracy score—89.693% for the test dataset—than the conventional method, which reaches 89.325%. The applied method also performs well on the image classification task, reaching an accuracy of 89.860% on the test dataset, better than the conventional method, which has an accuracy of 89.644%. Despite a slight difference in accuracy, the applied optimizer performs well in this dataset overall.

## 1. Introduction

Soft computing is available in several applications due to its usefulness in modeling and optimization. Numerous studies have focused on image and video processing with objectives such as detection and tracking. Various models have been proposed, including neural networks, deep learning, fuzzy logic, and hybrid methods [1]. However, their practical use in applications remains problematic because many applications require higher accuracy than the available models can supply. Hybrid methods that combine two or more soft computing techniques can often enhance the efficiency of image and video retrieval processes [2]. In an image context, a 3D geographical information system (GIS) data plan for

the WiMax network was integrated to optimize both the network performance and the investment costs, both of which are relevant to the required number of base stations and sectors [3]. In addition, soft computing plays an important role in GIS research [4–7]. One important aspect of implementing soft computing is the quality of the dataset. Soft computing can also be used to generate meaningful and human-interpretable big datasets by defining an interface between the numerical and categorical spaces, i.e., the data definition and the linguistic space of human reasoning [8]. Furthermore, datasets applied to investigate soft computing methods should use a benchmark dataset intended for validating various methods [1]. One example of applying soft computing for decision making was presented [9]; this is a



FIGURE 1: Example of images in the PlanesNet dataset labeled as the “plane” category.



FIGURE 2: Example of images in the PlanesNet dataset labeled as the “no-plane” category.

new method named the neurofuzzy analytical network process. The presented method works based on both fuzzy logic and an artificial neural network. Another implementation of soft computing was proposed for tunneling optimization [10]. This model analyzes the relationship between the target tunneling responses and the impact of input parameters, including both geometrical and geological factors. The proposed implementation is useful in reaching robust and low-cost soft computing solutions in the mining industry [11]. Soft computing can be applied in environmental management to predict vehicular traffic noise using data such as the volume per hour, percentage of heavy vehicles, and average speed of vehicles as inputs to neural networks or random forests [12]. Six methods are used for modeling soil water capacity parameters that are important in environmental management of targeted areas [13]. In the aviation industry, a multilayer perceptron neural network is employed to diagnose aerospace structure defects: the classical method uses signal processing and data interpretation [14]. Soft computing has also been implemented in path categorization of airplanes [15]. Soft computing can also be applied for estimating the position and orientation of spacecraft, which is useful for space technology development [16].

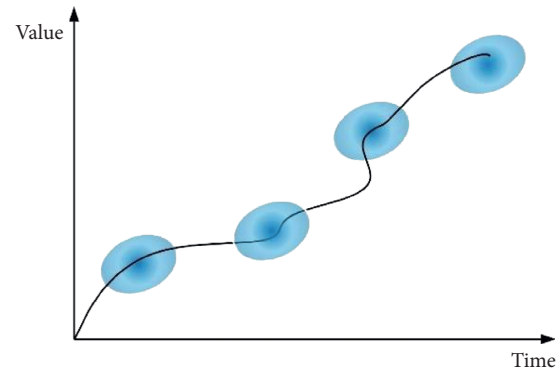


FIGURE 3: Working process of a particle filter.

Image classification and speech recognition remain demanding research topics, since they can be applied in various applications [17]. One example of an image classification method is a graph-based multiple rank regression model [18], for which the researchers presented a method that can reduce the losses in matrix data correlations that occur when an image is transformed into a vector suitable for image classification processes. An integrated recurrent neural network and a convolutional neural network (CNN), named the multipath  $x$ -D recurrent neural network (MxDRNN), has been proposed for image classification [19]. In addition, semisupervised deep neural networks implement a robust loss function to enhance image classification performance [20], and hyperspectral image classification has been widely used in many earth observation tasks, including object detection, object recognition, and surveillance. A new joint spatial-spectral hyperspectral image classification method based on differently scaled two-stream convolutional networks and spatial enhancement achieved improved classification performance [21]. Image classification for very high-resolution imagery (VHRI) is another challenging task because of the rich detail captured in the images. Many studies have focused on object-based convolutional neural networks (OCNNs) and proposed various innovations, such as integrating a multilevel context-guided classification method with an OCNN to achieve higher VHRI classification accuracy [22]. Image classification techniques have also been applied to medical applications such as breast cancer screening through histopathological imaging [23]. In addition, speech recognition research is useful for native language tasks, such as the implementation of deep neural networks for the Algerian dialect [24] and for code-switching among Frisian languages [25]. Other speech recognition research has concentrated on recognizing emotion from speech with regard to age and sex using hierarchical models [26]. A new approach for speech recognition based on the specific coding of time and frequency characteristics of speech using CNNs has been presented [27]. Visual object tracking by using an exponential quantum particle filter and mean shift optimization has been presented as another challenge for object tracking [28].

The applied method employs the particle filter technique, a state estimation technique, to optimize the gradient

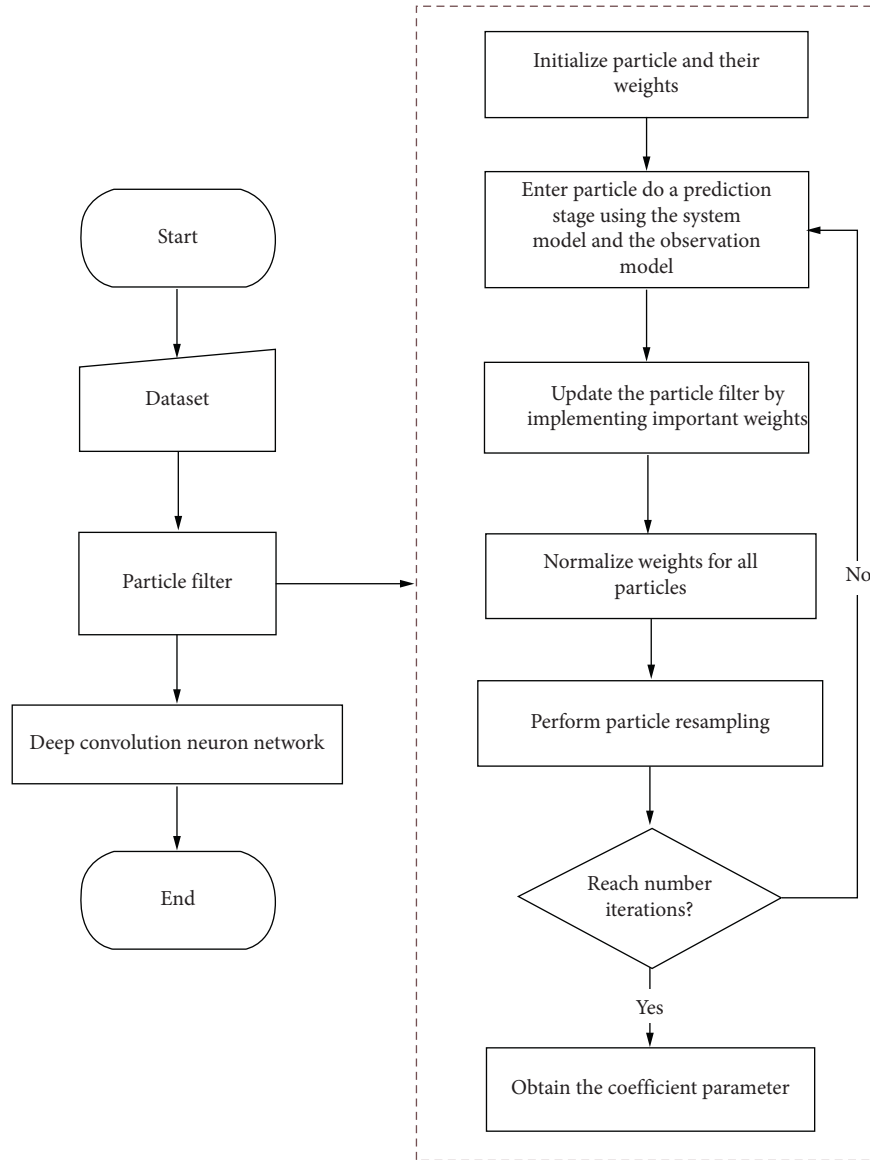


FIGURE 4: Working processes of the applied method.

TABLE 1: Results of the applied method and the gradient descent optimizer for image classification.

Method	Mean accuracy (%)	Mean cross entropy	Final test accuracy (%)
The applied method (50, 50)	<b>87.4291</b>	0.3196	89.482
The applied method (150, 100)	87.3806	0.3199	89.482
The applied method (180, 300)	87.4269	<b>0.3193</b>	<b>89.860</b>
The gradient descent method	87.4073	0.3200	89.644

descent optimizer. State estimation is often used in navigation and guidance applications and has sometimes been applied to other optimization methods. For example, for real-time traffic estimation, state estimation has been implemented using an extended Kalman filter instead of using Gaussian process regression models with respect to historical data [29]. A particle filter has also been implemented to adjust various parameters to improve image classification [30–32] and for some application such as crack

propagation filtering [33]. The gradient descent algorithm is mainly used to optimize an objective [34]. For instance, it was used to implement a demonstration of a morphing wing-tip for an aircraft to reduce low-speed drag [35]. Thermal power plants use state estimation to optimize various parameters [36]. The adaptive technique presented in this paper, which combines a particle filter with the gradient descent optimizer to adjust and improve the performance on image classification and speech recognition

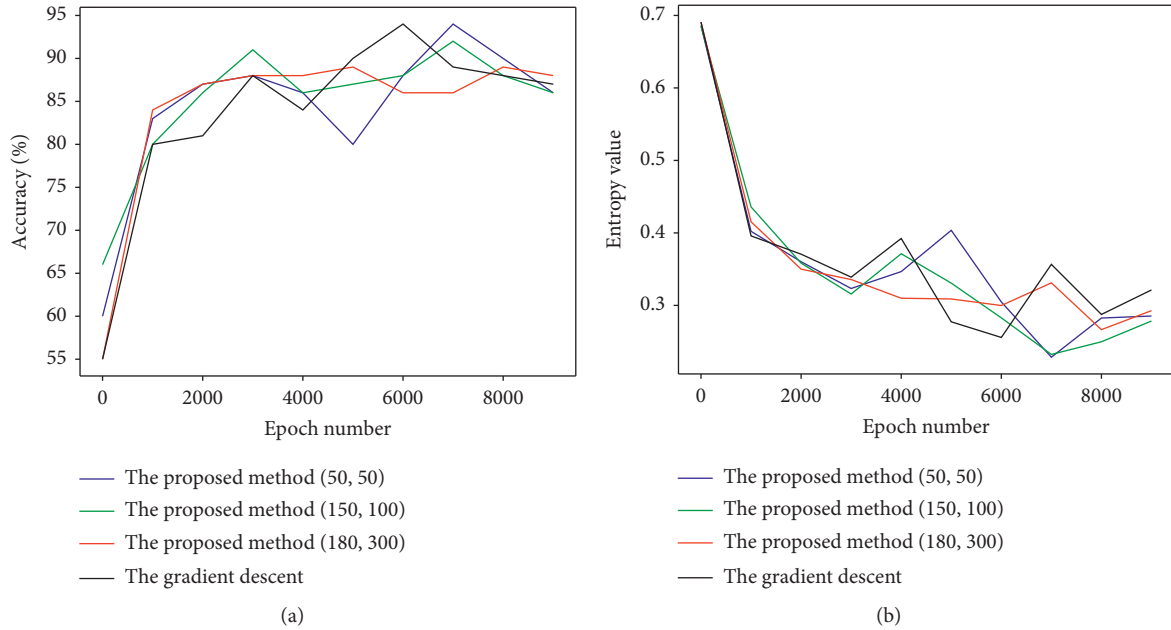


FIGURE 5: Image classification performance: (a) accuracy after each learning step; (b) cross entropy after each learning step.

tasks, is evaluated using the PlanesNet [37] and TensorFlow speech recognition challenge [38] datasets.

## 2. Materials and Methods

### 2.1. Materials

**2.1.1. PlanesNet Dataset.** Future airport designs should provide improved passenger convenience, such as reducing airplane delays or requiring less check-in time. Air traffic management, as the backbone of the aviation industry, is one factor leading airports to become more intelligent [17]. Airplane detection is a fundamental task in tracking, positioning, and predicting the positions of airplanes. PlanesNet is a medium-resolution, labeled, remote sensing image dataset that can serve as training data for training machine learning algorithms [37]. The dataset consists of  $20 \times 20$  RGB images labeled as “plane” or “no-plane” as shown in Figures 1 and 2, respectively. The “plane” images mainly consist of the wings, tail, and nose of the airplane. The images labeled “no-plane” may include land cover features such as water, vegetation, bare earth, or buildings and do not show any part of an airplane. Some example image data are presented in the following figures.

**2.1.2. Speech Commands Dataset.** Another dataset adopted in this study for testing the applied method is a public dataset for single word speech recognition, which was initially compiled for use in the TensorFlow Speech Recognition Challenge [38]. The dataset consists of audio files in which a single speaker says one word. The objective is to predict the audio files in the testing dataset, which are categorized in one of twelve categories: “silence,” “unknown,” “yes,” “no,” “up,”

“down,” “left,” “right,” “on,” “off,” “stop,” and “go.” It should be noted that the applied method is based on a CNN, which is normally applied to 2D spatial problems. In contrast, audio is inherently a one-dimensional continuous signal across time. The dataset was preprocessed into images by defining a time window into which the spoken words fit; then, the captured audio signal is converted into an image by grouping the incoming audio samples into short segments, just a few milliseconds long, and calculating the strength of the frequencies across a set of bands. Each set of frequency strengths from a segment is treated as a vector of numbers, and those vectors are arranged in time sequence to form a two-dimensional array. This array of values can then be treated such as a single-channel image called a spectrogram.

**2.2. Methods.** The applied method is implemented based on a combination of a particle filter and minibatch gradient descent optimizer processes as expressed in equation (1) with the goal of obtaining a suitable optimizer for the target dataset:

$$\theta = \theta - \eta \cdot g_{\theta} J(\theta; x^{(i:i+n)}; y^{(i:i+n)}), \quad (1)$$

where  $\theta$  is the weight,  $\eta$  is the learning rate, and  $g_{\theta}$  is a gradient of the cost function  $J(\theta)$  with respect to weight changes. Stochastic gradient descent (SGD) performs a parameter update after processing each training example  $x^{(i)}$  and label  $y^{(i)}$ , which means that the batch size is 1. The cost function in minibatch gradient descent is the average over a small data batch, which usually ranges in size between 50 and 256, but can vary depending on the application.

The applied method uses a generated particle process in combination with variables from the minibatch gradient

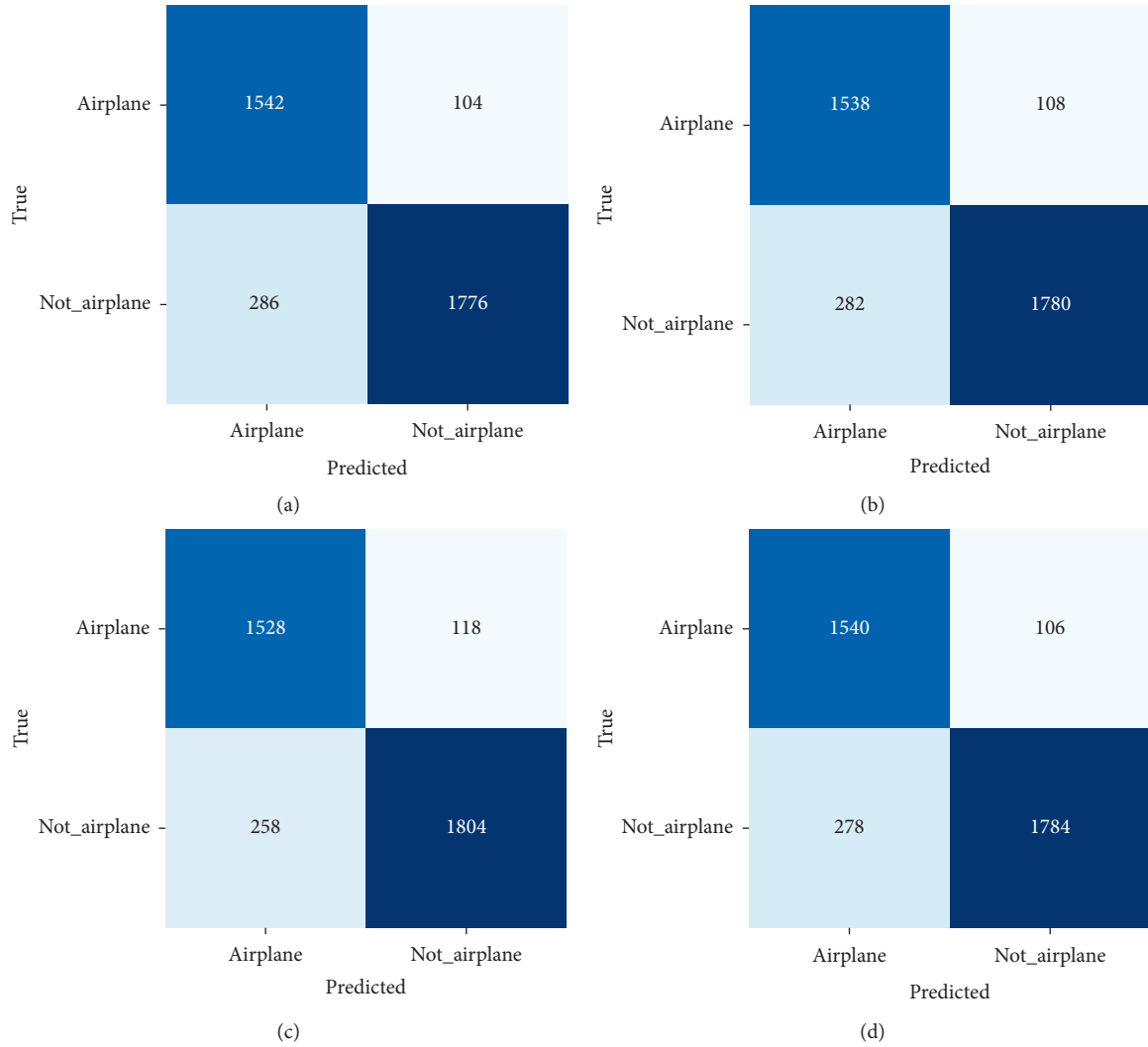


FIGURE 6: Confusion matrix: (a) the applied method (50, 50); (b) the applied method (150, 100); (c) the applied method (180, 300); (d) the gradient descent method.

TABLE 2: Results of the applied method and the gradient descent optimizer for speech recognition.

Method	Mean accuracy (%)	Mean cross entropy	Final test accuracy (%)
The applied method (50, 50)	<b>77.8163</b>	<b>0.6772</b>	<b>89.693</b>
The applied method (150, 100)	77.4286	0.6900	89.059
The applied method (180, 300)	77.2724	0.6952	89.141
The gradient descent method	77.4950	0.6853	89.325

descent optimizer. Consequently, the applied optimizer performs updates by using the computed variables instead of the conventional variables from the minibatch gradient descent optimizer. The applied method can be expressed as shown in the following equation:

$$\theta = \theta - (\eta \cdot g_{\theta} J(\theta; x^{(i:i+n)}; y^{(i:i+n)}) + K \times \text{learning rate}), \quad (2)$$

where  $K$  is an adjustment value obtained from the particle filter process.  $K$  is multiplied by the deep learning rate before being added to the second equation term of the conventional

minibatch gradient descent optimizer in equation (1). Figure 3 illustrates the working process of a particle filter. It works based on historical information from the prior stage. PF works iteratively by generating a particle, propagating it to the next time step  $t$ , and then performing an update to obtain an accurate value of the time step. A workflow of the applied method to obtain the  $K$  value is depicted in Figure 4.

The applied method shown in Figure 4 is described as follows [32]:

- (1) Initialization: at  $t = 0$ , generate  $n$  particles, and set their weights to  $\{x_0^i = \mathbf{x}_0, \pi_0^i = 1/n\}_{i=1}^n$

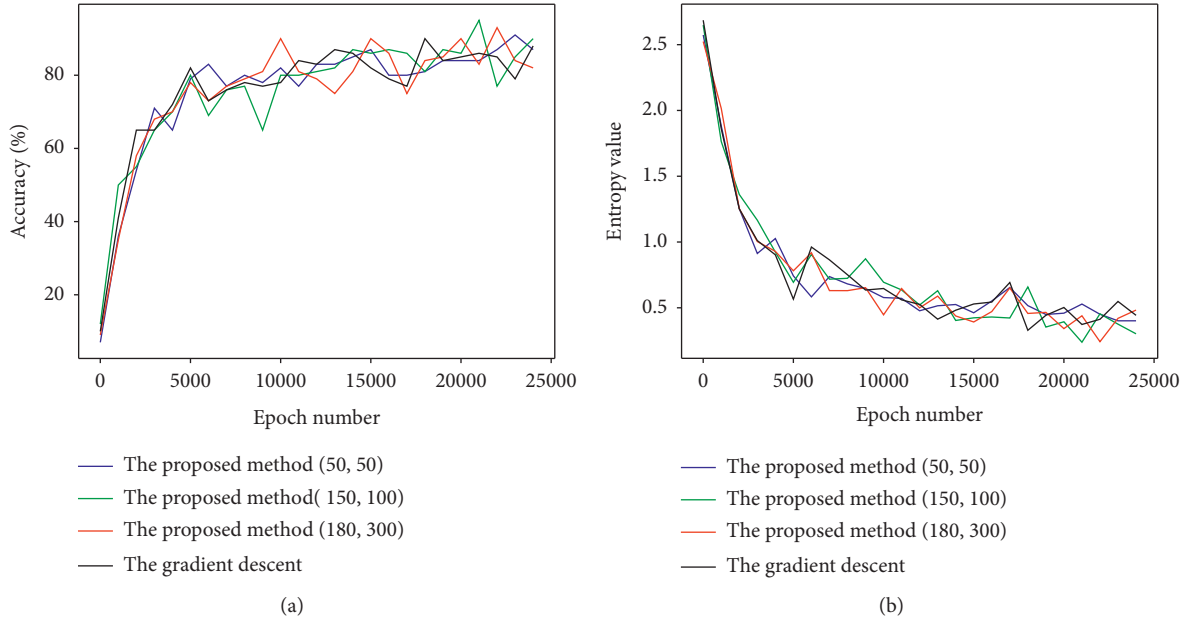


FIGURE 7: Speech recognition performance: (a) accuracy after each learning step; (b) cross entropy after each learning step.

(2) For  $t = 1, \dots, \text{end}$

- (a) Input the particle set  $\{\mathbf{x}_{t-1}^i, \pi_{t-1}^i\}_{i=1}^n$  to obtain the output  $\hat{\mathbf{x}}_{t|t-1}^i$  by using the system model equation, which is determined by the particle plus a value from the Gaussian process with zero mean and whose variance is equal to the deep learning rate
- (b) Predict the observation value  $\hat{z}_{t|t-1}^i$  by using  $\hat{\mathbf{x}}_{t|t-1}^i$  with the measurement value assigned based on the mean of the prior iteration
- (c) Update the particle weight based on the observation vector  $\mathbf{z}_t$  by  $\mathbf{h}_t(\cdot)$  or the observation model, which is set to 1. Calculate the importance weight using  $\pi_t^i = p(\mathbf{z}_t | \hat{\mathbf{x}}_{t|t-1}^i)$ ,  $i = 1, \dots, n$ .
- (d) Normalize the weights according to  $\tilde{\pi}_t^i = \pi_t^i / \sum_{j=1}^n \pi_t^j$ . Particle rejection or retention depends on the weight ( $\pi_t^i$ ) and multinomial resampling, which is determined by the resampling algorithm.

### 3. Results and Discussion

**3.1. Image Classification Result.** This experiment uses the inception\_v3 model, which is a pretrained model intended for image classification applications. The PlanesNet dataset deployed in this experiment has a total of 18,085 images divided into two classes (7,995 “plane” images and 10,090 “no-plane” images). The data are divided into a training set with 14,377 images and a testing set with 3,708 images. The training batch size is set to 100, the learning rate is 0.001, and the deep learning computation requires 10,000 epochs.

The results of the applied method are compared with those of the conventional gradient descent optimizer. The applied method shows three cases (with different numbers of

particles and particle filter iterations in parentheses). The results of the applied method and those of the gradient descent optimizer for image classification in Table 1 reveal that iterations using the applied method (180, 300) achieve the best performance as measured by the mean cross entropy in every iteration (0.3193) and by the final test accuracy (89.860%). The applied method (50, 50) achieves the best performance with regard to mean accuracy (87.4291%), which is calculated after every iteration.

The accuracy and cross entropy after each deep learning iteration are shown in Figure 5. The graphs do not clearly express different model efficiencies because the performance improves only slightly as shown in Table 1. However, both accuracy and cross entropy (Figures 5(a) and 5(b), respectively) present the values of the corresponding trends for the applied method and the conventional method.

The confusion matrices for all cases are shown in Figure 6, clearly revealing that the applied method with 180 particles and 300 particle filter iterations achieves the best prediction result for the category of “no-plane;” however, it shows poor prediction results for the “plane” category. The confusion matrices for the other three results in Figures 6(a), 6(b), and 6(d) show no large differences in either the “plane” or the “no-plane” categories. These results imply that differences in the number of particles and the number of iterations in the particle filter affect the overall performance of the applied method. Thus, each application should select the most appropriate model based on user requirements and acceptable model accuracy.

**3.2. Speech Recognition Result.** A simple deep CNN is used in this experiment to generate a model for the audio file. The models are trained for 25,000 epochs with a batch size of 100 and a learning rate of 0.001. The audio files include 105,829 individual files: 100,939 in the training dataset and 4,890 in

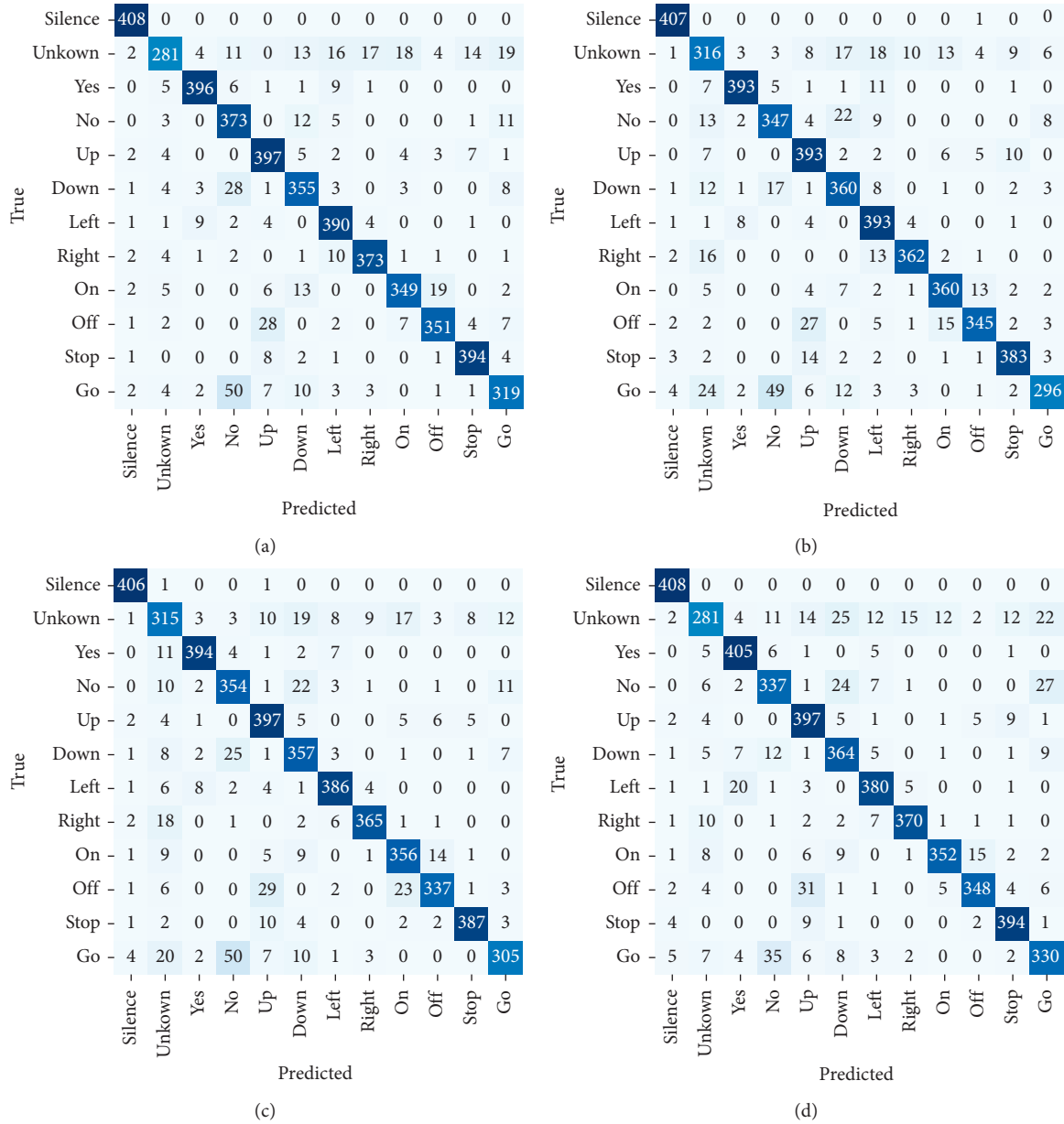


FIGURE 8: Confusion matrices: (a) the applied method (50, 50); (b) the applied method (150, 100); (c) the applied method (180, 300); (d) the gradient descent method.

the testing dataset. Similar to the image classification experiment, this experiment compares the results of the applied method under different numbers of particles and particle filter iterations with the results from the conventional minibatch gradient descent optimizer.

The results are presented in Table 2, which show that the applied method (50, 50) achieves exceptional performance compared to the other models and obtains the best mean accuracy (77.8163%), mean cross entropy (0.6772), and final test accuracy (89.693%). The conventional minibatch gradient descent optimizer is the second best. From these results, we can conclude that the applied method configured with an appropriate number of particles and particle filter iterations can achieve a better performance than the

conventional method. The accuracy and cross entropy results after each iteration are illustrated in Figure 7, which did not reveal obvious overall differences; therefore, the improvements are listed in Table 2. Confusion matrices are presented in Figure 8. The applied method (50, 50) shows exceptional performance on the “no,” “right,” and “off” classes. However, the conventional method achieves the best performance on the “yes,” “down,” and “go” classes. The other two versions of the applied method achieve a good performance on the “unknown” class. Finally, the applied method (150, 100) achieves the best results on the “left” and “on” classes.

The overall results of the speech recognition experiment show that the applied method performs better than the

conventional method in terms of both accuracy and cross entropy. However, the confusion matrix results should be considered in detail before selecting the most suitable model for a given application.

The overall performance of using the applied method with image classification and speech recognition provides better accuracy. However, confusion matrices for both image classification and speech recognition illustrate some failure cases that remain a challenging task for further research. This is a very important consideration for some applications that require high precision of image classification, such as in the health care industry, or high precision of speech recognition, such as in rescue processes. Therefore, the applied method in this experiment, based on state estimation and a well-known optimizer, is helpful to slightly improve performance in both applications. To apply this method in practical applications, more consideration of acceptable cases and failure cases using confusion matrices is required to reach optimal performance.

#### 4. Conclusions

The goal of this study was to use the particle filter technique to optimize a variable in a gradient descent optimizer. The applied method was validated by applying it to two different types of public datasets: the PlanesNet dataset (for image classification) and the Speech Commands dataset (for speech recognition). Moreover, three variations of the applied method that use different numbers of particles and different numbers of iterations were tested on those two datasets: the three model variations used 50 particles and 50 particle filter iterations, 150 particles and 100 particle filter iterations, and 180 particles and 300 particle filter iterations, respectively. The overall results show that the applied method achieves exceptional performances on both datasets, obtaining higher accuracy and lower cross entropy than the conventional method. The experiments also showed that the number of particles and the number of iterations used in the particle filter process affect the model's overall performance. Therefore, to build a high-accuracy model, appropriate parameter values should be selected for the particle filter process in the applied method according to each application. A confusion matrix can be used as an assistive tool to select the most suitable model for a given application.

#### Data Availability

The data used to support this study are available at PlanesNet Dataset and Speech Commands Dataset.

#### Conflicts of Interest

The authors declare that they have no conflicts of interest.

#### Acknowledgments

The authors thank the staff of the International Academy of Aviation Industry, King Mongkut's Institute of Technology Ladkrabang, for their contributions to this article. This research was funded by Academic Melting Pot, the KMITL

Research Fund, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.

#### References

- [1] M. Kaushal, B. S. Khehra, and A. Sharma, "Soft computing based object detection and tracking approaches: state-of-the-art survey," *Applied Soft Computing*, vol. 70, pp. 423–464, 2018.
- [2] H. Bhaumik, S. Bhattacharyya, M. D. Nath, and S. Chakraborty, "Hybrid soft computing approaches to content based video retrieval: a brief review," *Applied Soft Computing*, vol. 46, pp. 1008–1029, 2016.
- [3] L. H. Son and P. H. Thong, "Soft computing methods for WiMax network planning on 3D geographical information systems," *Journal of Computer and System Sciences*, vol. 83, no. 1, pp. 159–179, 2017.
- [4] A. U. Islam, M. J. Khan, K. Khurshid, and F. Shafait, "Hyperspectral image analysis for writer identification using deep learning," in *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–7, IEEE, Perth, Australia, December 2019.
- [5] M. J. Khan, H. S. Khan, A. Yousaf, K. Khurshid, and A. Abbas, "Modern trends in hyperspectral image analysis: a review," *IEEE Access*, vol. 6, pp. 14118–14129, 2018.
- [6] M. J. Khan, K. Khurshid, and F. Shafait, "A spatio-spectral hybrid convolutional architecture for hyperspectral document authentication," in *Proceedings of the International Conference on Document Analysis and Recognition (ICDAR)*, pp. 1097–1102, IEEE, Sydney, Australia, September 2019.
- [7] M. J. Khan, A. Yousaf, K. Khurshid, A. Abbas, and F. Shafait, "Automated forgery detection in multispectral document images using fuzzy clustering," in *Proceedings of the 13th IAPR International Workshop on Document Analysis Systems (DAS)*, IEEE, Vienna, Austria, pp. 393–398, April 2018.
- [8] G. Smits, O. Pivert, R. R. Yager, and P. Nerzic, "A soft computing approach to big data summarization," *Fuzzy Sets and Systems*, vol. 348, pp. 4–20, 2018.
- [9] D. A. Carrera, R. V. Mayorga, and W. Peng, "A soft computing approach for group decision making: a supply chain management application," *Applied Soft Computing*, vol. 91, Article ID 106201, 2020.
- [10] S. Isam and Z. Wengang, *Use of Soft Computing Techniques for Tunneling Optimization of Tunnel Boring Machines*, Underground Space, Upper Saddle River, NJ, USA, 2020.
- [11] C. K. Arthur, V. A. Temeng, and Y. Y. Ziggah, "Soft computing-based technique as a predictive tool to estimate blast-induced ground vibration," *Journal of Sustainable Mining*, vol. 18, no. 4, pp. 287–296, 2019.
- [12] D. Singh, S. P. Nigam, V. P. Agrawal, and M. Kumar, "Vehicular traffic noise prediction using soft computing approach," *Journal of Environmental Management*, vol. 183, pp. 59–66, 2016.
- [13] J. Shiri, A. Keshavarzi, O. Kisi, and S. Karimi, "Using soil easily measured parameters for estimating soil water capacity: soft computing approaches," *Computers and Electronics in Agriculture*, vol. 141, pp. 327–339, 2017.
- [14] G. D'Angelo and S. Rampone, "Feature extraction and soft computing methods for aerospace structure defect classification," *Measurement*, vol. 85, pp. 192–209, 2016.
- [15] P. Kamsing, P. Torteeka, S. Yooyen et al., "Aircraft trajectory recognition via statistical analysis clustering for Suvarnabhumi international airport," in *Proceedings of the 22nd International Conference on Advanced Communication*



- Technology (ICACT)*, pp. 290–297, IEEE, Phoenix Park, Republic of Korea, February 2020.
- [16] T. Phisannupawong, P. Kamsing, P. Tortceka, and S. Yooyen, “Vision-based attitude estimation for spacecraft docking operation through deep learning algorithm,” in *Proceedings of the 22nd International Conference on Advanced Communication Technology (ICACT)*, pp. 280–284, IEEE, Phoenix Park, Republic of Korea, February 2020.
- [17] P. Kamsing, P. Tortceka, and S. Yooyen, “Deep convolutional neural networks for plane identification on satellite imagery by exploiting transfer learning with a different optimizer,” in *Proceedings of the IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 9788–9791, IEEE, Yokohama, Japan, August 2019.
- [18] H. Yuan, J. Li, L. L. Lai, and Y. Y. Tang, “Graph-based multiple rank regression for image classification,” *Neurocomputing*, vol. 315, pp. 394–404, 2018.
- [19] R. Gao, Y. Huo, S. Bao et al., “Multi-path x-D recurrent neural networks for collaborative image classification,” *Neurocomputing*, vol. 397, pp. 48–59, 2020.
- [20] H. Cevikalp, B. Benligiray, and O. N. Gerek, “Semi-supervised robust deep neural networks for multi-label image classification,” *Pattern Recognition*, vol. 100, Article ID 107164, 2020.
- [21] M. Han, R. Cong, X. Li, H. Fu, and J. Lei, “Joint spatial-spectral hyperspectral image classification based on convolutional neural network,” *Pattern Recognition Letters*, vol. 130, pp. 38–45, 2020.
- [22] C. Zhang, P. Yue, D. Tapete, B. Shangguan, M. Wang, and Z. Wu, “A multi-level context-guided classification method with object-based convolutional neural network for land cover classification using very high resolution remote sensing images,” *International Journal of Applied Earth Observation and Geoinformation*, vol. 88, Article ID 102086, 2020.
- [23] R. Yan, F. Ren, Z. Wang et al., “Breast cancer histopathological image classification using a hybrid deep neural network,” *Methods*, vol. 173, pp. 52–60, 2020.
- [24] M. A. Menacer, O. Mella, D. Fohr, D. Jouvet, D. Langlois, and K. Smaili, “Development of the Arabic loria automatic speech recognition system (ALASR) and its evaluation for Algerian dialect,” *Procedia Computer Science*, vol. 117, pp. 81–88, 2017.
- [25] E. Yilmaz, H. van den Heuvel, and D. van Leeuwen, “Investigating bilingual deep neural networks for automatic recognition of code-switching Frisian speech,” *Procedia Computer Science*, vol. 81, pp. 159–166, 2016.
- [26] R. Flynn and E. Jones, “Robust distributed speech recognition in noise and packet loss conditions,” *Digital Signal Processing*, vol. 20, no. 6, pp. 1559–1571, 2010.
- [27] M. Kubanek, J. Bobulski, and J. Kulawik, “A method of speech coding for speech recognition using a convolutional neural network,” *Symmetry*, vol. 11, no. 9, p. 1185, 2019.
- [28] P. P. Dash and D. Patra, “An efficient hybrid framework for visual tracking using exponential quantum particle filter and mean shift optimization,” *Multimedia Tools and Applications*, vol. 79, no. 29–30, pp. 21513–21537, 2020.
- [29] J. Jin and X. Ma, “A non-parametric Bayesian framework for traffic-state estimation at signalized intersections,” *Information Sciences*, vol. 498, pp. 21–40, 2019.
- [30] P. Insom, C. Cao, P. Boonsrimuang et al., “A support vector machine-based particle filter for improved land cover classification applied to MODIS data,” in *Proceedings of the 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 775–778, IEEE, Beijing, China, July 2016.
- [31] P. Insom, C. Chunxiang Cao, P. Boonsrimuang et al., “A support vector machine-based particle filter method for improved flooding classification,” *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 9, pp. 1943–1947, 2015.
- [32] P. Kamsing, P. Tortceka, and S. Yooyen, “An enhanced learning algorithm with a particle filter-based gradient descent optimizer method,” *Neural Computing and Applications*, vol. 32, no. 16, pp. 12789–12800, 2020.
- [33] S. F. Karimian, R. Moradi, S. Cofre-Martel, K. M. Groth, and M. Modarres, “Neural network and particle filtering: a hybrid framework for crack propagation prediction,” *Signal Processing*, vol. 2004, 2020.
- [34] A. Ratre, “Stochastic gradient descent-whale optimization algorithm-based deep convolutional neural network to crowd emotion understanding,” *The Computer Journal*, vol. 63, no. 2, pp. 267–282, 2019.
- [35] A. Koreanschi, O. Sugar Gabor, J. Acotto et al., “Optimization and design of an aircraft’s morphing wing-tip demonstrator for drag reduction at low speed, Part I— aerodynamic optimization using genetic, bee colony and gradient descent algorithms,” *Chinese Journal of Aeronautics*, vol. 30, no. 1, pp. 149–163, 2017.
- [36] A. M. Kler, P. V. Zharkov, and N. O. Epishkin, “Parametric optimization of supercritical power plants using gradient methods,” *Energy*, vol. 189, Article ID 116230, 2019.
- [37] Rhammell, “PlanesNet—planes in satellite imagery,” <https://www.kaggle.com/rhammell/planesnet/version/2>.
- [38] P. Warden, “Speech Commands: A Public Dataset for Single-word Speech Recognition,” <https://www.kaggle.com/c/tensorflow-speech-recognition-challenge>.