Pertanika J. Sci. & Technol. 33 (S2): 73 - 84 (2025)

PERTANIKA JOURNALS

SCIENCE & TECHNOLOGY

Journal homepage: http://www.pertanika.upm.edu.my/

Dynamic Hand Gesture Recognition by Hand Landmark Classification Using Long Short-Term Memory

Khawaritzmi Abdallah Ahmad[1]*, Takahiro Higashi[2] and Kaori Yoshida[2]

[1]*Department of Information Systems, Hasanuddin University, Jl. Perintis Kemerdekaan No.KM.10, Tamalanrea Indah, Kec. Tamalanrea, Kota Makassar, Sulawesi Selatan 90245 Indonesia*

[1, 2]*Graduate School of Life Science and Systems Engineering Kyushu Institute of Technology, 2-4, Hibikino, Wakamatsu, Kitakyushu City, Fukuoka 808-0196 Japan*

ABSTRACT

Hand gestures are a valuable modality for human-computer interaction, conveying information that can be used as input. Dynamic hand gestures, prevalent in real-world scenarios, necessitate considering temporal factors such as gesture initiation, termination, and frame sequence. A Long Short-Term Memory (LSTM) based recognition model was proposed to address this challenge. Data availability for dynamic hand gesture research is a significant hurdle. The dataset introduced by Fronteddu et al. provides 27 classes of dynamic hand gestures, serving as a suitable training resource. MediaPipe Hands, a computer vision framework, was leveraged to extract keypoints from each frame, capturing spatial features fed into the LSTM model. Experiments were conducted to determine the optimal dropout rate for the LSTM model. Results indicated that a dropout rate of 70% yielded the highest accuracy, achieving up to 98.53% validation accuracy and 99.71% test accuracy. These findings demonstrate the effectiveness of the proposed LSTM-based recognition model for dynamic hand gestures. Future research could explore integrating other deep learning techniques, such as attention mechanisms, to enhance the accuracy and robustness of dynamic hand gesture recognition systems. Additionally, investigating the application of the proposed model in real-world scenarios, such as virtual and augmented reality, would be valuable in assessing its practical utility.

Keywords: Classification, dynamic hand gesture, human-computer interaction, long short-term memory

ARTICLE INFO

Article history:
Received: 13 February 2023
Accepted: 18 October 2024
Published: 25 February 2025

DOI: https://doi.org/10.47836/pjst.33.S2.05

E-mail addresses:
khawaritzmi@gmail.com (Khawaritzmi Abdallah Ahmad)
higashi.takahiro231@mail.kyutech.jp (Takahiro Higashi)
kaori@brain.kyutech.ac.jp (Kaori Yoshida)
*Corresponding author

INTRODUCTION

The advancement of technology has elevated computer vision to a pivotal role in human-computer interaction, serving as a bridge between humans and machines. The increasing sophistication of computers has empowered them to augment human capabilities, streamlining various tasks.

ISSN: 0128-7680
e-ISSN: 2231-8526

Consequently, human-computer interaction has become an inseparable part of human life (Sharma & Verma, 2015). Fang et al. (2007) stated that traditionally, humans use tools such as keyboards, mice, and joysticks as media to interact with computers, which is unnatural. Natural interaction can be done with speech, body movements, handwriting, and vision interfaces and is referred to as Natural User Interface (NUI) (Camargo et al., 2021). NUI is an emerging computer interaction methodology focusing on human abilities such as touch, vision, voice, motion and higher cognitive functions such as expression, perception, and recall (Camargo et al., 2021). Hand gestures are one of the many human modalities used in human-computer interaction and contain information that can be used as input for natural human-computer interaction (Hakim et al., 2019).

There are two kinds of hand gestures: static and dynamic. A single frame or spatial dimension characterizes static hand gestures. In contrast, dynamic hand gestures encompass multiple layers of temporal information, making them more prevalent in real-world applications. To effectively recognize dynamic hand gestures, it is imperative to employ methods capable of simultaneously processing both spatial and temporal features. To create and research dynamic hand gesture recognition models, researchers also face problems with the availability of suitable data for experiments. The dataset for dynamic hand gesture recognition systems provided by Fronteddu et al. (2022) proposes a dataset of 27 dynamic hand gesture types acquired at full HD resolution from 21 subjects (Fronteddu et al., 2022). Each subject performed 27 hand gestures three times for 1,701 videos, which is the proper dataset to be a sample to train a recognition model (Fronteddu et al., 2022). A hand landmark classification method was implemented using the MediaPipe Hands framework in this study. The hand landmark classification method is a classification method that uses 21 hand landmark keypoints produced by MediaPipe Hands at x, y, and z coordinates, which are used as features that are then used in the training process using machine learning or deep learning algorithms. The hand landmark classification method allows for easy determination and adjustment of the recognized class of hand gestures for a specific purpose. This method has worked very well on static hand gesture datasets using artificial neural networks (ANN) and support vector machines (SVM) (Ahmad et al., 2022, 2023). However, for the dynamic hand gesture dataset, the resulting features must be connected and interdependent inputs on certain frames, so a more complex architecture is needed to handle sequence data. For that, a long short-term memory (LSTM) model will be used to handle the temporal features of the dynamic hand gesture dataset.

Another problem faced when training a machine learning model with multiclass classification tasks is overfitting, where the model has the ability to classify different classes in the training data but is not good when given test data. Therefore, one way is to apply the dropout method to the architecture (Srivastava et al., 2014). Five different models were utilized in this study, each with a different dropout rate: 50%, 60%, 70%, 80%, and 90%. The performance results using LSTM can achieve a validation accuracy of 96.77% and an accuracy of 100.00% in 200 epochs. This method works very well on dynamic hand

gesture datasets. The model's good performance could be a reference for its integration into industrial machinery, mining equipment, and healthcare devices. It would allow users to interact with the systems more intuitively and efficiently, especially when touching traditional input devices like keyboards or mice is impractical or hazardous.

MATERIALS AND METHODS

This methodology uses the dataset to train an LSTM-based model for hand gesture recognition. Keypoints representing hand movements are extracted using the Mediapipe Hands model, followed by data preprocessing and splitting. The LSTM model is trained with various dropouts to improve generalization, and its performance is evaluated and visualized, completing the gesture recognition pipeline, as shown in Figure 1.

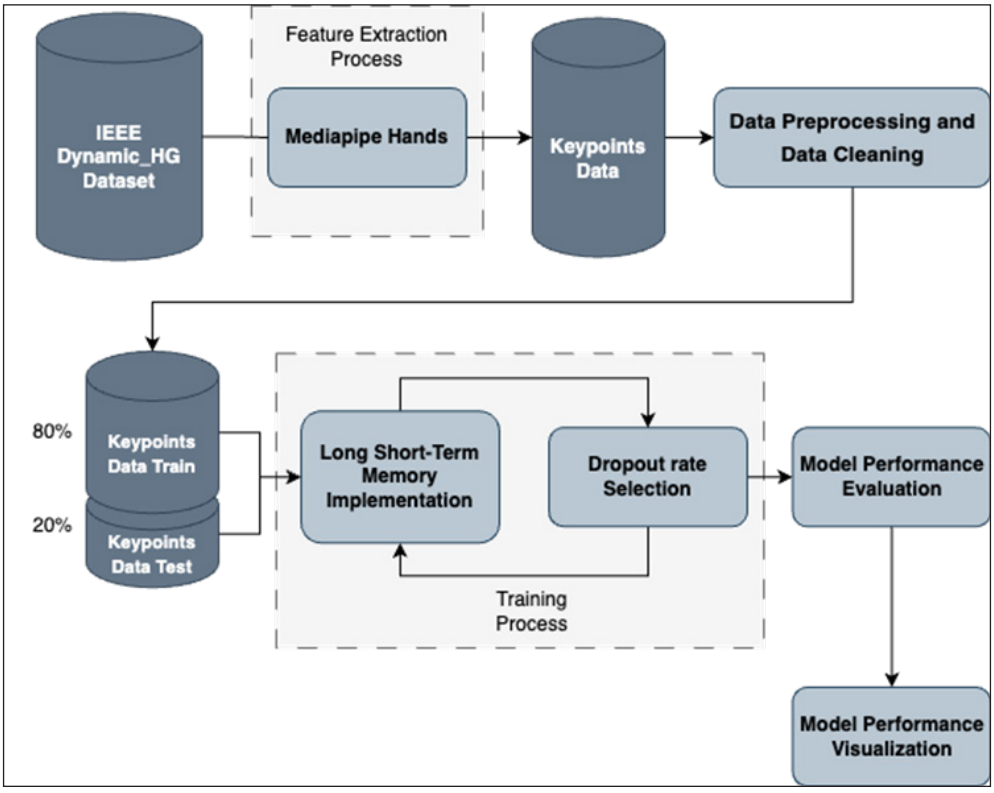


Figure 1. Method implementation flowchart

Mediapipe Hands

MediaPipe Hands is a framework used in this study to extract spatial features from hand gestures. Google LLC developed MediaPipe Hands, a framework that can be used to track and produce hand landmarks in the form of key points that indicate 21 connecting

points on the finger and palm, as shown in Figure 2. (Zhang et al., 2020). The MediaPipe Hands framework detects palms by training on three types of datasets with over 116,000 samples. These are 6000 samples from the In-the-wild dataset, 10,000 In-house collected dataset, and 100,000 samples from synthetic hand gestures (Zhang et al., 2020). A *z* value that points to the wrist depth value represents the relative depth of landmarks, and all *x, y*, and *z* data are normalized to [0.0, 1.0] (Zhang et al., 2020).

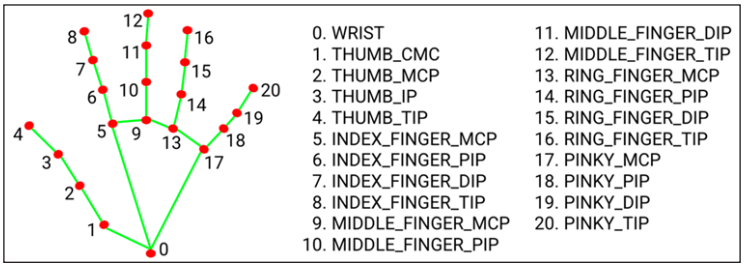


Figure 2. Mediapipe Hands keypoints (Zhang et al., 2020). 21 Keypoints represent human bone joints

Dataset

The dataset used in this research is the dataset provided by Fronteddu et al. (2022). This dataset was created by recording the subject's hand movements from the front using a camera with a full HD image resolution of 1080p (1,920*1,080 pixels) (Fronteddu et al., 2022). There were 21 subjects who demonstrated 27 classes of dynamic hand gestures labeled class_01 to class_27, as shown in Figure 3, where each subject was monitored very carefully in demonstrating hand gestures by the author to produce hand gestures consistent

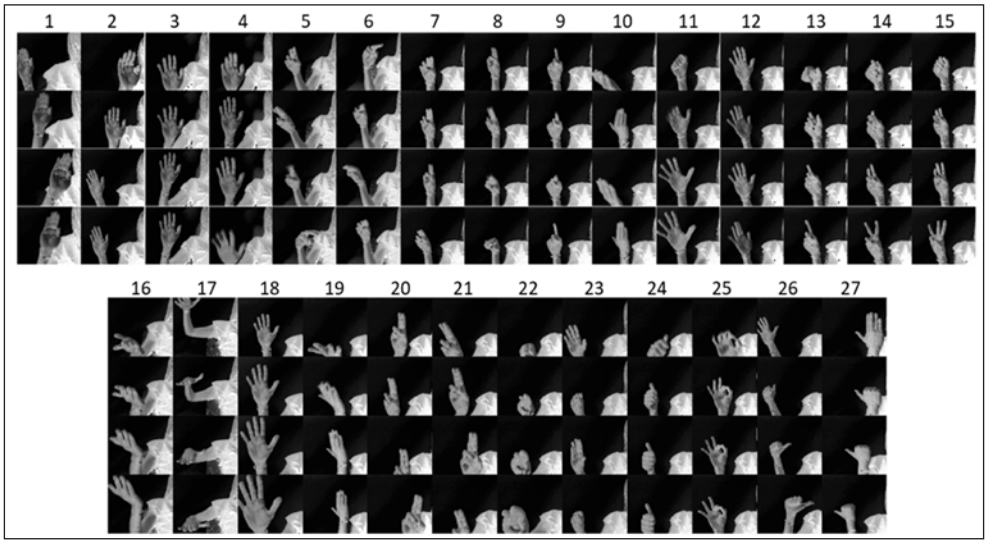


Figure 3. A representative of each 27 class on the dataset for dynamic hand gesture recognition systems (Fronteddu et al., 2022). The gesture frame comes from top to bottom

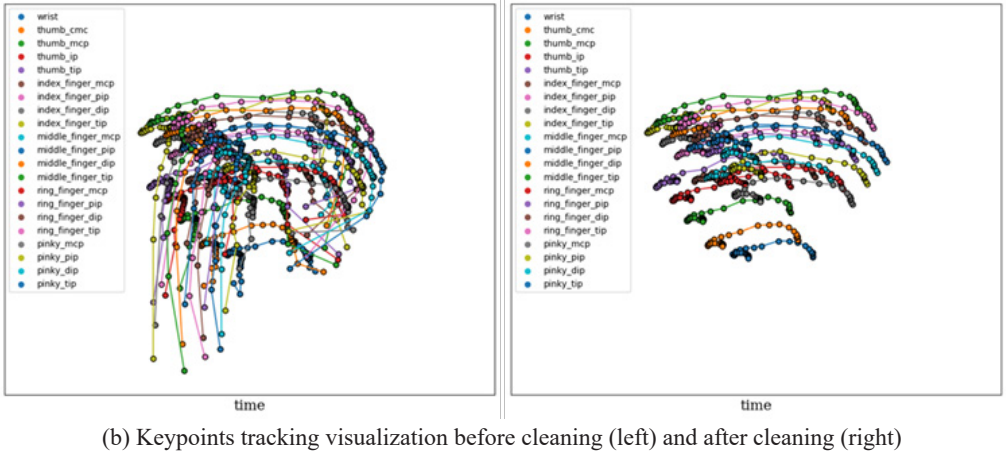
with those instructed (Fronteddu et al., 2022). Each subject made 27 hand gestures three times, with 1,701 videos and 204,120 corresponding video frames. The overall dataset video file size is 21.34 GB, with the video format being (.avi) (Fronteddu et al., 2022).

Preprocessing

Before entering the model training process, the data resulting from feature extraction enters the preprocessing process. In this process, the data has 120 frames and will produce features for each sample with dimensions of 120 rows and 63 columns. The preprocessing is to match the duration of the hand gesture made by the subject to each video sample provided in the metadata. In the metadata, two variables are provided, namely *start_frame,* which represents the frame at which the hand gesture starts to be counted as the desired hand gesture in the video.

The second variable is *end_frame*, which shows which frame the hand gesture ends at. After adjusting the duration of the hand gesture with the metadata provided, normalization is carried out on the feature vector. For each frame that occurs outside of the hand gesture's initial and final times, the entire value of the feature vector in that frame is converted to a "0" value so that all feature dimensions for each sample remain 120 rows. The preprocessing process result is visualized in Figure 4.

(a) Video dataset example from sample user1_1 on class_1 (Fronteddu et al., 2022)



(b) Keypoints tracking visualization before cleaning (left) and after cleaning (right) from sample user1_1 on class_1

Figure 4. The preparation approach involves cleaning the data, converting hand gestures from video to keypoints using Mediapipe Hands, and then fitting the hand gesture duration using metadata given by the dataset creators

Long Short-Term Memory (LSTM)

The LSTM is a recurrent neural network (RNN) commonly used for sequence modeling and analysis (Hochreiter & Schmidhuber, 1997). The LSTM model extracts temporal features from keypoint data generated from the preprocessing step. LSTM networks are crucial in deploying dynamic hand gesture recognition systems because they effectively capture temporal dependencies in sequential data. Unlike traditional models, LSTMs are designed to remember information for long periods, making them particularly suited for interpreting the continuous nature of gestures, which can vary in speed and duration.

The LSTM networks have been demonstrated to be highly effective in handling temporal features, particularly in real-time gesture recognition applications. By combining 3D Convolutional Neural Networks (3D CNNs) with LSTMs, spatial and temporal features can be extracted from video sequences, leading to significant improvements in accuracy. Studies have shown that this approach can achieve accuracy rates as high as 99% (Rehman et al., 2021; Hakim et al., 2019). With the advancement of LSTM, MediaPipe Hands is utilized to investigate the impact of skeleton-based spatial features combined with LSTM for handling temporal features in a dynamic hand gesture dataset. The aim is to develop a robust gesture recognition system that can effectively operate in dynamic environments and real-world scenarios.

Model Design, Training, and Evaluation

The dataset is represented by 120 rows representing the number of frames from one sample, 30 frames per second, with a duration of 4 seconds per sample, and 63 columns representing the dimensions of the spatial feature, which is initially 21*3 from $\vec{x} \in \mathbb{R}^{21}$, $\vec{y} \in \mathbb{R}^{21}$, and $\vec{z} \in \mathbb{R}^{21}$ and then arranged using the flattening method to become , resulting in the dataset $S = \{(\mathcal{X}_i \in \mathbb{R}^{120*21*3}, \mathcal{Y}_i \in \mathbb{R}^{27})\}_{i=1}^{n=1701}$ as shown in Figure 5. The training model employs an LSTM layer with 256 units of LSTM cells with dropout rates of 50%, 60%, 70%, 80%, and 90%. The *softmax* activation function is used in the classification stage. The total number of parameters in the model is 334,619.

The model is then constructed using loss function: categorical cross entropy and Adaptive Moment Estimation (Adam) optimizer before beginning the training phase, as shown in Table 1. The training procedure employs 200 epochs, 32 *batch_size*, and 0.1 *learning_rate*. The dataset is separated into 80% for the training process and 20% for testing data utilized in the validation step before beginning the training process. A portion of 80% is considered sufficient to train a model, with the data for each training session for each class being 50–51 samples and for training data being 12–13 samples.

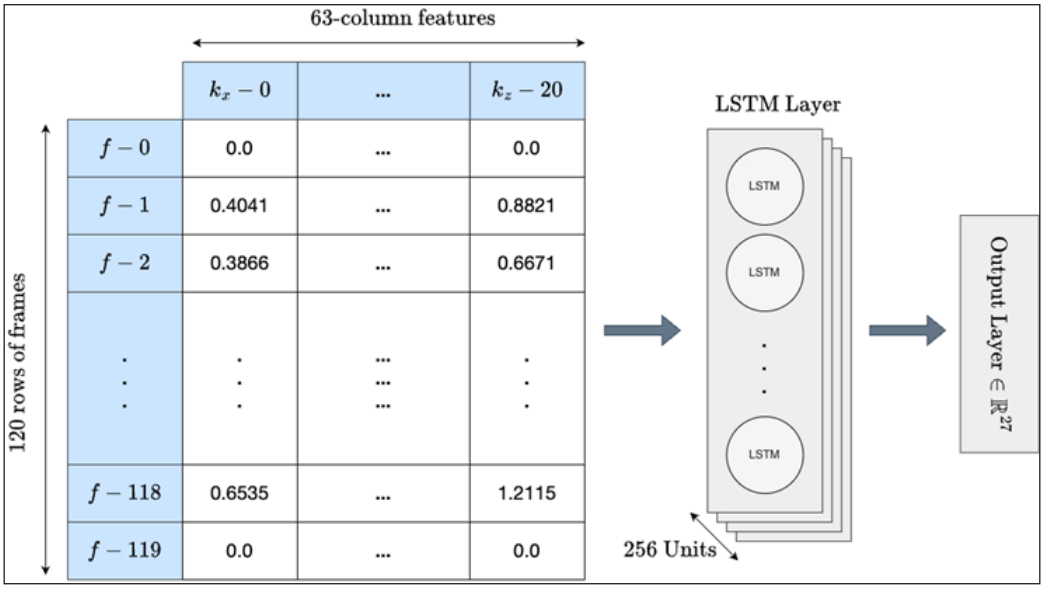


Figure 5. Model architecture using LSTM. Model input is the keypoints coordinate flattened in 1 row each frame for every sample

Table 1
Model Hyperparameters

Training Parameters	**Proposed Value**
Learning Rate	0.1
Batch Size	32
Epoch	200
Shuffle	False
Optimizer	Adam
Loss Function	Categorical Crossentropy
Activation Function	Softmax
Dropouts	50%, 60%, 70%, 80%, 90%
Total Parameters	334,619

Performance Evaluation Method

This study employs a multiclass classification methodology appropriate for datasets with more than two distinct categories. Given the dataset's 27 classes, it is well-suited for this approach. The model's performance after training was evaluated using standard metrics: accuracy, precision, recall, F1-score, and loss. Accuracy is the ratio of correct predictions to all predictions, independent of class (Equation 3). In contrast, the F1-score (Equation 4) is the harmonic average of the model's precision and recall, where precision (Equation 1)

is the proportion of how many relevant items are predicted and recall (Equation 2) is the ratio of how many relevant items are predicted. The variable values for precision, recall, accuracy, and F1-score can be derived from the confusion matrix in Table 2.

$$Precision_a = \frac{TP_a}{TP_a + \sum_X F_{a|X}} \quad [1]$$

$$Recall_a = \frac{TP_a}{TP_a + \sum_X F_{X|a}} \quad [2]$$

$$Accuracy = \frac{\sum_a TP_a}{\sum_a TP_a + \sum_a F_{X|a} + \sum_a F_{a|X}} = \frac{\sum_a TP_a}{N} \quad [3]$$

$$F1 - Score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad [4]$$

$$Categorical\ CE\ loss = -log\left(\frac{e^{s_i}}{\sum_j^C e^{s_i}}\right) \quad [5]$$

Table 2
Multiclassification confusion matrix

		PREDICTED Class		
	Classes	***a***	***b***	***c***
Actual Class	*a*	TP_a	$F_{b\|a}$	$F_{c\|a}$
	b	$F_{a\|b}$	TP_b	$F_{c\|b}$
	c	$F_{a\|c}$	$F_{b\|c}$	TP_c

Categorical cross-entropy, a function commonly used for multiclass classification tasks, was used for the loss calculation. This function is also usually called softmax loss (Equation. 5). Accuracy, loss rate, and F1-score will be evaluated to determine the quality of the built model. A model's efficacy correlates directly with its accuracy, particularly when the gap between validation and training accuracy is minimal. A high F1-score indicates the model's ability to effectively classify each class. A significant disparity between validation and training accuracy characterizes overfitting. Conversely, a lower loss rate value (closer to 0), with minimal divergence between validation and training loss, generally signifies a more effective model.

RESULTS AND DISCUSSIONS

The training and evaluation process was followed, and the results were analyzed and visualized. Five models were trained to assess the impact of different dropout rates with varying percentages (50%, 60%, 70%, 80%, and 90%). Figure 6 illustrates that the LSTM model with a 50% dropout rate consistently outperformed the others, demonstrating optimal and stable results. In contrast, models with higher dropout rates (60%, 70%, 80%, and 90%) exhibited slightly overfitting and lower validation accuracy.

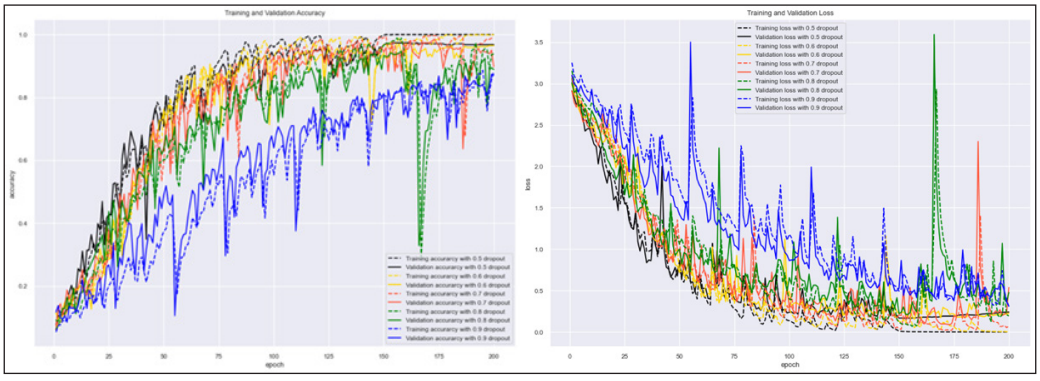


Figure 6. Model training result Accuracy Chart Comparison (left) and Loss Chart Comparison (right). This chart shows the model's accuracy and loss over 200 epochs, making it possible to evaluate the difference in accuracy and loss between the training and validation phases and whether there are any abnormalities

Table 3 demonstrates that the highest achievable accuracy was obtained with a dropout rate of 70%, reaching up to 98.53% validation accuracy and 99.70% test accuracy when the *early_stopping* function was implemented. This function terminates the training process based on specified parameters, such as validation accuracy. However, a *monitor_callback*

Table 3
Model performance comparison

Models (Dropout)	loss	accuracy	val_loss	val_accuracy	Epochs-n
MP-LSTM (0.5)	**0.00052933**	**100%**	**0.24167494**	**96.77%**	**200**
Best MP-LSTM (0.5)	**0.00447017**	**100%**	**0.16992576**	**97.36%**	**155**
MP-LSTM (0.6)	0.0050257	99.92%	0.22635585	96.48%	200
MP-LSTM (0.7)	0.0704495	98.75%	0.53832334	89.73%	200
Best MP-LSTM (0.7)	**0.01964993**	**99.70%**	**0.08934013**	**98.53%**	**173**
MP-LSTM (0.8)	0.19443941	93.82%	0.47621825	88.85%	200
MP-LSTM (0.9)	0.31087446	88.16%	0.31144518	87.09%	200

function was employed, providing checkpoints and indicating optimal training performance. This function continued the training process until this research's predetermined epoch parameter value reached 200. Consequently, the best results were selected to showcase the performance up to 200 epochs. The LSTM model's loss value in the 173rd epoch training process, using a 70% dropout rate, was 0.0196 out of 200, while the validation loss was 0.0893.

By not stopping the training process, the model's performance after the most optimal point during the training process, where the model's performance starts to decline, can also be visualized. In the LSTM model with a 50% dropout, the performance declines slowly after 155/200 epochs, which means the optimal training point is at the LSTM model generated at training epoch 155/200. Meanwhile, Figure 6 shows that the model with 70% has an optimal point at epoch 173/200, which then experiences a volatile performance decline and, at some point, is very significant. The training results of the LSTM model use a 50% dropout rate, as seen in Figure 6. Training accuracy is 100%, and validation accuracy is 97.36%. Likewise, the validation loss score value from the results obtained at the 200th epoch was 0.2417; at the 155th epoch, it was 0.1699.

The confusion matrix of the model resulting from the training process using the built LSTM model is shown in Figure 7. The result is that the number of misclassifications is 2/12

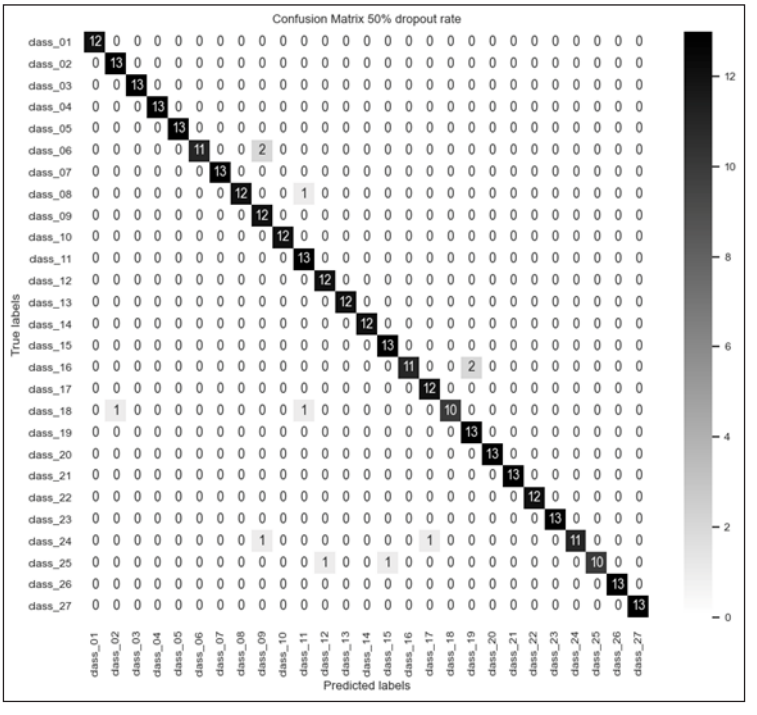


Figure 7. Confusion matrix of LSTM with 50% Dropout. This confusion matrix compares ground truth labels and prediction labels produced by the LSTM model with a 50% Dropout rate

misclassifications on two samples of test data and 2/13 misclassifications on four samples of test data. The highest percentage of misclassification is 15.384%, which is quite high, with the number of test data only reaching 12–13 samples, but in terms of performance, this is very good. The time required for the training process with data dimensions of reaches 2 minutes, 39.1 seconds, where the training process is considered very fast with 32 *batch_size* at 200 epochs.

CONCLUSION

This research was conducted to look at the performance of the hand landmark classification method, which can be obtained using MediaPipe Hands on dynamic hand gesture datasets. The dataset for dynamic hand gesture recognition systems with 27 classes and 21 different subjects, with each subject producing three different samples for each class, was used in this study. The result is that the hand landmark classification method, successfully used in previous studies using static hand gesture datasets (Ahmad et al., 2022; Ahmad et al., 2023), also works well for dynamic hand gesture datasets.

In this study, long short-term memory architecture is used to handle temporal features in the dataset, resulting in 100% accuracy training results with a validation accuracy that can achieve 97.36% for a 50% dropout rate, and 99.71% accuracy, 98.53% validation accuracy for 70% dropout rate. The misclassification rate with four classes from the dataset only reaches 15.38%, with more than that being less than that percentage. The F1-score, another performance metric used to evaluate the model, was 96.77% for the macro average F1-score at the 200th epoch, indicating the model's strong performance in classifying hand gestures with 27 different classes.

The model's demonstrated efficacy suggests its potential for widespread integration across diverse domains, including industrial machinery, mining equipment, and healthcare. This integration will contribute to developing more intuitive user interfaces, particularly in contexts where traditional input devices pose challenges due to safety or practicality concerns. Future research endeavors will involve the application of the proposed method to our self-curated dataset of natural hand gestures collected for human-robot interaction, human-computer interaction, and dynamic hand gestures in virtual reality. These studies will comprehensively evaluate the method's practical applicability by employing significantly larger datasets.

ACKNOWLEDGEMENT

This research was supported by the MEXT Scholarship from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

REFERENCES

Ahmad, K. A., Silpani, D. C., & Yoshida, K. (2022). Hand gesture recognition by hand landmark classification. *International Symposium on Affective Science and Engineering*, *2022,* Article 8. https://doi.org/10.5057/isase.2022-C000026

Ahmad, K. A., Silpani, D. C., & Yoshida, K. (2023). The impact of large sample datasets on hand gesture recognition by hand landmark classification. *International Journal of Affective Engineering*, *22*(3), 253–259. https://doi.org/10.5057/ijae.ijae-d-22-00022

Camargo, V. P. D., Balancieri, R., Teixeira, H. M., & Guerino, G. C. (2021, October 18-22). *Touchless modalities of human-computer interaction in Hospitals*. [Paper presentation]. Proceedings of the XX Brazilian Symposium on Human Factors in Computing Systems, Sao Paulo, Brazil. https://doi.org/10.1145/3472301.3484328

Fang, Y., Wang, K., Cheng, J., & Lu, H. (2007, July 2-5). *A real-time hand gesture recognition method*. [Paper presentation]. IEEE International Conference on Multimedia and Expo, Beijing, China. https://doi.org/10.1109/icme.2007.4284820

Fronteddu, G., Porcu, S., Floris, A., & Atzori, L. (2022). A dynamic hand gesture recognition dataset for human-computer interfaces. *Computer Networks*, *205*, Article 108781. https://doi.org/10.1016/j.comnet.2022.108781

Hakim, N. L., Shih, T. K., Kasthuri Arachchi, S. P., Aditya, W., Chen, Y. C., & Lin, C. Y. (2019). Dynamic hand gesture recognition using 3DCNN and LSTM with FSM context-aware model. *Sensors*, *19*(24), Article 5429. https://doi.org/10.3390/s19245429

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Sharma, R. P., & Verma, G. K. (2015). Human computer interaction using hand gesture. *Procedia Computer Science*, *54*, 721–727. https://doi.org/10.1016/j.procs.2015.06.085

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, *15*(1), 1929-1958.

Rehman, M. U., Ahmed, F., Attique Khan, M., Tariq, U., Alfouzan, F. A., Alzahrani, N. M., & Ahmad, J. (2021). Dynamic hand gesture recognition using 3D-CNN and LSTM networks. *Computers, Materials & Continua*, *70*(3), Article 4676. https://doi.org/10.32604/cmc.2022.019586

Zhang, F., Bazarevsky, V., Tkachenka, A., Sung, G., Chang, C. L., & Grundmann, M. (2020). *Media pipe hands: On-device real-time hand tracking*. arXiv. https://doi.org/10.48550/arXiv.2006.10214