SEUSL

Sri Lankan Journal of Technology Research

SLJoT
2025 Sp. Issue 147-159
ISSN-2773-6970

Real-Time Hand Gesture Recognition for Computer Command Execution Using Deep Learning and Skeleton Detection

Tharindu N. Sandaruwan[1], A. Fathima Sharfana[2*]

[1,2] Department of ICT, Faculty of Technology, South Eastern University of Sri Lanka, Sri Lanka

*Corresponding Author: sharfana.atham@seu.ac.lk || ORCID: 0000-0002-9080-2431

Received: 26-11-2024. * Accepted: 11-04-2025 * Published Online: 05-05-2025

Abstract- Accurate and effective gesture detection techniques are becoming more important as the use of gesture-based interfaces for human-computer interaction grows. In this research, we propose a deep learning-based hand gesture recognition system based on skeleton object detection and present an assistive system to operate a computer using hand gestures. In this approach, hand motions are extracted from the image using skeleton object identification and then classified using a deep learning system. This method was tested using a publicly available data set, and its performance in hand gesture recognition was demonstrated to be higher. The deep learning model presented in this study is designed for hand gesture recognition. It combines skeleton object detection and deep learning algorithms to accurately identify and interpret hand gestures. In terms of accuracy and performance evaluations, the model has proven highly effective. It achieves an impressive accuracy score of 0.9711 indicating that it correctly identifies hand gestures for nearly 97% of the data set. Additionally, the model boasts a remarkable F1 score of 0.9711, signifying its strong ability to accurately recognize positive instances while minimizing false positives and false negatives. These results imply that deep learning-based, skeleton object detection-based hand gesture recognition can enhance human-computer interaction in various contexts, including assistive technologies, gaming, virtual reality, and robotics. This study demonstrates the use of the skeleton-based deep learning model as an input paradigm to execute computer commands.

Keywords: Hand gesture recognition, computer control, skeleton-based feature extraction, human-computer interaction, deep learning

Sandaruwan, T. N., & Sharfana, A. F. (2025). Real-time hand gesture recognition for computer command execution using deep learning and skeleton detection. *Sri Lankan Journal of Technology. 6*(Special Issue). 147-159.

This work is licensed under a Creative Commons Attribution 4.0 International License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

1. Introduction

As information technology advances and computer systems become more ingrained in our lives, the development of user-friendly interfaces for human-computer interaction has become increasingly important (Sonkusare et al., 2015). Hand gesture recognition systems are a significant advancement in human-computer interaction (HCI), offering a novel way for users to interact with technology through natural and intuitive movements and enhancing user experience and efficiency while addressing the limitations of traditional input devices. It enhances the user experience and usage with its many benefits (Hrishikesh et al., 2024). It creates more engaging user interfaces and provides modalities for enhanced user interactions. It reduces the learning curve for users by mimicking human behaviors (Terreran et al., 2023). It provides an alternative means of interaction for users with physical disabilities or limitations without the need for physical manipulation. In virtual and augmented reality environments as well as robotic interactions, hand gesture recognition allows users to interact more naturally and improves the realism and effectiveness of such systems.

Hand gesture recognition is a significant area in computer vision with applications spanning from human-computer interaction to virtual reality. In the course of time, various object detection algorithms have been proposed to enhance the precision and efficiency of hand gesture recognition. The following section is divided into three subsections and discusses the traditional methods, machine, and deep learning-based methods used for hand gesture recognition.

A. Traditional Object Detection Methods

Histogram of Oriented Gradients (HOG) along with a Support Vector Machine (SVM), was one of the earliest methods for object detection (Reddy et al., 2018) (De Smedt et al., 2016). It relies on feature extraction from image gradients. While effective for simple hand gestures, HOG-based methods struggled with varying hand orientations and complex backgrounds. AdaBoost is an ensemble method that builds a strong classifier by combining weak classifiers. It was used in early hand detection systems to differentiate between hand and non-hand regions. However, its performance was limited by the quality of feature extraction and the ability to handle diverse hand gestures. Similarly, the Skin Color Detection method involves segmenting hands based on skin color in the HSV color space (Nguyen et al., 2015). While it is simple and computationally inexpensive, it is sensitive to lighting changes and variations in skin tone.

B. Machine Learning-Based Methods

Decision Trees (Song et al., 2019) and Random Forests (Luong et al., 2013) are the classifiers that use a combination of decision trees to identify hand gestures. They offer improved accuracy over traditional methods but require substantial feature engineering and can struggle with real-time performance (Sonkusare et al., 2015). Convolutional Neural Networks (CNNs) revolutionized hand gesture recognition by learning hierarchical features directly from images. Architectures like LeNet, AlexNet and later VGGNet provided substantial improvements in accuracy and robustness (Wang, Hu and Jin, 2021) (Hrishikesh et al., 2024).

C. Advanced Deep Learning Methods

Region-based CNNs (R-CNNs) improved object detection by proposing regions and applying CNNs to these regions (Wang, Cai and Zhang, 2018). This method, along with its variants Fast R-CNN and Faster R-CNN, provided significant advances in detecting and classifying hand

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

gestures by refining the region proposal network (Datta et al., 2020) (Liu, 2018). You Only Look Once (YOLO) offers real-time object detection by predicting bounding boxes and class probabilities in a single pass through the network. YOLO's efficiency makes it suitable for real-time hand gesture recognition tasks (Zhao et al., 2023) (S. Saxena et al., 2022) Hu et al., 2020; Lu, Zhang and Xie, 2020). Its efficiency and accuracy make it a popular choice for real-time hand gesture recognition applications. Mask R-CNN, extending Faster R-CNN, Mask R-CNN adds a branch for predicting object masks (Sapkota, Ahmed and Karkee, 2024). This method provides precise segmentation, which is particularly useful for detailed gesture recognition where hand boundaries need to be accurately defined.

(Nguyen et al, 2015) introduced an approach to recognize sign language based on principal component analysis (PCA) and Artificial Neural Network (ANN). The dataset for the experiment was created by using cameras in different angles to avoid overlapping of the gestures. They used colour information such as hue and saturation to highlight the skin in the image for segmentation. Then preprocessed images to detect hand and remove the other details from the image then applied PCA as a feature extractor and ANN as predictor. The system achieved 94.3% accuracy. However, this approach has limitations as skin colour can be affected by various lighting conditions and different skin tones. Also, the camera set-up in different angles makes the system more expensive.

Skeleton-based object detection offers a specialized approach that focuses on analyzing the skeletal structure of objects. This method is particularly effective for hand gesture detection, providing better performance by focusing on the relative positions of joints and limb movements rather than relying solely on image data (De Smedt, Wannous and Vandeborre, 2016). The skeleton-based methods use pose estimation to identify key points on the hand or body, such as joints and limb positions. This provides precise information about the hand's pose and movement, which is crucial for recognizing complex gestures. Unlike image-based methods, which may be affected by changes in lighting or background, skeleton-based methods focus on the relative positions of joints. This makes them more robust to environmental changes and variations in hand appearance. By focusing on key points rather than full image data, skeleton-based methods can be computationally more efficient (Bai et al., 2019). This is particularly important for real-time applications where processing speed is critical.

(De Smedt et al, 2016) leverages histograms of hand direction and wrist rotation features along with skeletal information for the prediction of static and dynamic gestures. The comparative studies on these features, gesture recognition, based on the skeletal information outperformed and it showed a promising direction towards the hand gesture recognition task using skeletal-like information. They also have achieved 83% accuracy on the Dynamic Hand Gesture dataset (DHG-14/28) for dynamic gesture recognition. The classification task has been done using an SVM linear classifier. They recommended using skeletal-based features along with depth-based information to enhance the recognition accuracy and robustness of the used algorithms. Even though the object detection frameworks can directly detect and classify gestures without the additional steps for skeleton extraction, they face challenges when hands are occluded, partially visible, or overlap with other objects (Hrishikesh et al., 2024).

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

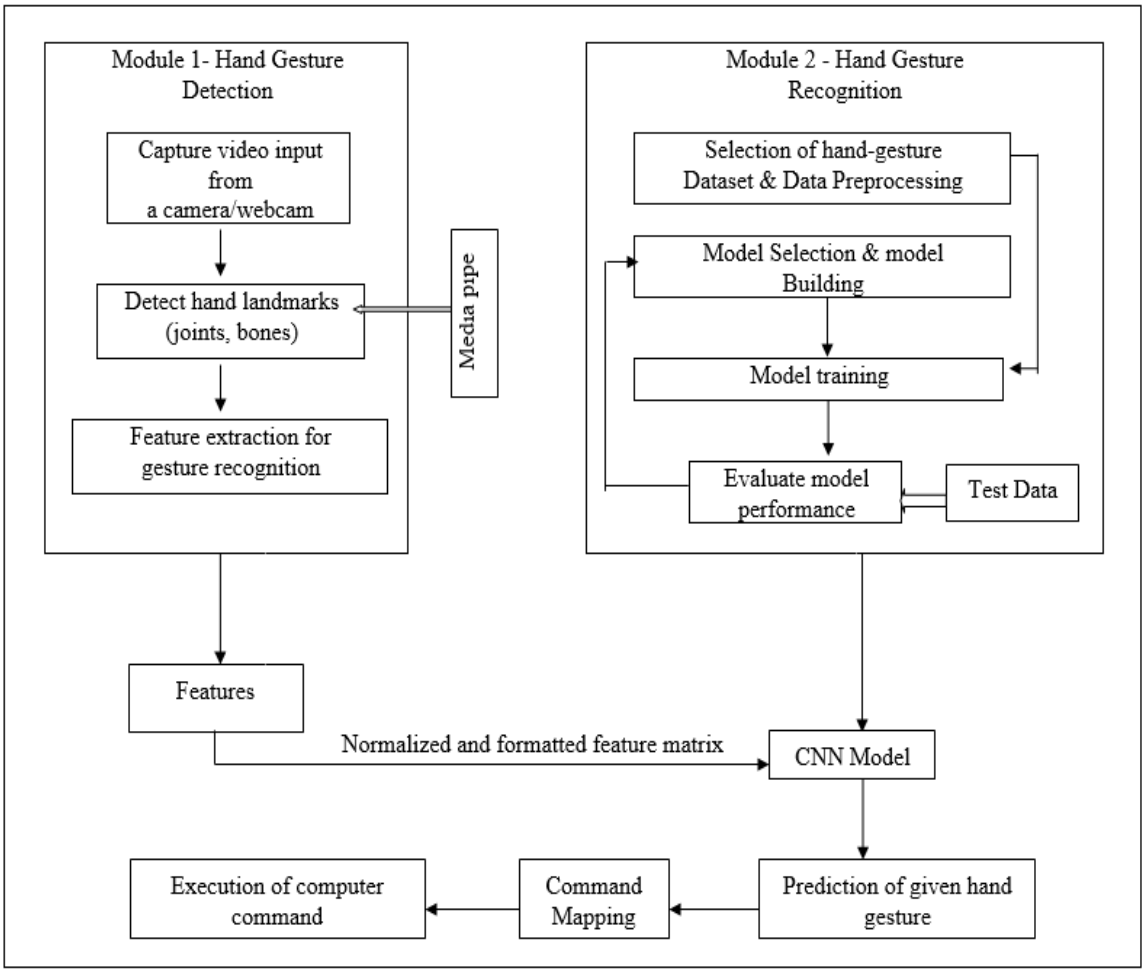

Figure 1: *System Architecture*

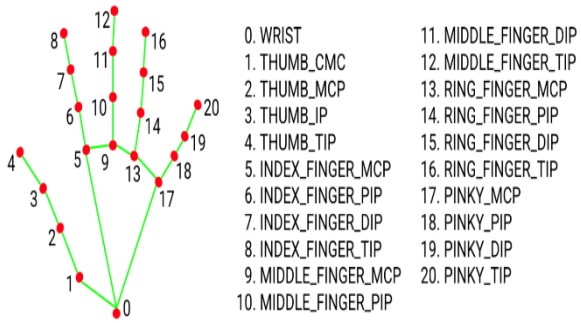

Figure 2. *MediaPipe Hands - 21 landmarks labelled from 0 - 20*

This paper proposes a system that can execute computer commands based on hand gestures as input. The system utilizes skeleton-based features for the detection and CNN model for the recognition of hand gestures, and presents an interactive mouse control paradigm using hand gestures, which is receiving increased attention in the design of computer assistive technologies. Although deep learning and skeleton detection have traditionally been employed independently, this study integrates both methodologies into a unified system to attain high

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

accuracy in gesture recognition and minimal latency for real-time command execution. Additionally, the study incorporates an adaptive learning mechanism that dynamically adjusts to variations in users' hand shapes, sizes, movement patterns, lighting conditions, and skin tones by utilizing a dataset with variations for model training and skeleton object detection techniques.

2. Methodology

The overall architecture of the proposed system consists of two modules, shown in Figure 1. Module 1 is a simple user interface to communicate with the user and get the video frames of the shown gestures. It then processes the input, detects the gesture, and creates a feature matrix of the gesture. Module 2 is the proposed CNN model, which can recognize hand gestures. These two modules are then integrated and developed as a system to detect and predict the given gesture, and then the system will map it to the corresponding command and execute the command.

A. Module 1: Hand Gesture Detection

This module is designed to detect the hand gestures shown in front of the camera. The module uses the video frame as input and detects the 21 hand joints using Mediapipe Hands. The coordinates (x, y, z) of each of the 21 joints are then extracted and formatted as a feature vector with 63 points. MediaPipe Hands is a solution developed by Google to track hands and fingers, leveraging machine learning to track the movements in real time. It can detect up to 21 landmarks (shown in Figure 2) per hand even in challenging conditions like partially obscured or overlapping hands. Mediapipe Hands has proven to provide promising results in hand detection approaches (Bora et al., 2022; Sundar and Bagyammal, 2022; Latreche et al., 2023).

B. Module 2: Hand gesture recognition

1) Dataset

The "Hand Gestures Dataset" on Roboflow Universe is publicly available (dataset) and designed to train computer vision models to recognize various hand gestures. This dataset includes 8,733 images of 21 different hand gestures, which are labeled and pre-processed to facilitate model training. The images have been captured under varying lighting, background and hand orientations which helps generalize across different environments. To capture the variability in hand shapes, sizes, and movements, the dataset includes gestures performed by multiple individuals with right and left hands and varying distances between the camera and the hand gestures maintained. This variability helps models learn to recognize gestures from different users with different physical characteristics. The dataset's diversity in hand poses, orientations, and backgrounds helps in creating robust models for applications like sign language recognition, and gesture-based control.

2) Data Preparation

Five easy-to-make hand gestures were chosen from the dataset, belonging to 5 classes. Each image was then sent through Mediapipe Hand to extract the 21 landmarks of each gesture position which consists of the relative x, y, and z coordinates creating a feature vector of 63 length for each gesture. The dataset was subsequently partitioned into training and testing sets, with 90% allocated for training and 10% reserved for testing. Figure 3 shows the gesture and its corresponding command.

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

3) Model Training

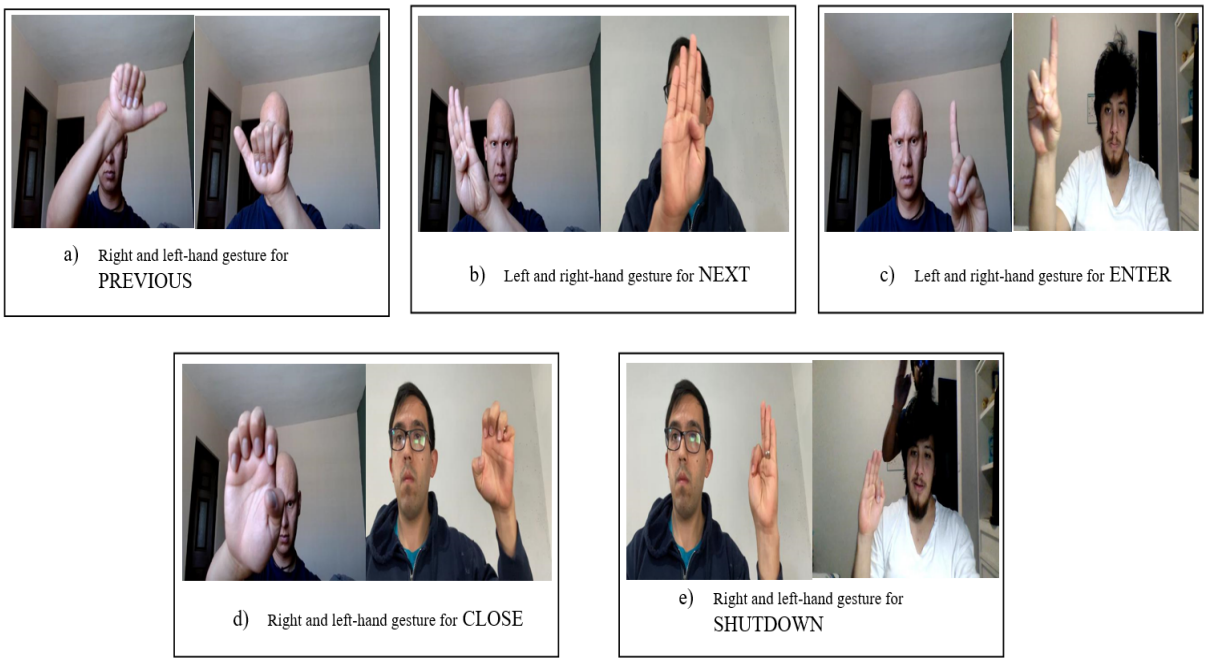

Figure 3. *Hand gestures and corresponding commands*

We employed a straightforward sequential CNN model featuring fully connected CNNs, Dense and ReLU activation and a few dropout layers. The input layer takes the input of shape (21,3,1). This represents the 21 landmarks, each with 3 coordinates (x,y,z), and the final dimension (1) is for the grayscale channel. The convolutional layers were implemented with a kernel size of 2x2 and activated with ReLU, which helps introduce non-linearity. MaxPooling layers were applied over a 2x1 window, with a 25% dropout rate to prevent overfitting. The final layer uses softmax activation for the probability distribution over the gesture classes. We have utilized Adam optimizer for its popularity and efficient handling of sparse gradients, with the categorical_crossentropy loss function as this is a multi-class classification task and the labels are one-hot encoded. The default learning rate was set to 0.001 while optimizing. The batch size was maintained as 32 samples per batch, and the model went through 100 epochs of training. Early stopping was beneficial in avoiding overfitting as the dataset size is relatively small. The model summary is shown in Figure 4. The model is relatively simple, more complex architectures like deeper CNNs could potentially improve accuracy but may require more computational resources for model training.

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 21, 3, 32)         160

 conv2d_1 (Conv2D)           (None, 21, 3, 32)         4128

 max_pooling2d (MaxPooling2D  (None, 10, 3, 32)        0
 )

 dropout (Dropout)           (None, 10, 3, 32)         0

 conv2d_2 (Conv2D)           (None, 10, 3, 64)         8256

 conv2d_3 (Conv2D)           (None, 10, 3, 64)         16448

 max_pooling2d_1 (MaxPooling  (None, 5, 3, 64)         0
 2D)

 dropout_1 (Dropout)         (None, 5, 3, 64)          0

 flatten (Flatten)           (None, 960)               0

 dense (Dense)               (None, 128)               123008

 dropout_2 (Dropout)         (None, 128)               0

 dense_1 (Dense)             (None, 5)                 645

=================================================================
Total params: 152,645
Trainable params: 152,645
Non-trainable params: 0
```

Figure 4. *Model Summary*

$$Accuracy = \frac{True\ Positives + True\ Negative}{Total\ Instances} \quad (1)$$

$$F1 - Score = 2 \times \frac{Precision.Recall}{Preciion + Recall} \quad (2)$$

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (3)$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (4)$$

4) Model Evaluation

To ensure the model's robustness, the testing dataset was created with varying backgrounds and lighting conditions of 3 different persons having varying skin tones and hand sizes in different directions. This dataset includes 75 images on average per class. The model's performance was evaluated using both accuracy and F1 score. Accuracy can be defined as the ratio of correctly predicted instances to the total instances (Equation 1), while the F1 score (Equation 2) represents the harmonic mean of precision (Equation 3) and recall (Equation 4)

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

5) System Integration

Module 1 formulates a feature vector of the detected hand gesture. Module 2 is a CNN model which can predict the given feature vector from module 1. Then the predicted gesture's relevant command is mapped, and executed with a voice command.

3. Results and discussion

The proposed deep learning-based hand gesture recognition system, utilizing skeleton object detection, has demonstrated impressive performance in recognizing hand gestures. The model was trained on a publicly available dataset comprising 8733 images across 21 different hand gestures, which were preprocessed to extract 21 hand landmarks using MediaPipe Hands.

The system achieved an accuracy score of 0.9711, meaning it correctly identified the hand gestures in nearly 97% of the cases. Figure 5 shows the accuracy and loss function of the model. This suggests the model's ability to accurately recognize true positive instances while minimizing both false positives and false negatives. Figure 6 shows the correctly predicted gestures of each label. This high accuracy indicates the robustness of the model in recognizing gestures across various poses, orientations, and backgrounds. The model attained a remarkable F1 score of 0.9711, reflecting its strong performance in balancing precision and recall.

Table 1
Precision, Recall, F1-Score, and Accuracy of each label

Labels	Precision	Recall	Accuracy	F1-score
Previous	1	1	1	1
Next	0.902	0.982	0.982	0.940
Enter	0.987	0.987	0.992	0.987
Close	1	0.977	0.992	0.988
Shutdown	0.974	0.914	0.976	0.943

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

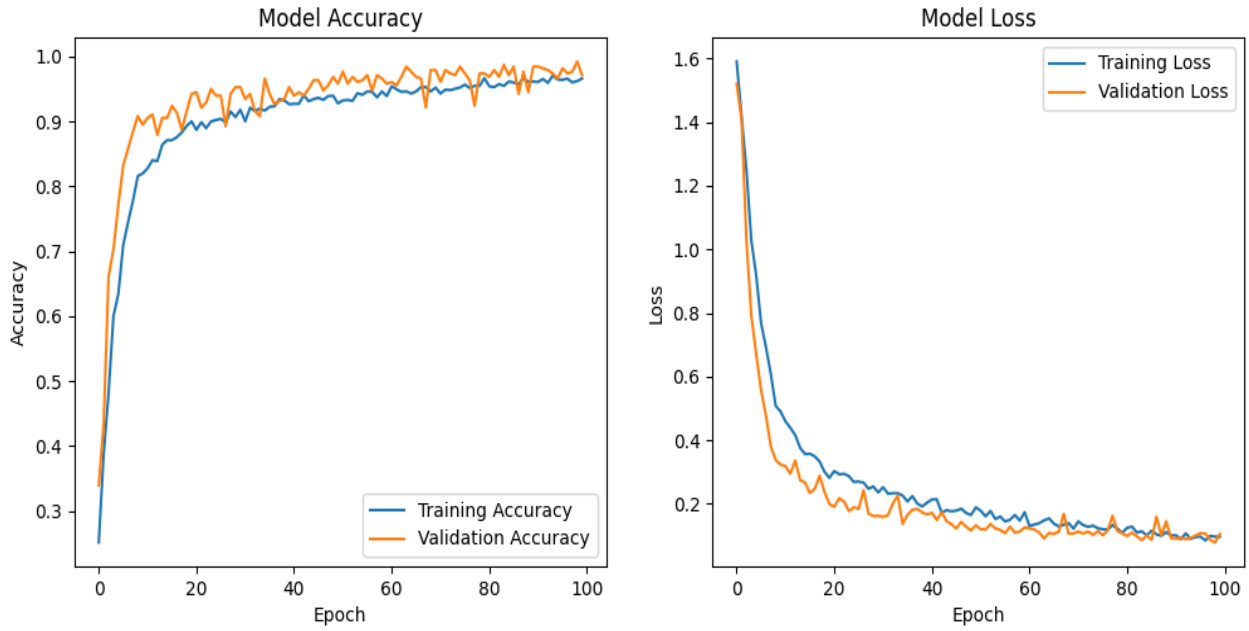

Figure 5. Accuracy and Loss Function

The classifier performs quite well overall, especially for the "close," "enter," "next," and "previous" categories. With very few misclassifications, the model demonstrates high precision and recall for these labels. The precision, recall, F1-score, and accuracy of each label are given in Table 1. These metrics indicate that the classifier generally performs well, but struggles slightly with the "shutdown" and "next" classes, as observed in their relatively lower F1-scores.

The confusion matrix shown in Figure 7 shows that the majority of misclassifications happen in the “shutdown” class, which tends to be confused with “next”. The error distribution across other classes is minimal, and no pattern suggests frequent cross-category confusion, which we see as a positive sign.

Figure 6. *Prediction of gestures*

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

Table 2
CNN Model performance comparison with skeletal dataset and image dataset

Dataset	Precision	Recall	Accuracy	F1-Score
Skeletal Landmark	0.9721	0.9711	0.9711	0.9711
Image	0.8986	0.8985	0.9101	0.8713

We examined the model's performance with the same dataset without extracting skeletal landmarks data, and its performance is compared with our model in Table 2. It demonstrates the fact that the skeletal dataset gives better performance through precision, recall, accuracy, and F1-score.

Including skeletal landmarks in the dataset leads to higher accuracy and F1-score, highlighting the importance of capturing detailed pose or gesture information. This additional data likely helps the model better understand subtle differences between actions, resulting in fewer misclassifications and more robust performance.

The integration of the detection and recognition modules with a command execution interface was successful. The system could map the recognized hand gestures to corresponding computer commands and execute them. The average time for the recognition and execution is 0.04485 seconds, thus validating the practical application of the proposed method in real-time scenarios.

4. Conclusion

In this paper, we explored the impact of using skeletal landmarks data to improve the performance of a classifier designed for action recognition. By comparing the results with and without skeletal data, our experiments demonstrated a significant improvement in both accuracy and F1-score when the skeletal information was included. The confusion matrix analysis revealed that the model with skeletal data was better at distinguishing between closely related actions, leading to fewer misclassifications. By focusing on skeletal features, the model proves to be robust against variations in lighting, hand appearance, and background, making it suitable for real-world applications. This research also confirms the effectiveness of using deep learning and skeleton object detection for hand gesture recognition, particularly in human-computer interaction contexts. The lower execution time underscores its potential for enhancing user experience in areas such as gaming, virtual reality, and assistive technologies.

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

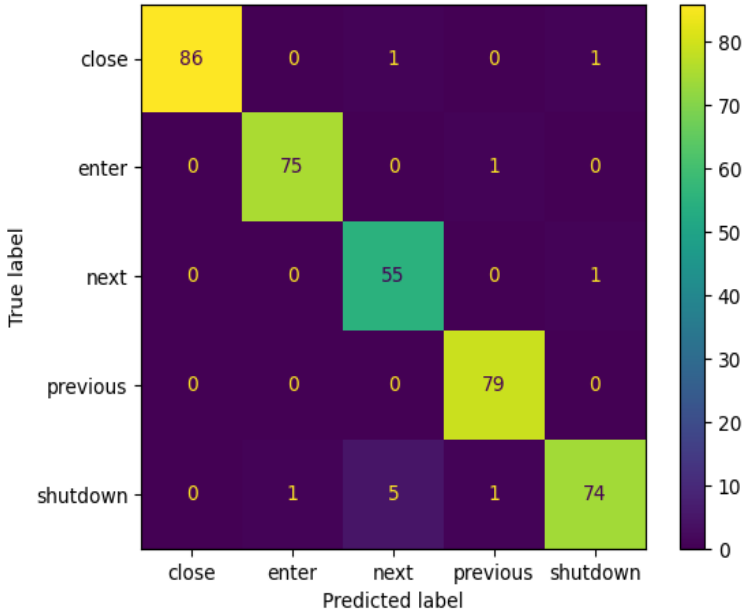

Figure 7. *Confusion Matrix*

However, this research has a notable limitation in that only five gestures were selected, and the computing commands considered were relatively few. While the results highlight the effectiveness of skeletal landmarks in improving action recognition, the scope of this study remains constrained by the small number of gestures and commands. Future research can address this limitation by expanding the range of gestures and incorporating a more comprehensive set of computing commands, ultimately providing a broader and more versatile model for real-world applications enhancing the practical utility of gesture-based interfaces.

References

Bai, Y. *et al.* (2019) 'A skeleton object detection-based dynamic gesture recognition method', *Proceedings of the 2019 IEEE 16th International Conference on Networking, Sensing and Control, ICNSC 2019*, (315100104), pp. 212–217. doi: 10.1109/ICNSC.2019.8743166.

Bora, J. *et al.* (2022) 'Real-time Assamese Sign Language Recognition using MediaPipe and Deep Learning', *Procedia Computer Science*, 218(2022), pp. 1384–1393. doi: 10.1016/j.procs.2023.01.117.

Datta, A. *et al.* (2020) 'Road Object Detection in Bangladesh using Faster R-CNN: A Deep Learning Approach', *Proceedings of 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering, WIECON-ECE 2020*, pp. 348–351. doi: 10.1109/WIECON-ECE52138.2020.9397954.

Hrishikesh, P. *et al.* (2024) 'Vision Based Gesture Recognition', *Procedia Computer Science*, 235, pp. 303–315. doi: 10.1016/j.procs.2024.04.031.

Hu, S. *et al.* (2020) 'Unified diagnosis framework for automated nuclear cataract grading based on smartphone slit-lamp images', *IEEE Access*, 8, pp. 174169–174178. doi: 10.1109/ACCESS.2020.3025346.

Latreche, A. *et al.* (2023) 'Reliability and validity analysis of MediaPipe-based measurement system for some human rehabilitation motions', *Measurement: Journal of the*

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

International Measurement Confederation, 214. doi: 10.1016/j.measurement.2023.112826.

Liu, Y. (2018) 'An Improved Faster R-CNN for Object Detection', *Proceedings - 2018 11th International Symposium on Computational Intelligence and Design, ISCID 2018*, 2, pp. 119–123. doi: 10.1109/ISCID.2018.10128.

Lu, Y., Zhang, L. and Xie, W. (2020) 'YOLO-compact: An Efficient YOLO Network for Single Category Real-time Object Detection', *Proceedings of the 32nd Chinese Control and Decision Conference, CCDC 2020*, pp. 1931–1936. doi: 10.1109/CCDC49329.2020.9164580.

Luong, D. D., Lee, S. and Kim, T. S. (2013) 'Human computer interface using the recognized finger parts of hand depth silhouette via random forests', *International Conference on Control, Automation and Systems*, (Iccas), pp. 905–909. doi: 10.1109/ICCAS.2013.6704043.

Nguyen, T.-N., Huynh, H.-H. and Meunier, J. (2015) 'Static Hand Gesture Recognition Using Principal Component Analysis Combined with Artificial Neural Network', *Journal of Automation and Control Engineering*, 3(1), pp. 40–45. doi: 10.12720/joace.3.1.40-45.

Reddy, D. A., Sahoo, J. P. and Ari, S. (2018) 'Hand Gesture Recognition Using Local Histogram Feature Descriptor', *Proceedings of the 2nd International Conference on Trends in Electronics and Informatics, ICOEI 2018*, (Icoei), pp. 199–203. doi: 10.1109/ICOEI.2018.8553849.

S. Saxena *et al.* (2022) 'Hand Gesture Recognition using YOLO Models for Hearing and Speech Impaired People', in *2022 IEEE Students Conference on Engineering and Systems (SCES)*. India: IEEE. doi: 10.1109/SCES55490.2022.9887751.

Sapkota, R., Ahmed, D. and Karkee, M. (2024) 'Comparing YOLOv8 and Mask R-CNN for instance segmentation in complex orchard environments', *Artificial Intelligence in Agriculture*, 13, pp. 84–99. doi: 10.1016/j.aiia.2024.07.001.

De Smedt, Q., Wannous, H. and Vandeborre, J. P. (2016) 'Skeleton-Based Dynamic Hand Gesture Recognition', *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1206–1214. doi: 10.1109/CVPRW.2016.153.

Song, W. *et al.* (2019) 'Design of a Flexible Wearable Smart sEMG Recorder Integrated Gradient Boosting Decision Tree Based Hand Gesture Recognition', *IEEE transactions on biomedical circuits and systems*, 13(6), pp. 1563–1574. doi: 10.1109/TBCAS.2019.2953998.

Sonkusare, J. S. *et al.* (2015) 'A review on hand gesture recognition system', *Proceedings - 1st International Conference on Computing, Communication, Control and Automation, ICCUBEA 2015*, pp. 790–794. doi: 10.1109/ICCUBEA.2015.158.

Sundar, B. and Bagyammal, T. (2022) 'American Sign Language Recognition for Alphabets Using MediaPipe and LSTM', *Procedia Computer Science*, 215, pp. 642–651. doi: 10.1016/j.procs.2022.12.066.

Terreran, M., Barcellona, L. and Ghidoni, S. (2023) 'A general skeleton-based action and gesture recognition framework for human–robot collaboration', *Robotics and Autonomous Systems*, 170(September), p. 104523. doi: 10.1016/j.robot.2023.104523.

Wang, F., Hu, R. and Jin, Y. (2021) 'Research on gesture image recognition method based on transfer learning', *Procedia Computer Science*, 187(2019), pp. 140–145. doi: 10.1016/j.procs.2021.04.044.

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).

Wang, N., Cai, A. and Zhang, S. (2018) 'The Study of RNN Enhanced Convolutional Neural Network for Fast Object Detection Based on the Spatial Context Multi-Fusion Features', *Proceedings - 2018 11th International Symposium on Computational Intelligence and Design, ISCID 2018*, 1, pp. 136–140. doi: 10.1109/ISCID.2018.00038.

Zhao, J. *et al.* (2023) 'Hand Gesture Recognition Based on Deep Learning', *2023 International Conference on Digital Applications, Transformation and Economy, ICDATE 2023*, 14(4), pp. 307–320. doi: 10.1109/ICDATE58146.2023.10248500.

Published by the Faculty of Technology, South Eastern University of Sri Lanka, under a Creative Commons Attribution 4.0 International License (CC BY 4.0). The copyright remains with the author(s).