Vol. 7, No. 2, August 2025, pp. 223~229, p-ISSN 2685-2381, e-ISSN 2715-2626
http://dx.doi.org/10.28989/avitec.v7i2.3132

Hand Gesture-Based Human-Computer Interaction using MediaPipe and OpenCV

Risma Dwi Tjutarjo Putri[1], Lasmadi[2], Anggraini Kusumaningrum[3], Riani Nurdin[4], Yenni Astuti[5]*
[1,2,5]Department of Electrical Engineering, Institut Teknologi Dirgantara Adisutjipto, Indonesia
[3]Department of Informatics, Institut Teknologi Dirgantara Adisutjipto, Indonesia
[4]Department of Industrial Engineering, Institut Teknologi Dirgantara Adisutjipto, Indonesia

Article Info

Article history:
Submitted July 3, 2025
Accepted July 16, 2025
Published July 18, 2025

Keywords:
Hand gesture;
gesture recognition;
human-computer interaction;
MediaPipe;
OpenCV.

ABSTRACT

This study presents the design and implementation of a real-time hand gesture recognition system for directional movement using MediaPipe and OpenCV. The system aims to enhance Human-Computer Interaction (HCI) by recognizing four primary hand gestures—forward, backward, left, and right—based on real-time video input from a standard webcam. The proposed method extracts 21 hand landmarks using MediaPipe, then analyzes landmark displacement to determine the direction of movement. Experiments were conducted under three lighting conditions (bright, moderate, dim) and at three distances (200 cm, 300 cm, and 450 cm). Results show that the system achieved 100% recognition accuracy for all gestures at 200 cm. At 300 cm, accuracy slightly decreased, particularly for backward gestures (down to 77.5%). At 450 cm, performance dropped significantly, with accuracy for some gestures falling below 30%, especially under dim lighting. These findings demonstrate that the proposed system performs reliably at short to medium distances and is sensitive to lighting conditions and user proximity. This research contributes to the development of touchless interfaces for smart environments, presentations, and other interactive applications.

Corresponding Author:
Yenni Astuti,
Department of Electrical Engineering, Institut Teknologi Dirgantara Adisutjipto,
Jalan Janti, Blok-R, Lanud Adisutjipto Yogyakarta, Indonesia.
Email: *yenniastuti@itda.ac.id

1. INTRODUCTION

Recent advancements in Human–Computer Interaction (HCI) have shifted focus towards touchless interfaces [1], [2], especially hand gesture-based systems, enabling more natural and intuitive user experiences [3]. Deep learning approaches using convolutional neural networks (CNNs) have significantly improved gesture recognition accuracy, achieving over 90% on benchmark datasets [4], [5], but often require high-end hardware and large training datasets. Other studies have employed wearable or sensor-based systems for gesture recognition on edge devices [6], which, while precise, are less practical for general users due to cost and hardware constraints. MediaPipe has recently emerged as a lightweight alternative for real-time hand tracking, with applications in presentation control [7] and sign language recognition using LSTM models [8]. These studies demonstrate the effectiveness of landmark-based pipelines but are often limited to domain-specific tasks or require additional training complexity. A previous study also explored a more accessible approach using histogram-based features and Euclidean distance on static hand images, achieving only 30–60% accuracy [9], highlighting the limitations of static methods. In contrast, this study aims to develop a general-purpose, low-cost gesture recognition system using only a standard webcam, without machine learning classifiers or specialized sensors, and evaluate its performance under various lighting and distance conditions.

Drawing from these works, this research proposes a real-time directional hand gesture recognition system utilizing MediaPipe's 21-landmark detection and OpenCV image processing tools. The decision to use MediaPipe was based on its balance of performance and efficiency. Unlike many deep learning-based gesture recognition systems that require large datasets and high computational resources, MediaPipe offers pre-trained, lightweight models capable of real-time hand tracking using only RGB input. Its ability to deliver 21 hand landmarks with high spatial precision, even on standard CPU hardware, makes it an ideal choice for low-cost applications. This aligns with the study's goal to create an accessible and deployable gesture recognition system without relying on GPUs or specialized hardware. Furthermore, OpenCV was selected due to its compatibility with MediaPipe and its efficient handling of image frames, which is essential for real-time processing. Moreover, its wide adoption, rich documentation, and open-source nature make it a practical tool for academic research and rapid prototyping, especially in environments with limited computational infrastructure.

The system is designed to be low-cost, lightweight, and easily deployable using a standard webcam and widely available software libraries. To assess its feasibility and robustness, the system is evaluated across different user-to-camera distances and lighting conditions. While several previous studies have demonstrated the use of deep learning models, high-resolution sensors, or structured environments for accurate hand gesture recognition, such approaches are often constrained by high hardware costs and limited scalability. In contrast, this study addresses the research gap by offering a low-cost, real-time alternative that performs well under realistic environmental conditions—without relying on depth sensors or complex neural networks. This contribution is significant for enabling practical, accessible, and scalable gesture-based interaction in everyday settings, such as smart environments, presentations, and assistive technologies.

This paper is structured as follows: Section 2 outlines the methodology (system architecture, data acquisition, gesture classification). Section 3 presents experimental results and discussion. Section 4 concludes with findings and suggestions for future enhancements.

2. METHODOLOGY

This research employed an experimental approach to develop and evaluate a real-time directional hand gesture recognition system based on MediaPipe and OpenCV. The methodology covered system design, hardware and software setup, data acquisition, and performance evaluation.

2.1 System Design.

The system was designed to recognize four basic hand movement directions—forward, backward, left, and right—based on real-time video captured by a standard webcam. Using MediaPipe's hand tracking module, the system detects and extracts 21 hand landmarks (see Figure 1), which are then processed using OpenCV and NumPy to analyze spatial displacement across video frames. The directional classification is determined by comparing changes in landmark positions relative to time, following the system flowchart illustrated in Figure 2. The system workflow begins with camera initialization and frame acquisition. Each frame is processed to detect the presence of a hand. If a hand is detected, landmark coordinates are extracted and analyzed. The recognized direction is then visualized on the screen, and results are recorded for evaluation.

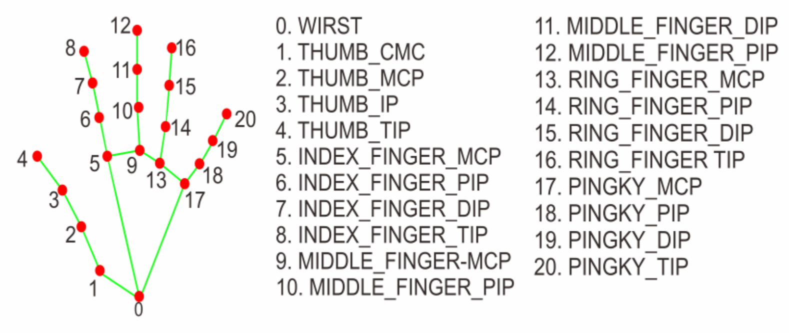

Figure 1. Hand Landmark [10]

The functional components of the system are illustrated in Figure 3, which shows the block-level architecture. The process starts from image acquisition, followed by hand detection and landmark extraction using MediaPipe. The movement analysis module then determines the gesture direction, which is finally passed to the output module for visualization and logging. The movement recognition process is based on analyzing the temporal displacement of selected hand landmarks—particularly the wrist and index fingertip—across a sequence of consecutive frames. The system does not rely on machine learning classifiers; instead, it applies a rule-based method that calculates the difference in landmark coordinates over time to infer the direction of motion. For instance, a forward gesture is recognized when the z-coordinate (depth proxy) of the wrist consistently decreases over a small window of frames, indicating movement toward the camera. A calibrated threshold is applied to filter out minor, unintentional movements. This approach allows real-time classification without requiring labeled datasets or model training, making it suitable for lightweight, low-latency applications.

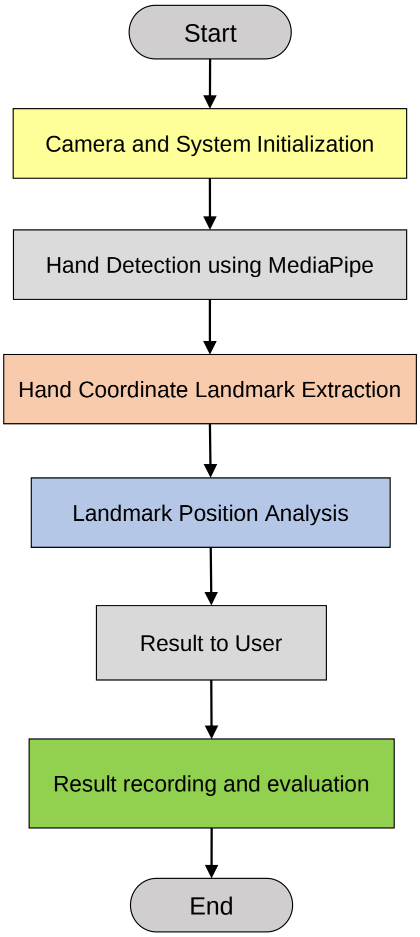
Start
Camera and System Initialization
Hand Detection using MediaPipe
Hand Coordinate Landmark Extraction
Landmark Position Analysis
Result to User
Result recording and evaluation
End

Figure 2. System flowchart

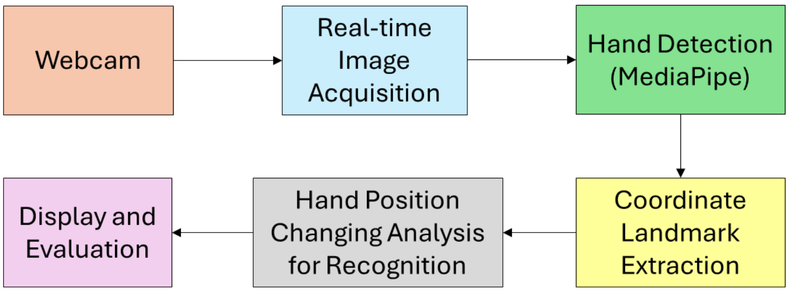
Webcam
Real-time Image Acquisition
Hand Detection (MediaPipe)
Coordinate Landmark Extraction
Hand Position Changing Analysis for Recognition
Display and Evaluation

Figure 3. System block diagram

2.2 Hardware and Software Setup

The system was implemented on a laptop with the following specifications: Intel Celeron N4500 processor, 4 GB RAM, Windows 10 operating system, and webcam with at least 720p resolution. The software environment included Python 3.x with supporting libraries such as MediaPipe, OpenCV, NumPy, Pandas, and Matplotlib.

2.3 Data Acquisition

Data was collected through live video capture in a laboratory environment. The user performed the four directional hand movements under three lighting conditions—bright, moderate, and dim—at three distances from the camera: 200 cm, 300 cm, and 450 cm. The dataset was collected from ten participants with varying hand sizes and skin tones to ensure diversity in the gesture samples. Each participant performed four directional gestures—moving the hand forward (toward the camera), backward (away from the camera), left, and right—under three lighting conditions (bright, moderate, and dim) and at three distances (200 cm, 300 cm, and 450 cm). Each gesture was repeated 20 times per lighting and distance scenario, resulting in a total of 7,200 gesture samples (4 gestures × 3 lighting levels × 3 distances × 10 participants × 20 repetitions). The gestures were performed in a consistent manner to maintain temporal stability while allowing for natural variations in hand orientation and motion speed. The data acquisition scenario is shown in Figure 4.

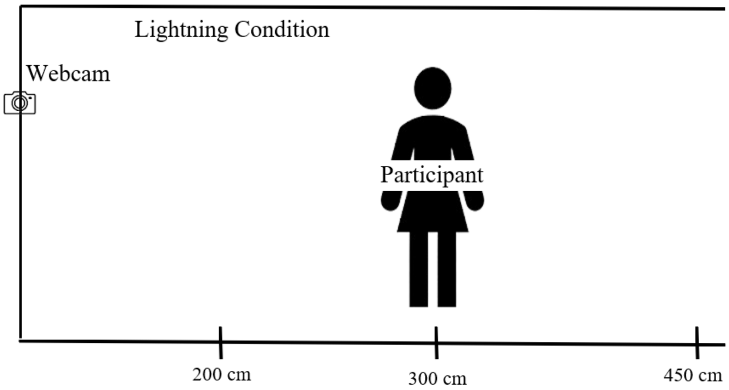

Figure 4. Data acquisition scenario

2.4 Performance Evaluation

System performance was assessed using accuracy as the primary metric, calculated as the ratio of correctly recognized gestures to the total number of attempts, as shown in Equation (1). This evaluation provided insights into the system’s reliability across varying environmental conditions, including differences in lighting and user-to-camera distances. Accuracy scores were tabulated and compared across all test conditions to analyze system robustness.

$$Accuracy = \frac{Correct\ recognition}{Total\ attempts} \times 100\% \tag{1}$$

2.5 MediaPipe

MediaPipe is a framework developed by Google for building multimodal machine learning pipelines and is widely used in computer vision applications [8]. It provides a set of pre-trained models and tools that enable efficient and flexible development of real-time perception systems. One of its most significant contributions to hand gesture recognition is the MediaPipe Hands solution, which performs both palm detection and landmark localization in a single pipeline [11].

The theoretical foundation of MediaPipe’s hand tracking lies in a two-stage machine learning architecture. First, a Palm Detection Model identifies the presence of a hand by locating the palm using a lightweight detector trained specifically on palm images, rather than relying on full hand bounding boxes. This method enhances robustness against occlusion and reduces the computational load by focusing only on the most stable region of the hand. Once the palm is detected, the Hand Landmark Model refines the analysis by cropping the region of interest (ROI) and predicting 21 hand landmarks, including the wrist, finger joints, and fingertips. The resulting coordinates are provided in normalized 2D image space (x, y) along with a relative depth estimate (z), enabling accurate tracking of hand gestures in real time.

This combination allows for fast, accurate, and real-time hand tracking using only RGB input, without requiring depth sensors. The inference pipeline is optimized for CPU and mobile deployment, making it suitable for low-cost and embedded systems. In gesture recognition, these landmarks serve as the primary feature set from which movement patterns and gesture classifications are derived [12]. By observing the temporal displacement of landmarks such as the wrist and index fingertip across video frames, directional movements can be effectively detected. MediaPipe's modular and graph-based architecture also allows for integration with other vision tools such as OpenCV, enhancing the system’s flexibility for academic and practical implementations.

2.6 OpenCV

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library that provides over 2,500 optimized algorithms for real-time image and video processing [13]. Since its inception by Intel in 2000, OpenCV has become one of the most widely used libraries in academic and industrial computer vision projects. The theoretical foundation of OpenCV lies in its capability to manipulate matrices, perform image transformation, and extract features from visual data. OpenCV supports a wide range of functionalities, including object detection, edge detection, motion tracking, and feature point analysis, all of which are essential for developing interactive vision-based systems. OpenCV’s compatibility with Python, C++, and other languages makes it a flexible choice for rapid prototyping and deployment in embedded systems or academic experimentation [14]. Its efficiency in frame handling and array manipulation makes it well-suited to work alongside MediaPipe in building real-time gesture recognition pipelines.

In the context of this research, OpenCV plays several crucial roles in the gesture recognition pipeline. It is first used to initialize the camera and capture video frames in real-time, with each frame processed and formatted for compatibility with MediaPipe [15]—this includes resizing, flipping, and converting color spaces (e.g., from BGR to RGB). For visualization and debugging purposes, OpenCV offers drawing utilities such as cv2.circle, cv2.line, and cv2.putText, which enable the system to render hand landmarks in real time, providing visual feedback and aiding in trajectory analysis. Together with NumPy, OpenCV supports numerical operations on landmark coordinates obtained from MediaPipe. By analyzing the positional changes of specific landmarks across sequential frames, the system determines the direction of hand movement—forward, backward, left, or right. To reduce false positives, a calibrated threshold value is used to filter insignificant movements. The system only classifies and records a valid gesture when the displacement of key points (e.g., wrist, fingertips) exceeds the threshold.

3. RESULTS AND DISCUSSION

The performance of the proposed gesture recognition system was evaluated by testing directional hand movements under varying conditions of distance and lighting. A total of 14,400 gesture samples were collected across all combinations of distance (200 cm, 300 cm, 450 cm), lighting (bright, moderate, dim), and movement directions (forward, backward, right, left). Each configuration consisted of 200 repetitions for consistency.

3.1 Accuracy by Distance and Lighting

Table 1 summarizes the recognition accuracy for each directional gesture at different distances and lighting conditions. The system achieved 100% accuracy at 200 cm for all directions and lighting conditions, indicating that the MediaPipe-based approach is highly effective at short range. At 300 cm, the system remained robust, with only moderate decreases in accuracy for the backward direction (as low as 77.5% in dim lighting). However, a significant performance drop occurred at 450 cm, especially for backward and left movements under low-light conditions, where accuracy dropped to 0%.

Table 1. Recognition accuracy by distance, lighting, and direction (%)

Distance	Lighting	Forward	Backward	Right	Left
200 cm	Bright	100	100	100	100
200 cm	Moderate	100	100	100	100
200 cm	Dim	100	100	100	100
300 cm	Bright	100	85	100	100
300 cm	Moderate	100	79	100	100
300 cm	Dim	100	77.5	100	95
450 cm	Bright	74	15	30	10
450 cm	Moderate	53	0	10	0
450 cm	Dim	28.5	0	10	0

3.2 Performance Analysis

The performance degradation at longer distances can be attributed to the reduced resolution and reliability of hand landmark detection. MediaPipe's tracking becomes less stable as the hand occupies fewer pixels in the frame, leading to misclassification or failure to detect movement. Lighting also played a crucial role. Dim conditions resulted in a less distinct contrast between the hand and background, making it difficult for the system to maintain consistent tracking of the hand's landmarks. Interestingly, forward gestures were recognized more reliably across all distances compared to backward or lateral movements. This is likely due to more pronounced displacement of fingertips toward the camera, resulting in clearer directional changes.

Compared to Ref. [7], which achieved 97% accuracy for real-time gesture recognition in controlled indoor environments using MediaPipe, our system achieved 100% accuracy at 200 cm under three lighting levels, indicating high robustness for short-range interaction. However, recognition performance dropped significantly to 450 cm, particularly under dim lighting, highlighting the limitations of RGB-based landmark detection at long range. In contrast, Ref. [8] employed a more complex pipeline combining MediaPipe with LSTM to recognize sign language gestures, demonstrating strong temporal modeling but at the cost of increased computational complexity and the need for labeled gesture sequences. Our study focuses on a rule-based approach using spatial displacement analysis, which allows for real-time performance without requiring large datasets or training. Although it lacks temporal learning capabilities, the simplicity and efficiency of our method make it well-suited for embedded applications and environments where lightweight implementation is critical. This comparison shows that while deep learning models offer superior flexibility, rule-based systems can remain competitive when optimized for specific, well-defined tasks.

3.3 Implications and Limitations

The proposed system has several practical implications for real-world human-computer interaction scenarios. Its ability to recognize hand gestures in real-time using only a standard webcam makes it suitable for contactless control in various settings, such as classroom presentations, smart kiosks, home automation, and assistive technologies for users with mobility impairments. Particularly in the context of post-pandemic digital interaction, where minimizing physical contact is desirable, this system offers a hygienic and intuitive alternative to traditional input devices. The simplicity of its implementation also makes it ideal for educational purposes, rapid prototyping, and deployment in resource-limited environments where access to high-end hardware is restricted. Despite these advantages, the system's performance degrades significantly at longer distances or in poor lighting, indicating a limitation of using RGB-based computer vision alone without depth sensing or adaptive preprocessing. For practical deployment in uncontrolled environments, further enhancements are recommended, such as integrating adaptive brightness correction, background subtraction, or adding depth sensors to improve spatial accuracy at longer ranges.

4. CONCLUSION

This study presented the design and implementation of a real-time hand gesture recognition system using MediaPipe and OpenCV for directional human–computer interaction. The system successfully recognized four basic gestures—forward, backward, left, and right—under various environmental conditions, utilizing only a standard webcam and open-source tools, thereby demonstrating a low-cost yet effective solution for gesture-based interaction. Experimental results demonstrated that the system achieved perfect accuracy (100%) at 200 cm for all lighting conditions. At 300 cm, performance remained high, with only a moderate decrease in accuracy for backward gestures, particularly under dim lighting. However, at 450 cm, system accuracy declined significantly, especially for backward and left gestures, which were sometimes undetectable in low-light environments. These findings confirm that the proposed system performs effectively within short to medium interaction ranges and under well-lit conditions. The use of MediaPipe's hand tracking model proved to be highly efficient for real-time gesture recognition without requiring high-end hardware.

Future work will focus on improving system robustness by incorporating brightness normalization, gesture smoothing algorithms, and exploring integration with depth sensors or machine learning classifiers to extend functionality in more complex environments and longer distances.

ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to the Institute for Research and Community Service (LPPM) of Institut Teknologi Dirgantara Adisutjipto for funding and supporting this research through the 2025 Internal Research Grant Program.

REFERENCE

[1] V. Gentile, A. Adjorlu, S. Serafin, D. Rocchesso, and S. Sorce, "Touch or touchless?: Evaluating usability of interactive displays for persons with autistic spectrum disorders," in *Proceedings - Pervasive Displays 2019 - 8th ACM International Symposium on Pervasive Displays, PerDis 2019*, 2019, pp. 1–7. https://dx.doi.org/10.1145/3321335.3324946

[2] M. Modaberi, "The Role of Gesture-Based Interaction in Improving User Satisfaction for Touchless Interfaces," *Int. J. Adv. Hum. Comput. Interact.*, vol. 2, no. 2, pp. 20–32, 2024.

[3] Yaseen, O. J. Kwon, J. Kim, J. Lee, and F. Ullah, "Evaluation of Benchmark Datasets and Deep Learning Models with Pre-Trained Weights for Vision-Based Dynamic Hand Gesture Recognition," *Appl. Sci.*, vol. 15, no. 11, 2025. https://dx.doi.org/10.3390/app15116045

[4] P. Xu, "A Real-time Hand Gesture Recognition and Human-Computer Interaction System," *arXiv:1704.07296*, pp. 1–8, 2017.

[5] O. Köpüklü, A. Gunduz, N. Kose, and G. Rigoll, "Real-time hand gesture detection and classification using convolutional neural networks," *Proc. - 14th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2019*, 2019. https://dx.doi.org/10.1109/FG.2019.8756576

[6] E. Fertl, E. Castillo, G. Stettinger, M. P. Cuéllar, and D. P. Morales, "Hand Gesture Recognition on Edge Devices: Sensor Technologies, Algorithms, and Processing Hardware," *Sensors*, vol. 25, no. 6, pp. 1–46, 2025. https://dx.doi.org/10.3390/s25061687

[7] A. D. Agustiani, S. M. Putri, P. Hidayatullah, and M. R. Sholahuddin, "Penggunaan MediaPipe untuk Pengenalan Gesture Tangan Real-Time dalam Pengendalian Presentasi [The use of MediaPipe for real-time hand gesture recognition in presentation control]," *Media J. Informatics*, vol. 16, no. 2, 2024. https://dx.doi.org/10.35194/mji.v16i2.4788

[8] M. Z. Uddin, C. Boletsis, and P. Rudshavn, "Real-Time Norwegian Sign Language Recognition Using MediaPipe and LSTM," *Multimodal Technol. Interact.*, vol. 9, no. 3, pp. 1–15, 2025. https://dx.doi.org/10.3390/mti9030023.

[9] Y. Astuti and I. D. Ariyanti, “Recognition of hand gestures using image with histogram feature extraction and Euclidean distance classification method,” vol. 13, no. 2, pp. 117–122, 2024. https://dx.doi.org/10.28989/compiler.v13i2.2640

[10] D. Oktaviyanti, A. Nugroho, and A. F. Suni, “Pemanfaatan Hand Tracking untuk Membuat Program Virtual Painter sebagai Alternatif Menggambar Digital,” *Petir*, vol. 15, no. 2, pp. 287–294, 2022 https://dx.doi.org/10.33322/petir.v15i2.1523

[11] Indriani, M. Harris, and A. S. Agoes, “Applying Hand Gesture Recognition for User Guide Application Using MediaPipe,” *Proc. 2nd Int. Semin. Sci. Appl. Technol. (ISSAT 2021)*, vol. 207, no. Issat, pp. 101–108, 2021. https://dx.doi.org/10.2991/aer.k.211106.017

[12] G. Sánchez-Brizuela, A. Cisnal, E. de la Fuente-López, J. C. Fraile, and J. Pérez-Turiel, “Lightweight real-time hand segmentation leveraging MediaPipe landmark detection,” *Virtual Real.*, vol. 27, no. 4, pp. 3125–3132, 2023. https://dx.doi.org/10.1007/s10055-023-00858-0

[13] J. Abedalrahim, J. Alsayaydeh, T. Lee, C. Jie, R. Bacarra, and B. Ogunshola, “Handwritten text recognition system using Raspberry Pi with OpenCV TensorFlow,” *Int. J. Electr. Comput. Eng.*, vol. 15, no. 2, pp. 2291–2303, 2025. https://dx.doi.org/10.11591/ijece.v15i2.pp2291-2303

[14] K. Patil, S. Ladake, S. Nirgude, and V. Naphade, “Translating Hands Gestures into Text and Speech,” *Int. J. Ingenious Res. Invent. Dev.*, vol. 4, no. 1, pp. 163–172, 2025. https://dx.doi.org/10.5281/zenodo.15168979

[15] P. Abirami, B. Triveni, and D. A. Reddy, “International Journal of Innovative Research in Science Engineering and Technology (IJIRSET) Enhanced Communication through Hand Gesture Recognition and Speech Recognition,” *Int. J. Innov. Res. Sci. Eng. Technol.*, vol. 14, no. 4, pp. 9273–9279, 2025. https://dx.doi.org/10.15680/IJIRSET.2025.1404466