

Real Time Machine Learning based Assistive Communication System Using Mid-Air Gestures and Voice Feedback for Children with Physical Disabilities

Dr. Anand M¹, Dr. Vishwesh J², Padma R³, Dr. Usha R G⁴, *Dr. Pradeep Kumar R⁵, Dr. Chaya P⁶

¹ Associate Professor, Department of Information Science and Engineering, GSSS Institute of Engineering and Technology for Women, Mysuru, Karnataka, India 570016, anandm@gsss.edu.in

² Associate Professor, Department of Computer Science and Engineering, GSSS Institute of Engineering and Technology for Women, Mysuru, Karnataka, India 570016, vishweshj@gsss.edu.in

³ Assistant Professor, Department of Electronics & Communication Engineering, GSSS Institute of Engineering and Technology for Women, Mysuru, Karnataka, India 570016, padmar@gsss.edu.in

⁴ Professor and Principal, Department of Education, Onkarmal Somani College of Education, Mysuru, Karnataka, India 570009, ushaprada@gmail.com

⁵ Assistant Professor, Department of Mathematics, GSSS Institute of Engineering and Technology for Women, Mysuru, Karnataka, India 570016 pradeepr.mysore@gmail.com

⁶ Associate Professor, Department of Information Science and Engineering, GSSS Institute of Engineering and Technology for Women, Mysuru, Karnataka, India 570016 chayaneetha@gmail.com

Corresponding Author Mail id: pradeepr.mysore@gmail.com

HOW TO CITE:

Anand M, Vishwesh J, Padma R, Usha R G, Pradeep Kumar R, Chaya P (2026). Real Time Machine Learning based Assistive Communication System Using Mid-Air Gestures and Voice Feedback for Children with Physical Disabilities. International Journal of Special Education, 41(2s), 209-218.

COPYRIGHT STATEMENT:

Copyright: © 2026 Authors.
Open access publication under the

ABSTRACT:

Gesture-based interaction offers an important alternative mode of communication in human-computer interaction, especially for users who cannot conveniently operate conventional input devices such as keyboards. While simple gestures are easy to learn and perform, they are often limited in expressiveness and are not sufficient for text entry. To address this issue, this work presents a mid-air word gesture and voice alert system that recognizes letters written in free space and converts them into speech and alert messages. The proposed system uses an Arduino Uno interfaced with an MPU-6050 accelerometer attached to the user's hand. The accelerometer captures motion data corresponding to hand movements in air. These signals are processed and used to train a machine learning model capable of recognizing user-defined alphabet gestures. Once trained, the system interprets gestures as letters, displays the generated text, and converts it into voice output. In addition, when a sentence is completed using a predefined full-stop gesture, the message can be sent to a nurse through email, while a default SMS notification is generated through a GSM module to alert the nurse. This system is particularly useful for physically challenged users who may find conventional communication methods difficult. By

terms and conditions of the Creative CommonsAttribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).	combining gesture recognition, text generation, speech synthesis, and alert messaging, the work provides a practical assistive communication solution. Keywords: Machine Learning, Gesture Recognition, Air Writing, Voice Alert System, Assistive Communication
---	--

1. Introduction

Human-computer interaction has evolved significantly with the development of natural and intuitive input methods. Among these, gesture-based interaction has gained importance because it allows users to communicate with machines through body movement rather than physical contact with input devices. Gesture systems are especially helpful for individuals with physical disabilities, as they provide an accessible and flexible means of interaction.

Simple motion gestures can be used to trigger predefined commands, but they are often not expressive enough for detailed communication such as text input. To overcome this limitation, the concept of air writing has emerged. Air writing refers to drawing letters or words in free space using hand movements. This technique is useful in environments where keyboards are not practical or where the user is unable to use conventional typing methods.

This work proposes a machine learning-based mid-air gesture recognition system that can be trained to understand gestures made by the user and associate them with specific letters or commands. Unlike conventional gesture-based systems that rely only on fixed predefined motions, the proposed system allows training with multiple samples of the same gesture so that the machine can learn its pattern more effectively.

The hardware consists of an Arduino Uno and an MPU-6050 accelerometer mounted on the user's hand. The accelerometer records motion along the X, Y, and Z axes, and these readings are sent to a computer for further processing. A machine learning algorithm classifies these gesture patterns into corresponding letters. The recognized letters are then displayed as text and converted into speech. Once a complete sentence is entered using a special

termination gesture, the system sends the message to the nurse's email and simultaneously alerts the nurse through GSM-based SMS.

The main objective of this work is to create an affordable and effective assistive communication system for physically challenged users, enabling them to communicate more independently and efficiently.

2. Literature Review

A literature review helps identify the current state of research, existing techniques, and challenges in the field of gesture recognition and assistive communication systems.

Several researchers have contributed to gesture-based text input and recognition systems using machine learning and sensor technologies:

1. Sam Sunny and Sindhu (2020)

This work demonstrates a machine learning-based hand gesture keyboard, enabling text input without physical contact. It highlights improved accessibility but may face challenges in accuracy under varying lighting conditions.

2. Saria Beg et al. (2018)

The paper proposes a vision-based system for recognizing air-written gestures. It effectively converts gestures into text, though performance depends heavily on camera precision and gesture clarity.

3. J. P. Wachs et al. (2017)

This study focuses on depth-image-based hand gesture recognition, improving robustness compared to 2D methods. However, it requires specialized depth sensors, increasing system cost.

4. Bharathi M. et al. (2019)

The authors present a gesture keyboard using machine learning techniques. The system is user-

friendly but may struggle with real-time responsiveness and complex gesture variations.

5. Pengcheng Wang and Junyang Lin (2020)

This research introduces an air-writing tracking system that captures motion trajectories. It shows promising accuracy but requires precise motion tracking mechanisms.

6. Stephane Perrin et al. (2020)

The study utilizes laser-based tracking for gesture recognition, offering high precision. However, the approach may not be practical for widespread use due to hardware complexity.

7. Roland Aigner et al. (2020)

This paper analyzes user preferences in gesture-based interactions, emphasizing usability aspects. It provides valuable insights but does not focus on implementation techniques.

8. L. Anusha and Y. Usha Devi (2020)

The system integrates gesture recognition with voice and language translation. It enhances communication but may face limitations in multilingual accuracy.

9. Chinnam Datta Sai Nikhil et al. (2020)

This work presents a virtual keyboard using finger recognition techniques. It improves interaction efficiency but depends on precise finger detection.

10. Depeng Zhu et al. (2019)

The paper focuses on gesture recognition for military applications. It ensures reliability in critical scenarios but may require high computational resources.

11. Li Zhang et al. (2023)

This study applies Vision Transformers for gesture recognition, achieving higher accuracy than traditional CNNs. However, it demands significant computational power.

12. Ahmed El-Sayed et al. (2023)

The paper combines CNN with MediaPipe for real-time gesture detection. It offers efficient performance but may struggle with occlusions and complex gestures.

13. Priya Sharma and Ankit Verma (2023)

This work uses deep neural networks for HCI applications, improving recognition accuracy. However, model training requires large datasets.

14. K. R. Natarajan and S. Venkatesh (2024)

The hybrid CNN-LSTM model enhances temporal gesture recognition. It improves sequence understanding but increases model complexity.

15. Mohammad Ali Khan et al. (2024)

This research focuses on air-writing recognition using deep learning. It achieves good accuracy but depends on consistent gesture patterns.

16. J. Kim and H. Park (2024)

The study introduces multimodal sensor fusion for gesture recognition, improving robustness. However, integration of multiple sensors raises system cost.

17. Rakesh Kumar and Neha Gupta (2024)

This paper presents a gesture-based virtual keyboard using computer vision. It improves usability but may face challenges in low-light environments.

18. S. Balaji and P. Meenakshi (2025)

The proposed system uses advanced deep learning models for smart interfaces. It enhances accuracy but requires high computational resources.

19. David Lee et al. (2025)

This work explores edge AI for real-time gesture recognition, reducing latency. However, edge devices may limit model complexity.

20. Arjun Patel and Meera Joshi (2025)

The study uses transformer-based models for gesture recognition, offering improved contextual understanding. Yet, it demands large-scale training data.

From the above studies, it is clear that gesture recognition systems have been implemented using image processing, radar, laser tracking, and accelerometer-based approaches. Among these, accelerometer-based systems are attractive because they are low-cost, portable, and less affected by lighting conditions. The present work builds upon

these advantages and combines them with machine learning and communication modules to assist physically challenged individuals.

3. System Analysis and Design

The proposed system integrates sensor hardware, a microcontroller, machine learning software, and communication modules into a single assistive framework.

3.1 System Architecture

The architecture consists of the following major units:

- Input Unit: MPU-6050 accelerometer attached to the user's hand
- Controller Unit: Arduino Uno

- Processing Unit: Computer system running Python and Scikit-learn
- Output Unit: Text display and speech output
- Communication Unit: Email and GSM-based SMS alert system

The accelerometer captures movement in three dimensions. These readings are transferred to the Arduino Uno, which sends the data to the computer through serial communication. The processing unit extracts features from the sensor data and classifies the gesture using a machine learning algorithm. The predicted character is displayed on the screen and converted into speech. When the user completes a sentence, the system sends the message to the nurse and triggers a mobile alert.

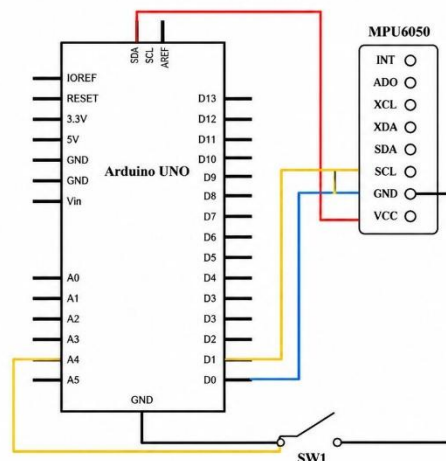


Fig1. System Architecture

4. Hardware Components

4.1 Arduino Uno

Arduino Uno is an open-source microcontroller platform widely used for embedded applications. It provides digital and analog input/output pins that support sensor interfacing and serial communication. In this work, Arduino Uno receives motion data from the accelerometer and forwards it to the computer for classification.

Its key role includes:

- Interfacing with the accelerometer through I2C communication
- Reading sensor data
- Transmitting data to the processing unit

- Supporting real-time operation of the gesture system

4.2 MPU-6050 Accelerometer Module

The MPU-6050 is a MEMS-based motion sensor capable of measuring acceleration along three axes. It can detect tilt, movement, and orientation, making it suitable for hand gesture recognition.

A gesture can be represented as:

$$G = (A_x, A_y, A_z) \quad G = (A_x, A_y, A_z) \quad G = (A_x, A_y, A_z)$$

where A_x , A_y , and A_z represent acceleration values along the X, Y, and Z axes respectively.

To improve feature extraction, each gesture signal can be divided into multiple segments or frames.

These frames help capture the temporal variation of motion. However, increasing the number of frames excessively may lead to overfitting and higher computational complexity. Therefore, an appropriate frame size must be selected experimentally.

4.3 Switch

A switch is used to start and stop gesture recording. When the user presses the switch, the system begins capturing accelerometer data. When the switch is released, transmission stops. This helps isolate meaningful gesture input from random hand movement.

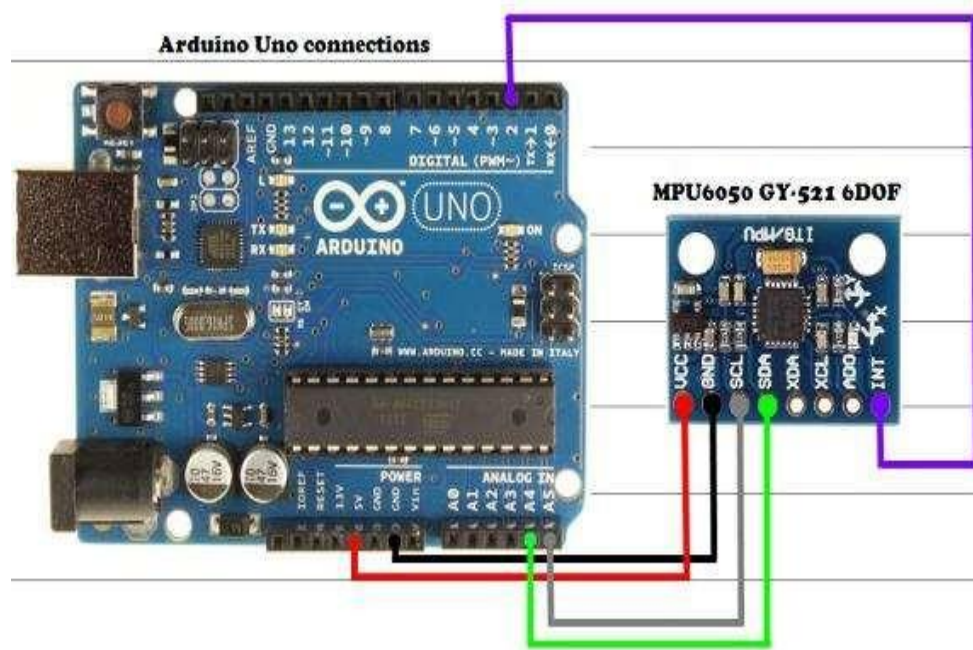


Fig 2: Arduino Uno Connections

4.4. Arduino and Accelerometer Connection

The MPU-6050 communicates with Arduino Uno using the I2C protocol. The connections are as follows:

- VCC to power supply
- GND to ground
- SDA to Arduino analog pin A4
- SCL to Arduino analog pin A5
- AD0 to ground for default I2C addressing

The sensor outputs digital values corresponding to acceleration along the three axes. These values are read by the Arduino and sent to the computer via USB serial communication. The baud rate used in this work is 9600 bps.

4.5. Algorithm for Gesture Recognition

The gesture recognition process follows these steps:

1. Power is supplied to the transmitter module.
2. Arduino Uno initiates communication with the MPU-6050 by sending the slave address.
3. The required sensor registers are configured.
4. Acceleration values for X, Y, and Z axes are read continuously.
5. These values are transmitted to the processing unit through USB.
6. The collected gesture data is stored and preprocessed.
7. Features are extracted from the gesture sequence.
8. A machine learning classifier predicts the corresponding character.

9. The output character is displayed and converted into speech.
10. When a sentence is completed, the message is sent to the nurse's email and a GSM alert is triggered.

5. Dataset Preparation

The first step is to create a dataset containing gesture samples for each character. Each gesture is

recorded multiple times so that the classifier can learn natural variations in user motion.

5.1 Data Preprocessing

The raw sensor data may contain noise and irrelevant variation. Preprocessing techniques are applied to normalize the data, reduce noise, and improve classification quality. Efficient preprocessing also helps handle large datasets and improves training speed.

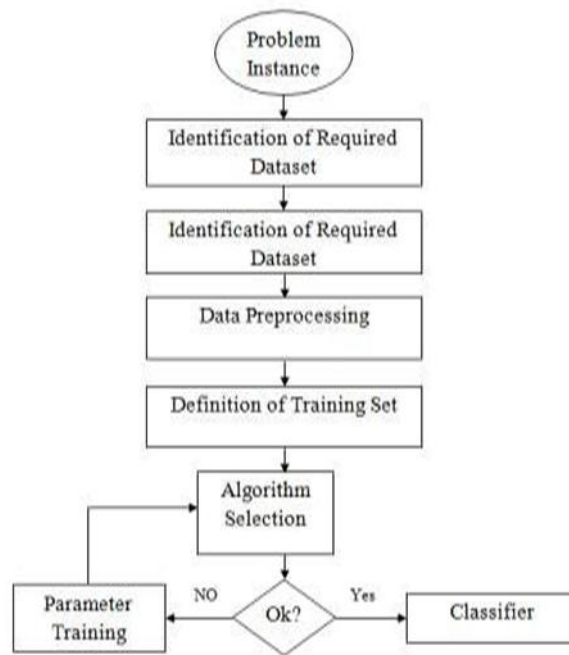


Fig 3: Workflow of the system

5.2 Training Set Formation

The processed gesture samples are grouped into classes, where each class corresponds to a particular letter or symbol. These labeled samples form the training set used to build the classifier.

5.3 Algorithm Selection

The choice of classifier strongly affects system performance. In this work, Support Vector Machine (SVM) is used due to its effectiveness in

classification tasks involving structured numeric features. The dataset is divided into training and testing sets, and the model is evaluated based on classification accuracy.

5.4 Classification and Evaluation

Classification performance is measured using a confusion matrix, which indicates how many test samples are correctly or incorrectly classified. This helps determine the reliability of the gesture recognition model.

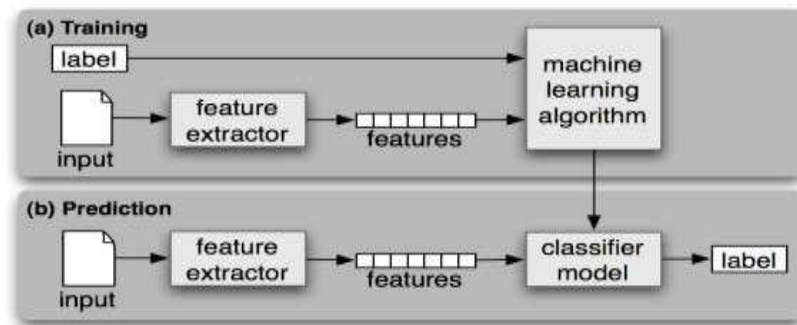


Fig4: Classification and evaluation

5.5 Gesture Dataset

To improve accuracy, the system allows continuous expansion of the training dataset. If the accelerometer position changes or a different user wears the device, the gesture patterns may vary. Therefore, adding more samples over time improves model robustness.

The system supports up to 40 different gestures, each mapped to a letter or command. During data collection:

- The user selects a target gesture label.
- Multiple gesture samples are recorded.
- Each batch is saved separately to avoid overwriting earlier samples.
- The complete dataset is later used to train the classifier.

This incremental training process makes the system more adaptable and personalized.

6. Workflow of the Gesture Keyboard System

Initially, the machine does not know any gesture patterns. The user must first train the system by performing gestures repeatedly. The accelerometer attached to the hand captures motion values continuously, and these values are used to estimate the path of the hand in free space.

The recorded motion data is transformed into feature vectors and mapped to corresponding user-defined letters. Once enough examples are collected, the machine learning classifier can predict letters from new gestures made by the user. For example, if the gesture for the letter “A” has been trained sufficiently, the system will recognize a similar motion and print “A” on the display.

This process allows the machine to learn personalized gesture patterns rather than relying solely on fixed universal gestures.

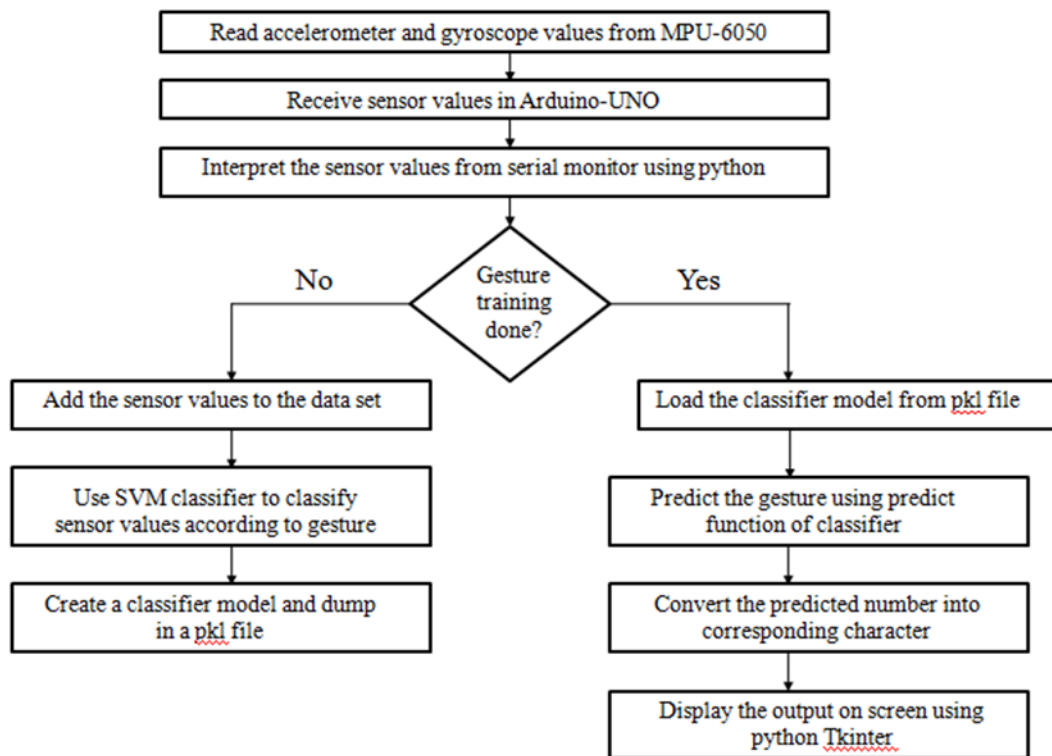
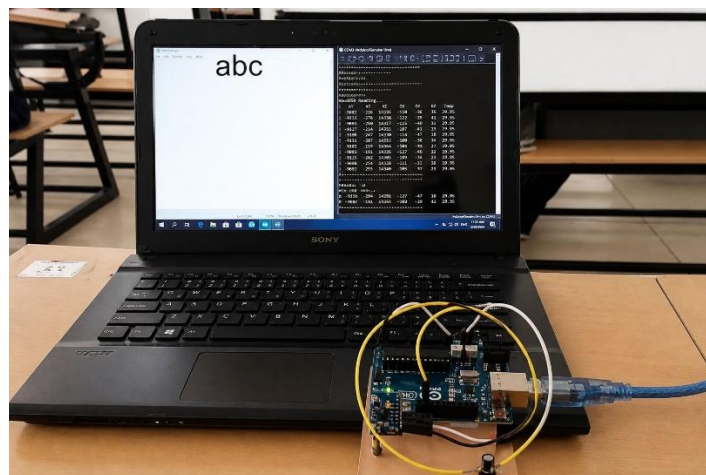


Fig 5: Workflow of the gesture keyboard

7. Results and Discussion

The work was implemented and tested successfully. The system demonstrated the ability to recognize

letters drawn in air through hand gestures using the accelerometer-based input device. The proposed method effectively interprets user motion and maps it to meaningful text output.



The recognized letters can be displayed on the monitor and converted into speech, thereby supporting communication for physically challenged individuals. Once the user finishes a sentence using a predefined punctuation gesture, the message is transmitted to the nurse's email, and an alert SMS is sent.

The work uses only a small number of components, making it cost-effective, portable, and easy to implement. However, the number of gestures that can be recognized accurately is still limited. The system performs best when sufficient training data is provided and the user performs gestures consistently. Overall, the work demonstrates that practical gesture-based communication is achievable using simple hardware and machine

learning techniques. Although the current results are preliminary, they show strong potential for assistive technology applications.

8. Conclusion

This work presents a mid-air word gesture and voice alert system designed to support physically challenged users in communication. By using an accelerometer attached to the hand, the system captures motion data corresponding to letters drawn in air. These gestures are classified using

machine learning and converted into text and speech. The system also extends beyond simple recognition by including communication features such as email transmission and nurse alert notifications through GSM. This makes the work particularly useful in healthcare or assisted living environments, where rapid and accessible communication is critical. The proposed system is low-cost, simple, and effective, and it demonstrates how gesture recognition can be used to improve interaction for users with physical limitations.

References

- Journal Article: 2020, Sam Sunny and Sindhu, "Hand Gesture Keyboard Using Machine Learning," ASIET.
- Journal Article: 2018, Saria Beg, M. Fahad Khan, and Faisal Baig, "Visual Gesture Recognition for Text Writing in Air," IEEE
- Journal Article: 2017, J. P. Wachs, M. Kölsch, H. Stern, and Y. Edan, "Hand Gesture Recognition with Depth Images," IEEE.
- Journal Article: 2019, Bharathi M., Bhavani R., Keerthana Priya B., and R. Paavaikarasi, "Implementation of Gesture Keyboard Using Machine Learning," IJRSET.
- Journal Article: 2020, Pengcheng Wang and Junyang Lin, "A Gesture Air Writing Tracking System," IEEE.
- Journal Article: 2020, Stephane Perrin, Alvaro Cassinelli, and Masatoshi Ishikawa, "Gesture Recognition Using Laser-Based Tracking System," IEEE.
- Journal Article: 2020, Roland Aigner, Daniel Wigdor, and Hrvoje Benko, "A Study of Human Preferences in Usage of Gesture Types for HCI," MSR-TR.
- Journal Article: 2020, L. Anusha and Y. Usha Devi, "Implementation of Gesture Based Voice and Language Translator," ASIET.
- Journal Article: 2019, Chinnam Datta Sai Nikhil, Chukka Uma Someswara Rao, and T. Anandhi, "Finger Recognition and Gesture Based Virtual Keyboard," IEEE
- Journal Article: 2020, Depeng Zhu, Weida Zhan, Ranran Wei, and Ziqiang Hao, "Individual Soldier Gesture Intelligent Recognition System," IEEE.
- Journal Article: 2023, Li Zhang, Yifan Chen, and Rui Wang, "Deep Learning-Based Hand Gesture Recognition Using Vision Transformers," IEEE Access.
- Journal Article: 2023, Ahmed El-Sayed, Noor Alshammari, and Fatima Hussain, "Real-Time Hand Gesture Recognition System Using CNN and MediaPipe," IEEE.
- Journal Article: 2023, Priya Sharma and Ankit Verma, "AI-Based Gesture Recognition for Human-Computer Interaction Using Deep Neural Networks," Springer.
- Journal Article: 2024, K. R. Natarajan and S. Venkatesh, "Vision-Based Hand Gesture Recognition Using Hybrid CNN-LSTM Model," IEEE.
- Journal Article: 2024, Mohammad Ali Khan, Sara Ahmed, and Imran Raza, "Real-Time Air Writing Recognition Using Deep Learning Techniques," IEEE Access.
- Journal Article: 2024, J. Kim and H. Park, "Robust Hand Gesture Recognition Using Multimodal Sensor Fusion and Machine Learning," IEEE Sensors Journal.
- Journal Article: 2024, Rakesh Kumar and Neha Gupta, "Gesture-Based Virtual Keyboard Using Computer Vision and AI," Elsevier.

Journal Article: 2025, S. Balaji and P. Meenakshi, "Advanced Gesture Recognition System for Smart Interfaces Using Deep Learning," Springer.

Journal Article: 2025, David Lee, Kevin Brown, and Emily Clark, "Next-Generation Gesture Recognition Using Edge AI and Real-Time Processing," IEEE.

Journal Article: 2025, Arjun Patel and Meera Joshi, "AI-Driven Gesture Recognition for Interactive Systems Using Transformer Models," IEEE Access.