ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

AI Based Real-Time Air Gesture Recognition & Drawing System

Nayana Prashanth[1]; Dr. Prathapchandra[2]

[1]Sri Krishna Institute of Technology, Bangalore, Department of ISE,
[2]Shri Dharmasthala Manjunatheshwara Institute of Technology, Ujire, Department of ECE, Karnataka

Publication Date: 2025/07/15

Abstract: The AI Air Gesture Draw seeks to revolutionize human-computer interaction by removing the dependence on physical input devices. This cutting-edge system allows users to draw in mid-air using intuitive hand gestures, which are captured in real time by a standard webcam and interpreted through sophisticated deep learning models. Utilizing Google's QuickDraw dataset and implemented with TensorFlow, the system is trained to accurately recognize a diverse array of hand-drawn patterns. By integrating computer vision and gesture recognition, it provides a seamless and touchless drawing experience. With significan t applications in education, accessibility, and digital design, this technology creates new opportunities for inclusive and intuitive interfaces, particularly aiding users with limited mobility or those in settings where touch-based input is unfeasible.

Keywords: *Air Gesture Recognition, Artificial Intelligence, Convolutional Neural Network, Recurrent Neural Network, Human-Computer Interaction, Deep Learning, TensorFlow.*

How to Cite: Nayana Prashanth; Dr. Prathapchandra (2025) AI Based Real-Time Air Gesture Recognition & Drawing System. *International Journal of Innovative Science and Research Technology*, 10(7), 719-725.
https://doi.org/10.38124/ijisrt/25jul230

I. INTRODUCTION

Human-computer interaction (HCI) is undergoing a paradigm shift as emerging technologies strive to create more natural, intuitive, and accessible interfaces. Conventional input devices, such as keyboards, mice, and touchscreens, although efficient, impose physical limitations and may restrict engagement for individuals with mobility disabilities or in hands-free settings. In response to these limitations, gesture- based interaction has gained prominence as a touchless alternative that enhances user experience and accessibility.

The AI-Air-Gesture-Draw introduces an innovative approach to gesture-based input, enabling real-time, contactless drawing through hand movements captured via a standard webcam. This system leverages the power of AI, computer vision, and deep learning to interpret mid-air gestures as digital sketches, eliminating the need for physical input devices. Utilizing TensorFlow and trained on Google's QuickDraw dataset, the model employs Convolutional Neural Networks (CNNs) facilitate the extraction of spatial features, while Recurrent Neural Networks (RNNs) enable the analysis of temporal sequences, thereby ensuring accurate and adaptable gesture detection. [2].

A Docker-based deployment environment ensures platform independence and streamlined installation, facilitating broader adoption across varied application domains. The suggested approach enhances digital art and design while providing substantial possibilities in accessibility technologies, gesture-controlled interfaces in AR/VR and gaming, smart home interactions, and educational applications.

This paper presents the system design, implementation details, and performance evaluation of the AI-Air-Gesture-Draw, demonstrating its effectiveness as a robust and scalable solution for real-time, touchless human-computer interaction.

II. ARCHITECTURE

Figure 1 depicts the structure of the intended AI-driven system for painting with air gestures, outlining the sequential flow of processes. The system initiates with a webcam feed that captures real-time hand gestures performed by the user in mid-air. These gestures serve as the primary input for the drawing interface.

The incoming video stream is processed with OpenCV, where each frame is retrieved and undergoes several preprocessing stages. This includes conversion to grayscale and resizing to a standardized 28×28-pixel resolution [1][3]. These transformations enhance gesture visibility, reduce computational complexity, and eliminate background noise, thereby improving the model's prediction accuracy.

ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

The preprocessed image frames are then input into a Convolutional Neural Network (CNN) model specifically trained for gesture recognition. CNN comprises multipl layers designed to extract spatial features, identify gesture patterns, and classify the input with high accuracy.

Once the CNN processes the input, it outputs a predicted gesture label, which can be either displayed on the user interface or used to trigger corresponding system actions. This closed-loop interaction ensures an intuitive and responsive user experience.

To further support system design and user interaction, a case diagram is employed. This diagram outlines the functional relationships between the user and system components, providing a visual overview of the gesture recognition pipeline and its responses. It helps with mapping the user's hand gesture inputs into the system’s prediction and output mechanisms, thereby enhancing comprehension of the system’s operational flow.

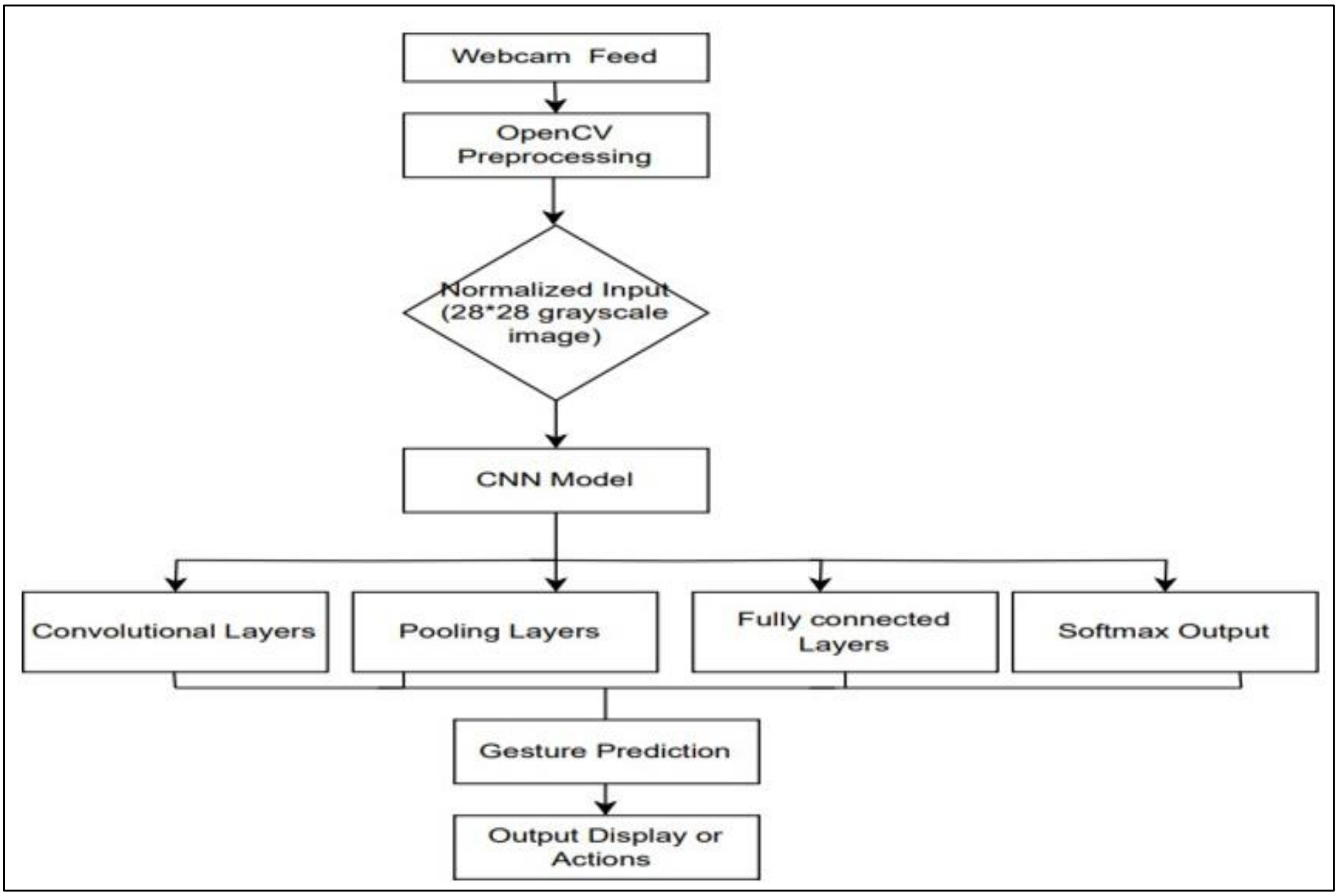


Fig1 Architecture Diagram of Air Gesture Draw

Figure 2 illustrates the use case diagram of the AI Air Gesture Drawing system, providing a structured overview of user interactions and system functionalities. This diagram is essential for modeling the system's behavioral features, especially the interaction between the user and other system components.

The primary user initiates the interaction by performing hand gestures in mid-air. These gestures are captured in real time via a webcam, which serves as the system’s input device. The acquired frames are then subjected to a preprocessing pipeline implemented using OpenCV. This stage involves converting the images to grayscale and resizing them to a fixed dimension, optimizing the input for further processing by the recognition model.

After preprocessing, the frames are sent to a convolutional neural network (CNN), which is responsible for recognizing the movements based on the features it has learned. CNN examines spatial patterns and generates gesture predictions with a high level of precision.

These predicted gestures are then interpreted by the system and translated into corresponding drawing commands or functional actions, which are dynamically rendered on the screen. This interactive feedback loop enhances the usability and intuitiveness of the system.

The use case diagram serves as a visual aid to understand the flow of interactions within the system. It delineates how the user's gestures are processed and converted into meaningful outputs, thereby aiding both system designers and end users in understanding the operational flow of gesture recognition and response generation.

ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

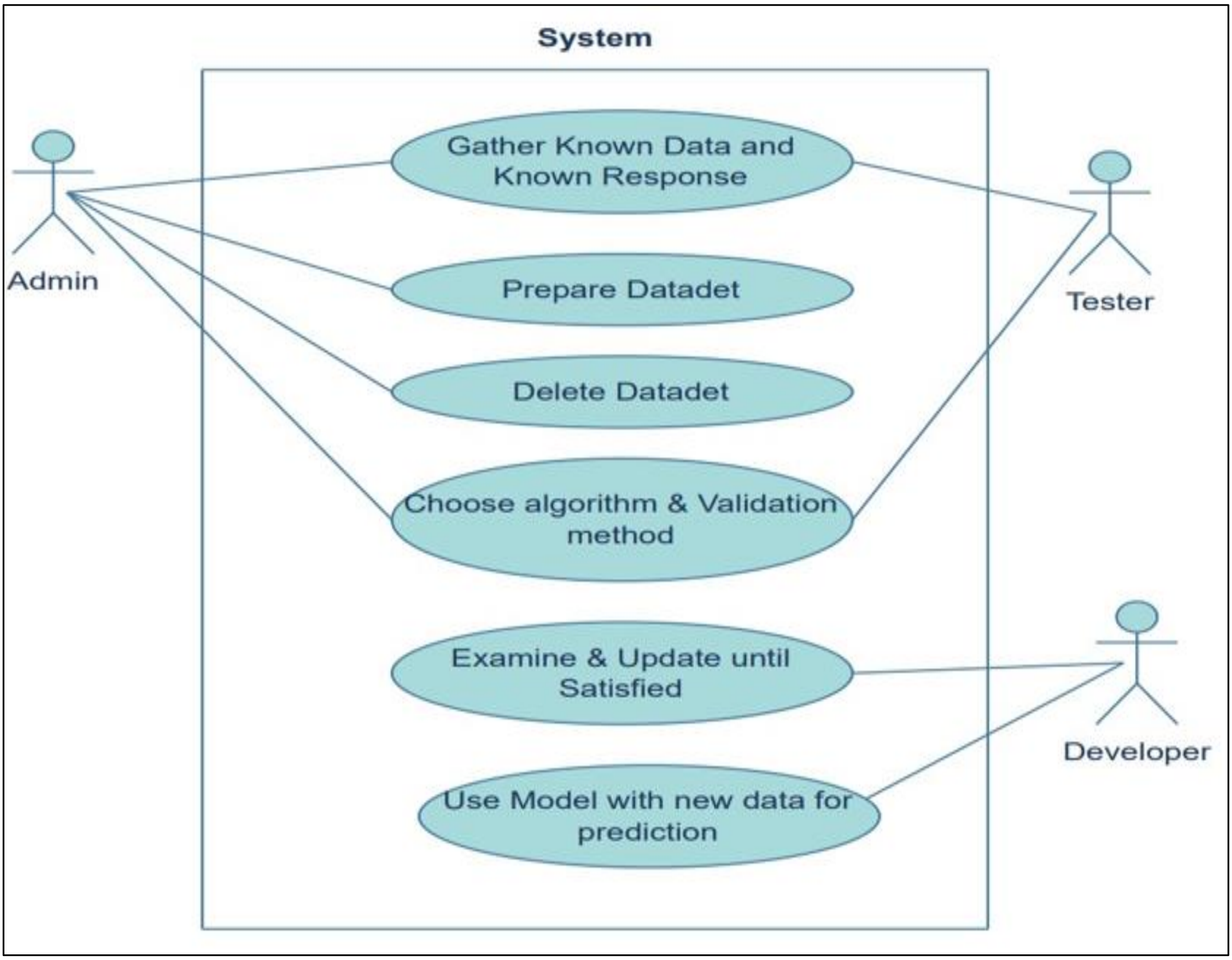


Fig 2 Use Case Diagram of AI Air Gesture Draw

III. SEQUENCE DIAGRAM

Figure 3 presents the sequence diagram of the AI Air Gesture Drawing system, illustrating the chronological flow of operations involved in real-time gesture-based interaction. The diagram provides a comprehensive view of how various system components communicate and process data to deliver a seamless user experience.

The workflow begins with the user performing hand gestures in mid-air, which are captured continuously via a webcam. These frames serve as the primary input for the system. Once captured, the frames are directed to a preprocessing module based on OpenCV. This module converts the images into grayscale format and resizes them to a standardized resolution, optimizing them for subsequent analysis by the deep learning model.

The processed frames are then passed to a Convolutional Neural Network (CNN) that has been trained on the QuickDraw dataset. This dataset contains a wide variety of human-drawn sketches, enabling CNN to learn and generalize gesture patterns effectively. Based on this training, CNN classifies the current input gesture and generates a prediction corresponding to a specific shape or action.

The predicted gesture is then mapped to a visual representation or functional command, which is rendered on the screen in real time. This dynamic visualization allows users to draw or interact with digital content without any physical contact, making the system intuitive and user-friendly.

In an extended use case scenario, the system is also designed to handle machine learning tasks such as medical image analysis. For instance, users can upload a lung tissue dataset, which is processed through a Data Preparation Module. This prepared data is then forwarded to a Training Module, where a machine learning model is trained. Progress and results are communicated back to the user, enabling both gesture-based interaction and AI-driven analytics within the same framework.

This sequence diagram provides a clear and structured view of the entire system process—from getting input to showing output—highlighting gesture detection and additional machine learning features.

ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

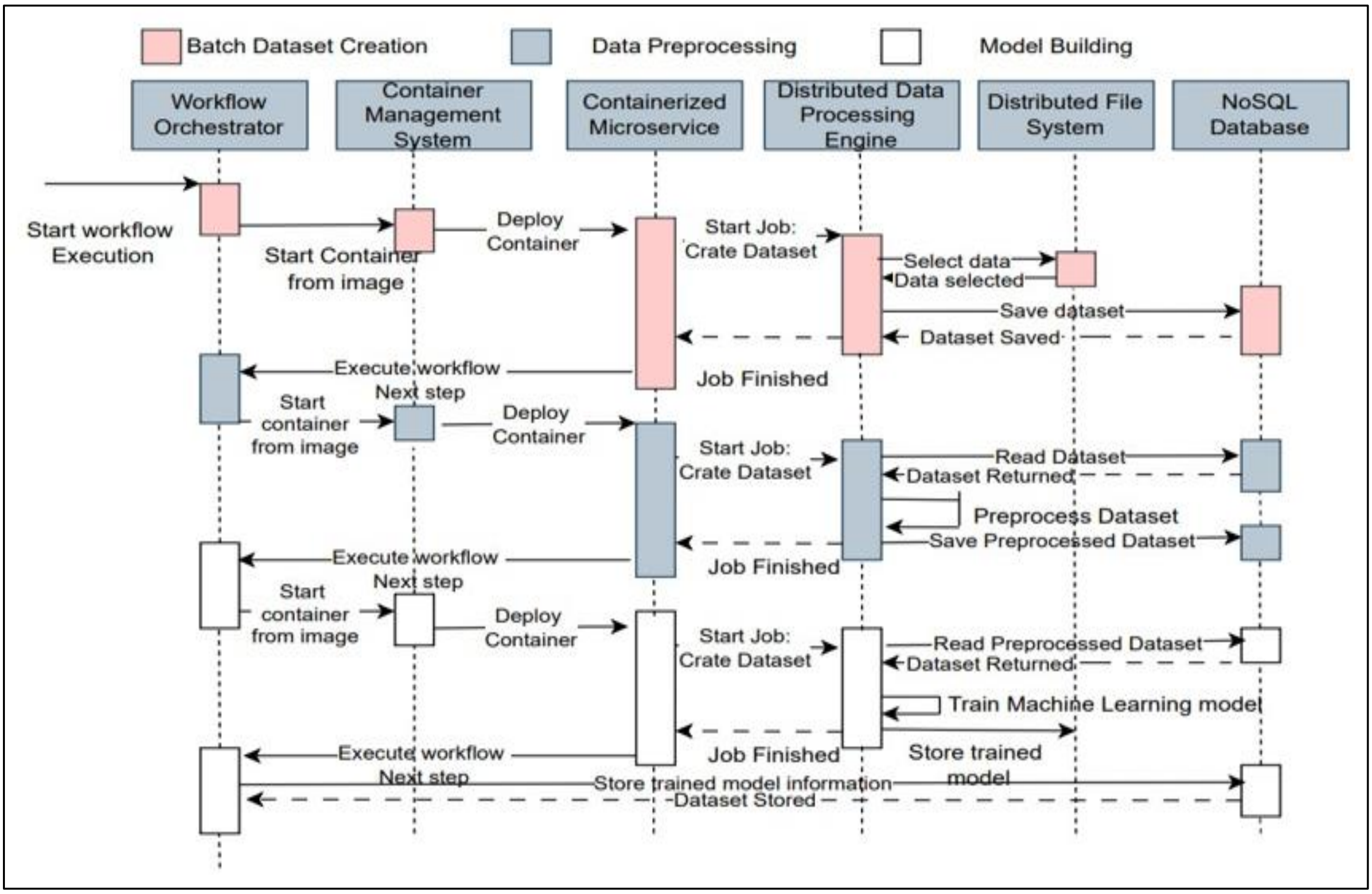


Fig 3 Sequence Diagram of Air Gesture Draw Workflow

IV. ALGORITHM OF RNN

Recurrent Neural Networks (RNNs) are specifically designed for processing sequential input, such as time series data, text, and gesture sequences. Unlike traditional feedforward neural networks, RNNs retain a memory of past information through connections that loop back, enabling the model to incorporate earlier inputs when producing current outputs. This modeling of temporal dependencies is essential for tasks where context over time is important. At each time step t, the RNN processes the input x_t and updates its hidden state h_t based on the current input and the previous hidden state h_{t-1}. This iterative approach allows the network to capture temporal dependencies effectively. However, standard RNNs often encounter difficulties in learning connections over long durations due to challenges such as vanishing or exploding gradients during training.

To address these limitations, advanced architectures like Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) have been introduced. LSTM networks utilize specialized memory cells and gates (input, forget, and output gates) to regulate the flow of information, which aids the model in retaining significant details over extended sequences. GRUs simplify this architecture while preserving the ability to efficiently capture long-term dependencies.

Figure 4 illustrates the architecture of an LSTM cell, highlighting the interactions between inputs, hidden states, and cell states over time.

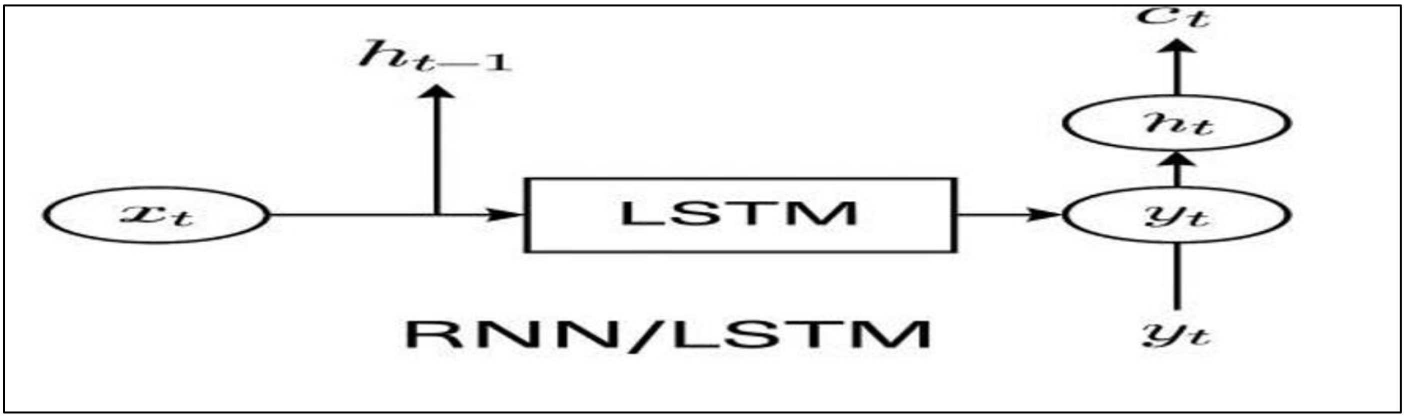


Fig 4 Schematic representation of an LSTM unit

ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

The proposed gesture recognition system incorporating an RNN/LSTM framework follows the steps below:

- *Input Acquisition:*
  Capture sequential input data (e.g., gesture point series or frame features) using a webcam.

- *Preprocessing:*
  Normalize and reshape the data to fit the input requirements of the LSTM layer.

- *Sequence Modeling:*
  Feed the sequential data into one or more LSTM or GRU layers to capture temporal dependencies and retain contextual memory across frames.

- *Regularization:*
  Apply dropout layers to reduce overfitting during training.

- *Classification:*
  Use fully connected dense layers with SoftMax activation to predict the gesture label.

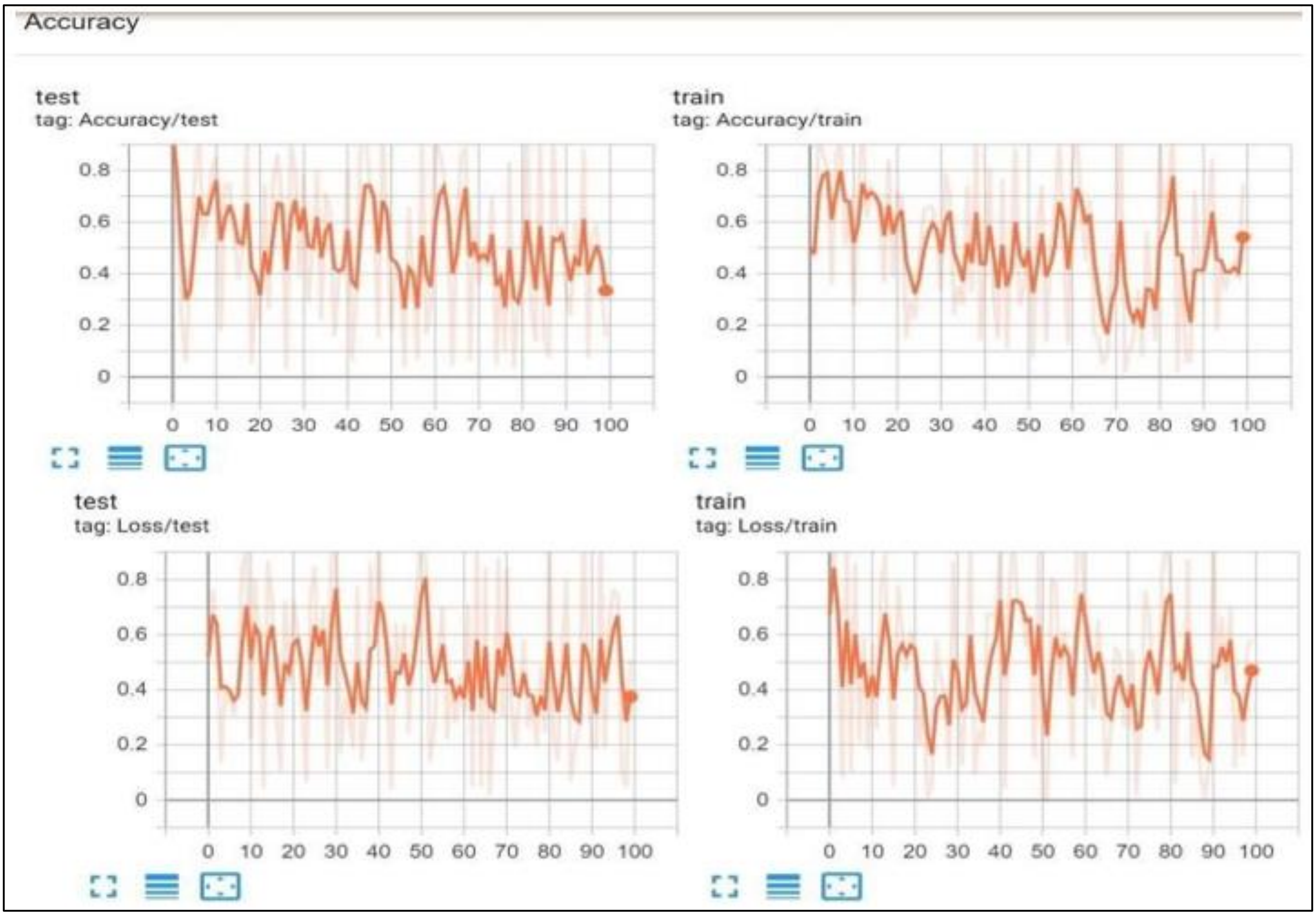
- *Output Mapping:*
  Map the predicted label to a specific drawing command or system action, enabling real-time interaction.

V. RESULTS AND DISCUSSIONS

The efficacy of the proposed AI Air Gesture Drawing system was assessed utilizing a test set of hand gesture inputs under varying conditions. The model demonstrated robust performance in recognizing well-defined gestures captured under optimal lighting and background conditions. An average classification accuracy of 90% was achieved, indicating the system's effectiveness in learning and predicting gesture patterns accurately. Figure 5 illustrates the accuracy and loss for both training and validation phases.

Further performance metrics, including precision and recall, remained consistently high, affirming the model's capability to minimize both false positives and false negatives. These results confirm that the system can reliably classify most gesture inputs with a high degree of confidence in controlled environments.

However, the system's performance declined when exposed to more challenging inputs, such as noisy, incomplete, or low- resolution images. Under such suboptimal conditions,classification accuracy dropped to 60–75%, highlighting the limitations of the current preprocessing pipeline and the need for more advanced noise-handling techniques. These findings suggest that although the model generalizes well within the trained distribution, its robustness to diverse or degraded inputs must be enhanced.



ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

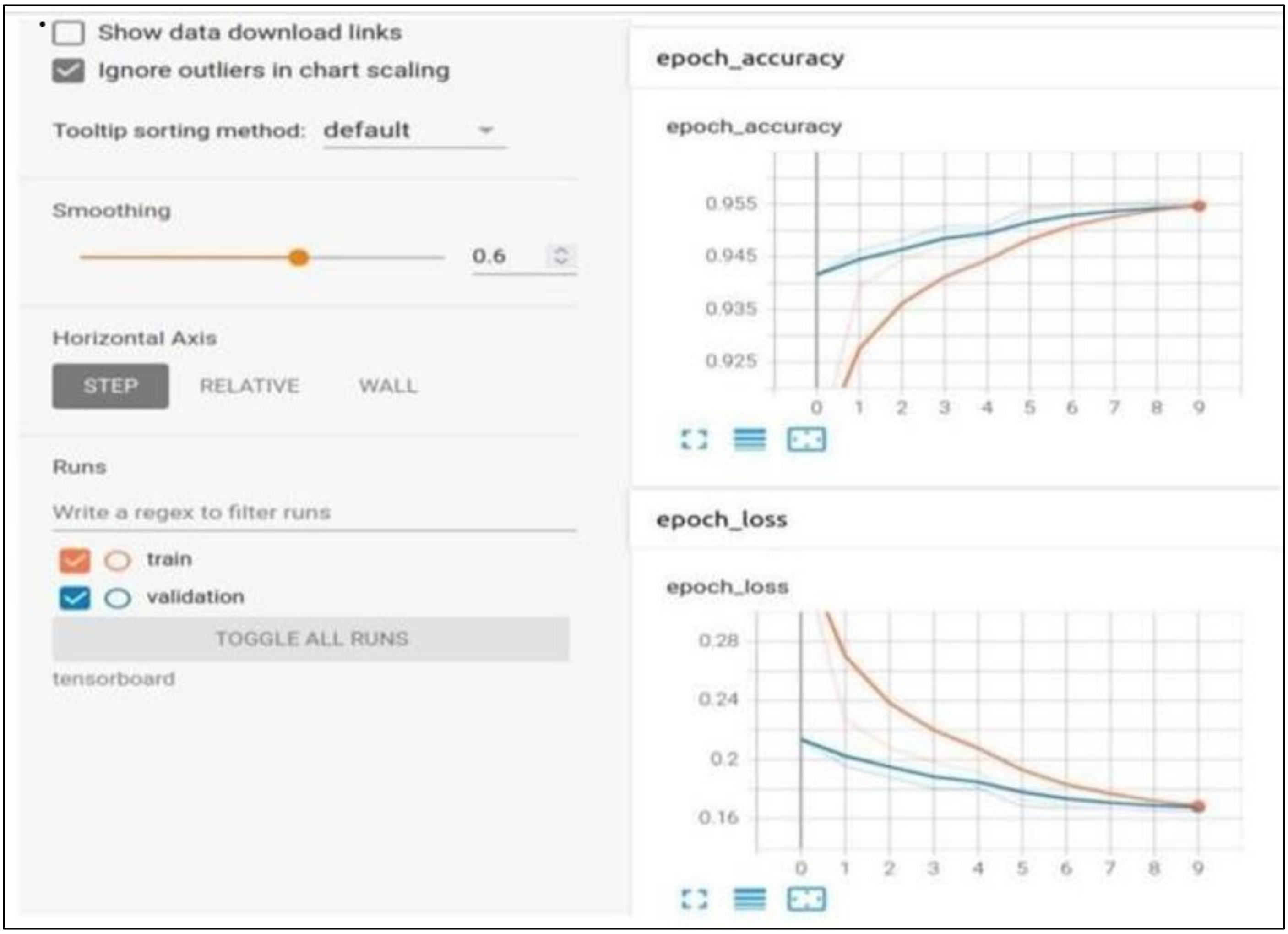


Fig 5 Training and Validation Accuracy and Loss

The system's response time was measured between 100 and 150 milliseconds, supporting real-time interaction without noticeable latency. This confirms the feasibility of deploying the model in interactive environments where immediate feedback is critical.

The efficacy of the recurrent neural network (RNN) model was statistically assessed utilizing conventional classification criteria, such as precision, recall, F1-score, and accuracy. These metrics offer a thorough evaluation of the model's proficiency in accurately recognizing gesture classes while reducing both false positives and false negatives. The comprehensive evaluation findings are encapsulated in Table 1, illustrating the efficacy of the suggested methodology in sequential gesture recognition tasks.

- Macro Average: Precision: 0.76 | Recall: 0.78 | F1- Score: 0.77
- Weighted Average: Precision: 0.89 | Recall: 0.87 | F1- Score: 0.82
- Accuracy: 85%

Table 1 Performance of Model

Model	Precision	Recall	F1 Score	Accuracy
RNN	83%	82%	82.5%	85%

Table 2 Performance Metrics of RNN Model for Air Gesture Draw

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP): High (Majority gesture recognized)	False Negative (FN): minimal
Actual Negative	False Positive (FP): Low	True Negative (TN): High

VI. CONCLUSION

Presented a real-time AI Air Gesture Drawing system that integrates computer vision and deep learning methodologies for natural, frictionless human-computer interaction. The system exhibited exceptional accuracy and responsiveness in optimal conditions, confirming the efficacy of its architecture and model design.

Despite its promising performance, challenges remain in ensuring consistent accuracy under diverse real-world conditions. Future work will focus on improving

ISSN No: 2456-2165 https://doi.org/10.38124/ijisrt/25jul230

preprocessing robustness, increasing model adaptability, and expanding the dataset to include more varied gesture types and environmental scenarios. Incorporating more advanced architectures such as attention mechanisms or transformer models may further enhance performance in complex settings.

Overall, the system provides a solid foundation for gesture- based interfaces and holds potential for broader applications in education, accessibility, and human-computer interaction systems.

REFERENCES

[1]. N. Hendy, H. M. Fayek, and A. Al-Hourani, "Deep Learning Approaches for Air-Writing Using Single UWB Radar" IEEE Sensors J., vol.22, no. 12, Jun. 2022.

[2]. S. Ahmed, D. Wang, J.-Y. Park, and S. H.Cho, "UWB- Gestures, A Public Dataset Of Dynamic Hand Gestures Acquired Using Impulse Radar Sensors" Data Science, vol. 8,

[3]. 2021.

[4]. Das, A., Gawde, S., Suratwala, K., Kalbande, D. "Sign Language Recognition Using Deep Learning on Custom Processed Static Gesture Images". International Conference on Smart City and Emerging Technology (ICSCET), (2018).

[5]. V. K. Verma, S. Srivastava, and N. Kumar, "A Comprehensive Review on Automation of Indian Sign Language," IEEE Int. Conference, Advanced Computer Engineering Application, Mar 2015.

[6]. Saira Beg, M. Fahad Khan and Faisal Baig, "Text Writing

[7]. in Air", Journal of Information Display, vol 14, 2013.

[8]. R. Z. Khan, "Comparative Study of Hand Gesture Recognition System" Computer Science and Information Technology (CS & IT), vol 4, 2012.