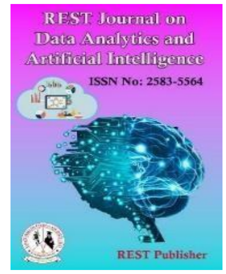
REST Journal on Data Analytics and Artificial Intelligence
Vol: 4(1), March 2025
REST Publisher; ISSN: 2583-5564
Website: http://restpublisher.com/journals/jdaai/
DOI: https://doi.org/10.46632/jdaai/4/1/59



Hand Gesture Recognition

***Archana, M Sahithi Reddy Gosika, Kavya Sree, D Karthik**
Anurag University, Hyderabad, Telangana, India
**Corresponding Author Email: 21eg107a53@anurag.edu.in*

Abstract. *The "Real-Time Hand Gesture Recognition System" aims to enhance human-computer interaction using computer vision and machine learning techniques. Using the webcam for real-time video capture, the system uses OpenCV to process each frame in real-time. Thus, it recognizes hand gestures using models such as Cvzone's, Hand Detector or Mediapipe. After the hand is detected, it crops, resizes, and normalizes it for further review. Then, a pre-trained Convolutional Neural Network (CNN) model is used to classify the gesture, mapping that into predefined labels such as numbers, letters, or ASL signs. This therefore provides an efficient and natural interface for the users in particular with assistive technology applications.*

Keywords: *Hand Gesture Recognition, Machine Learning (ML), Convolutional Neural Network (CNN), OpenCV, Real-time Processing, Feature Extraction, Image Preprocessing, Human-Computer Interaction (HCI), Assistive Technology, Gesture Classification, Deep Learning, Python, TensorFlow, Keras, Computer Vision, Smart Interfaces, Non-contact Interaction, Automated Control Systems.*

1. INTRODUCTION

Hand gesture recognition is becoming increasingly important in enhancing human-computer interaction (HCI), particularly in applications such as assistive technologies, gesture-based controls, and communication systems for individuals with disabilities. Traditional input methods like keyboards, mice, or touchscreens can be limiting, especially for people with physical challenges. Moreover, these systems lack the intuitive, natural interaction provided by vision-based gesture recognition. Therefore, a more efficient, accessible, and intuitive solution is necessary to address these challenges. This project focuses on developing a Real-Time Hand Gesture Recognition System that leverages computer vision and machine learning to identify and interpret hand gestures. By using OpenCV for real-time video processing and a pre-trained Convolutional Neural Network (CNN) for gesture classification, the system allows users to interact with devices using simple hand gestures. This contactless approach improves both usability and accessibility, providing a seamless user experience without the need for traditional input devices. The system is designed to recognize static gestures, such as those in American Sign Language (ASL), and can be expanded to include dynamic gestures. By incorporating models like cvzone's HandDetector or Mediapipe for hand detection and tracking, the project ensures robust performance in real-time environments. Python libraries, deep learning frameworks like TensorFlow or Keras, and intuitive design allow the system to offer reliable and accurate gesture recognition.

2. LITERATURE SURVEY

TABLE 1.

Author(s)	Title	Published	Proposed System
Devira Anggi, Maharani Hanif, Fakhrurroja Riyanto, Carmadi Machbub	Hand Gesture Recognition using K-Means clustering and SVM	2018 School of Electrical Engineering and Informatics	Illustrates the overall structure system for skeletal tracking and processing
Vina Ayumi	Performance evaluation of support vector machine algorithm for human gesture recognition	2020, International Journal of Scientific Research in Science Engineering and Technology	A wearable sensor-based data detection with camera setup and analysis
Meng Wu	Gesture Recognition based on Deep Learning	2024, School of Physics and Information Engineering	Uses deep learning techniques for hand detection

Existing System

Hand gesture recognition systems use a variety of techniques to identify hand movements, including Sensor-based systems. Use depth sensors and leap motion sensors to detect the position and orientation of the hand, as well as the number and position of fingers. The system then uses image processing to identify patterns in the hand's shape and position. Some challenges of using hardware-based systems include: The hardware can be expensive, methods can be limited, and The hardware can be sensitive to user motions.

Limitation of Existing System

- High Sensitivity to Environmental Conditions: Accuracy drops with changes in lighting or complex backgrounds.
- Hardware Limitations and Costs: Specialized hardware is expensive and can be prone to wear and tear.
- Accuracy and Robustness Issues: Real-time recognition struggles with varied gesture speeds, angles, and user differences.
- User Fatigue: Continuous gesture use can lead to physical fatigue, reducing system usability.
- Limited Gesture Vocabulary: Expanding the gesture set can cause confusion between similar movements.
- Privacy and Security Concerns: Camera-based systems may raise privacy issues due to continuous image capture.
- Calibration and Setup Requirements: Systems often need time-consuming calibration and frequent adjustments.

3. PROBLEM STATEMENT

Existing hand gesture recognition systems face significant challenges with two main issues: sensitivity to diverse environments and limited gesture options. The systems are often limited by factors such as sensitivity to lighting changes, complex backgrounds, and the need for expensive hardware like depth sensors, infrared sensors, or wearable devices In addition, long periods of usage of a hand gesture recognition system can make the users tired, while expanding the gesture vocabulary may introduce ambiguity. Moreover, the existing systems involve complex configuration processes that make it hard for users to use the technology satisfactorily. Hence, it's needed to make a simpler, quicker, and friendlier system in hand gesture recognition that uses routine hardware, including accuracy and flexibility improvement, reduction of physical effort, and resolution of privacy issues, which would then make the interaction smoother. This project aims to address these limitations by developing a more robust, cost-effective, and user-friendly hand gesture recognition system that ensures high accuracy and responsiveness in real-world applications.

4. PROPOSED SYSTEM

Input Module:

- Video Capture: Captures live video stream from a webcam using OpenCV (cv2.VideoCapture).
- Preprocessing: Frames are processed in real-time, and hands are detected and cropped using the hand detection model (e.g., cvzone, HandDetector).

Processing Module:

- Hand Detection: A Convolutional Neural Network (CNN) or a pre-trained model is used to detect the hand and extract its bounding box from the video frame.
- Image Normalization: The cropped hand is resized to a standard size (e.g., 300x300 pixels) while maintaining the aspect ratio and placed on a white background.

Classification Module:

- Gesture Classification: A machine learning model (e.g., CNN) is used to classify the hand gesture. The model outputs the predicted gesture and a corresponding label.
- Post-processing: The recognized gesture is displayed in real-time on the screen using cv2.putText().

Output Module:

- Real-time Feedback: Displays the bounding box and the predicted gesture label on the live video feed.
- Gesture Saving: Users can press a key to save the processed hand image with a unique timestamp.

Hand Gesture Recognition Algorithm

- Hand Detection: Detect the hand in each frame using the hand detection model (cvzone.
- HandDetector). If a hand is detected, extract the bounding box (x, y, width, height) for further processing.
- Video Capture: Initialize the video stream using the default webcam.
- Image Preprocessing: Crop the detected hand from the frame. Resize the hand image while maintaining the aspect ratio, and place it onto a 300x300 white background.
- Gesture Classification: Feed the processed hand image into a pre-trained CNN classifier. The model returns the predicted gesture and its label.
- Real-time Feedback: Display the recognized gesture on the screen along with a bounding box around the detected hand.
- Save Gesture Image: On a specific key press, save the hand image with a unique timestamp to a designated folder.

Methods

Convolutional Neural Network (CNN):

- A CNN is used for gesture classification, which is ideal for processing images due to its ability to capture spatial hierarchies in data.
- Input is the pre-processed hand image.
- Output is the Gesture label (e.g., "Thumbs Up," "Okay").

Preprocessing and Normalization:

- Preprocessing involves cropping the detected hand, resizing the image, and placing it on a fixed background to standardize the input for the classifier.

Hand Detection:

- Hand detection is performed using cvzone. HandDetector or a similar model capable of identifying hands in real time from video input.

Real-time Feedback:

- Visual feedback is provided by overlaying the gesture label on the live video feed, making the system interactive and user-friendly.

5. DESIGN

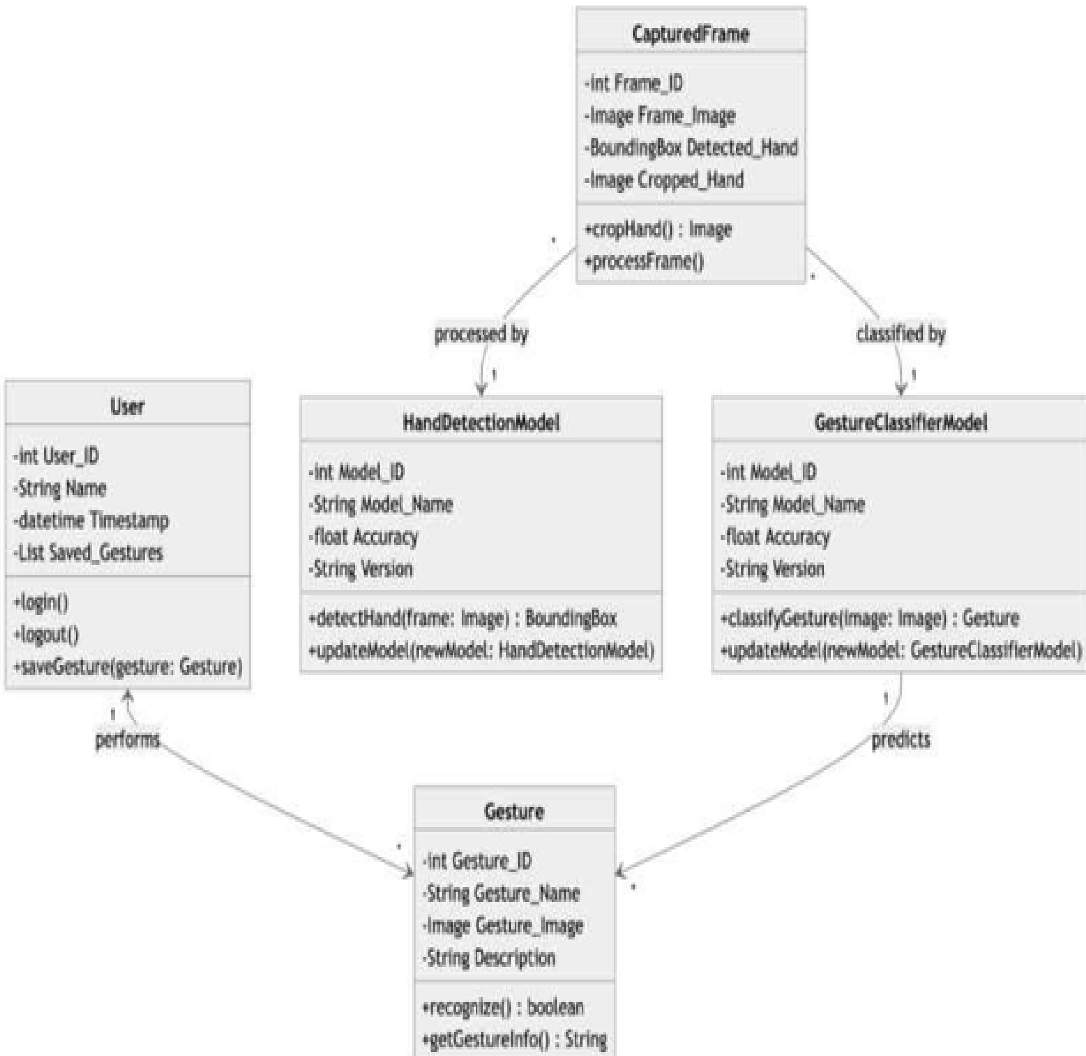


FIGURE 1. DFD

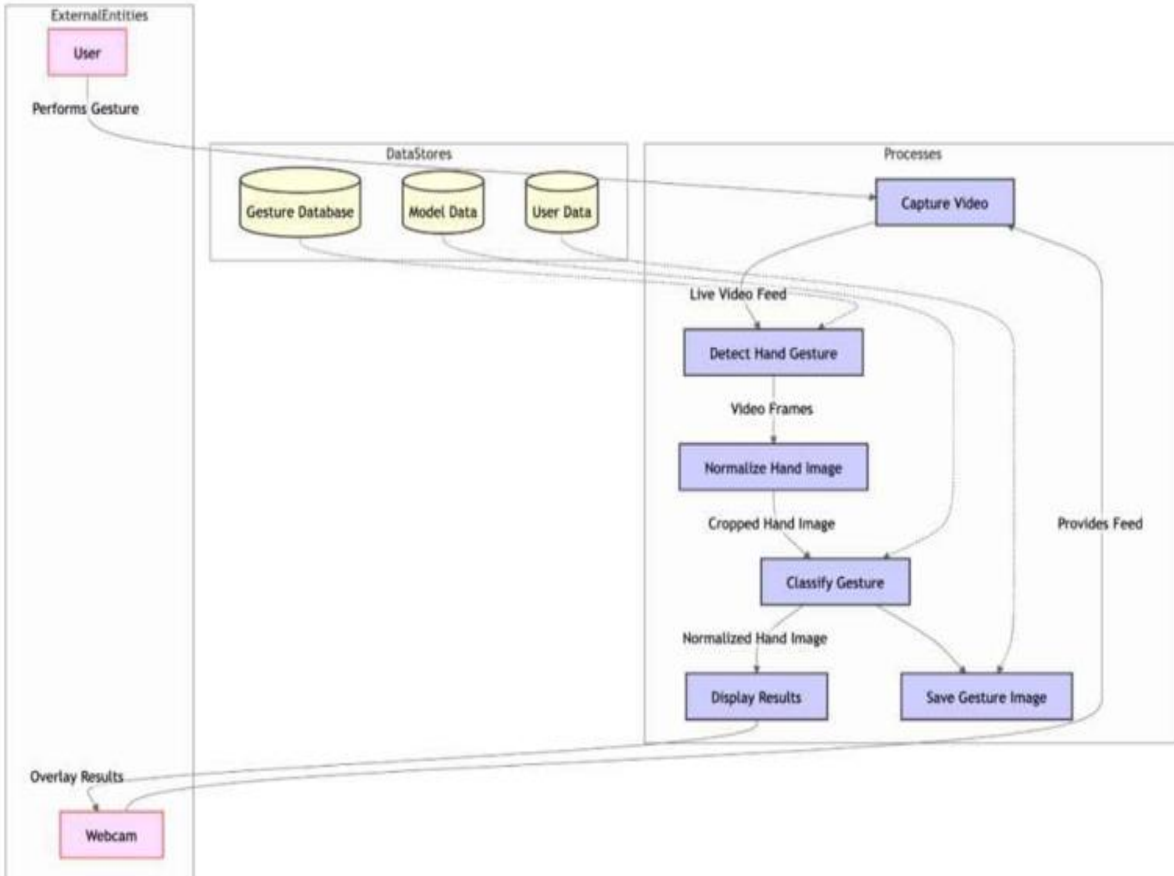


FIGURE 2. UML

6. IMPLEMENTATION & TESTING

Technology Used

- Hand Gesture Recognition System depends on various technologies, which range from hand detection to gesture classification and further on to user interaction. The following technologies are used in it.
- Programming Language: Python 3. x, famous for its vast libraries, to support wide-ranging applications in computer vision and machine learning.
- Hand Detection Library: OpenCV, with the capabilities to do real-time image and video capture, and hand detection.
- Gesture Detection Library: cvzone, the library is used for hand tracking and gesture recognition by making the process easier.
- Machine Learning Framework: This project will use either TensorFlow or Keras to build and train the CNN used for gesture classification.

Image Processing Libraries:

NumPy: The library is meant for efficient array manipulation along with numerical operations on images.

- Pillow: Image processing operations like resizing and formatting
- User Interface Library: Tkinter or similar framework for implementing a user interface with the system.
- Real-Time Feedback: OpenCV functions that use drawing, including putting gesture labels and bounding boxes in the live video feed.
- Database: SQLite or similar lightweight database for storing gesture data, logs, and user interaction.

Procedures

Here are the procedures for your Real-Time Hand Gesture Recognition System:
Video Capture and Hand Detection:

- The webcam captures video feeds live in real-time.
- OpenCV and CV zone recognize hands using a hand tracking model and draw a bounding box around that detected hand.

Image Preprocessing:

- The resized hand with an aspect ratio, taken from the frame, is pasted onto a 300x300 white background for consistency with input size.

Gesture Classification:

- Feed the pre-processed image of the hand into a pre-trained Convolutional Neural Network (CNN).
- Use the network to decide which gesture the analyzed hand image represents.
- Returns the predicted label for the gesture

Real-Time Feedback:

- The assigned gesture label is then displayed in real-time over the video feed using text overflows within OpenCV.
- A bounding box has also been drawn around the detected hand to provide suitable and instant feedback to the user.

Saving the Hand Image:

- Users are enabled to save the processed hand image by clicking a predefined key.
- The image is saved with an appended time stamp to avoid overwriting of older files.

User Interface:

- The system may contain a simple GUI for users to interact with, offering options for saving and displaying recognized gestures.
-

Testing: Below are some test cases designed to validate the system:

Test Case 1: Hand Detection Accuracy

- Statement: The system should be able to identify hands, especially in low-light environments and different backgrounds.
- Outcome: Normal, low light and highly illuminated situations should be properly detected by the system and the detection should work irrespective of the complexity of the background.

Test Case 2: Gesture Recognition Accuracy

- Statement: It should be able to categorize different gestures provided by the registered users in the right manner.
- Outcome: The gestures and the gesture labels, such as "Thumbs Up," and "Palm Open," should be identified and labeled with maximum reliability.

Test Case 3: Real-Time Feedback

- Statement: For example, if it identifies a certain body gesture, the system should give the user instantaneous visual feedback on videos.
- Outcome: Real-time reporting of gesture labels is expected; changes in the bounding box around the detected hand should also be correct.

Test Case 4: System Response Time

- Statement: Hand detection and gesture recognition should be done in the same manner as they take a relatively shorter time than executing other complex tasks in real

- Outcome: The fingers and palms should be recognized the action they are miming should be categorized and the changes on the screen should be reflected immediately in less than 2 seconds after the hand movement has been captured.

Test Case 5: Gesture Vocabulary Expansion

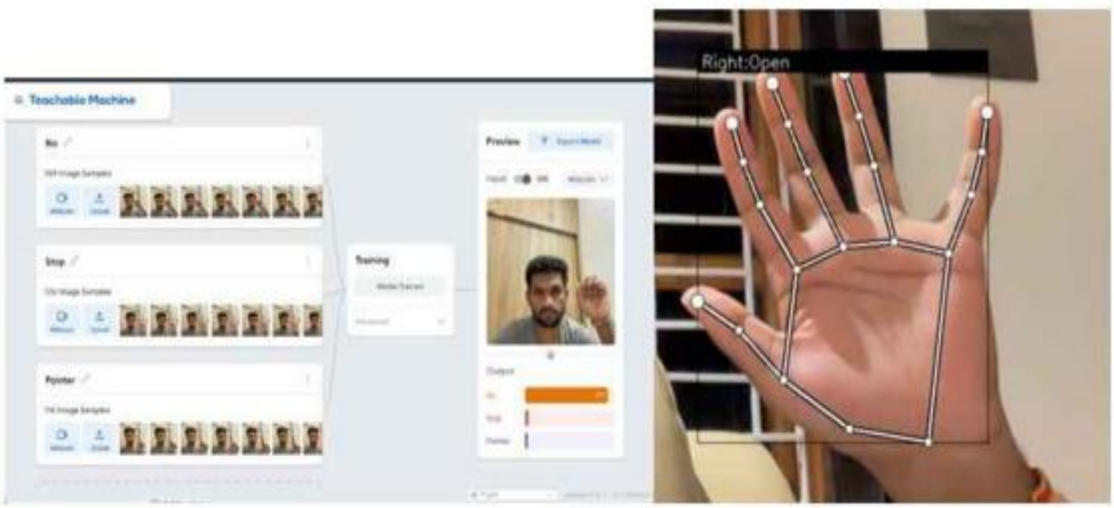
- Statement: New gestures added into the system should be recognized and classified without much tweaking on the system.
- Outcome: Ideally, the system should be able to extend the number and type of recognizable gestures while little or no decrease in the accuracy of the gesture recognition is acceptable.

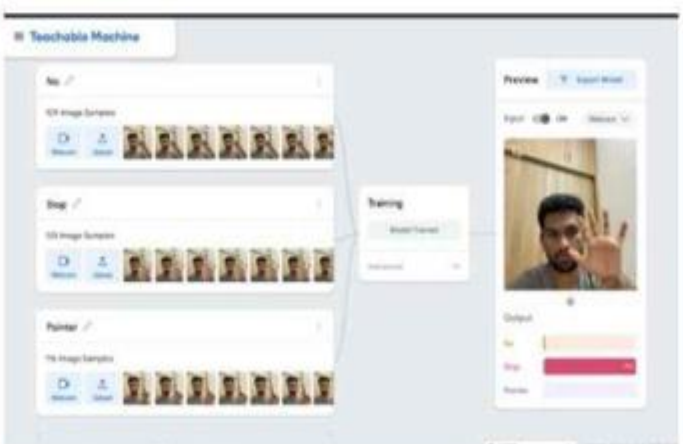
Test Case 6: User Experience and Interface Usability Further Reading

- Statement: For one, saving gestures and displaying identified labels ought to be easily achievable from the system.
- Outcome: The users should be able to easily save the gesture images and should be able to comprehend whatever information is being displayed on the interface.

7. RESULTS

The training setup in a tool like Teachable Machine, where multiple hand gestures are labeled and trained using image samples.





The model identifies gestures in real time by comparing the live camera feed against the trained classes. The output shows the detected gesture label. The model tracks hand landmarks, with lines and dots marking the key points on the hand, ensuring accurate gesture recognition. Output suggests a fully functioning gesture recognition system capable of interpreting and identifying specific hand gestures from live video, which can be used for ASL interpretation.

Result Analysis
System Performance:

- The system integrates real-time gesture recognition using machine learning techniques, with Python libraries like OpenCV and Mediapipe for video capture and hand detection.
- The system accurately recognizes both static and dynamic hand gestures, achieving real-time recognition suitable for assisting deaf and mute individuals.
- During testing, the system effectively detected gestures under various conditions, but performance may drop slightly in complex backgrounds or fast movements.

Accuracy and Efficiency:
- The system demonstrates high accuracy in recognizing various American Sign Language (ASL) gestures.
- Response time is optimized to provide real-time feedback within milliseconds, ensuring the system operates efficiently for seamless user interaction.

Security:
- While security is less concerned than facial recognition systems, the system is robust against background noise, ensuring that only relevant hand gestures are detected and processed.

Usability:
- The system's automated recognition of gestures significantly reduced manual effort, aiding users who need assistance in sign language communication.

Limitations:
- The system showed some sensitivity to complex backgrounds and lighting variations, which can impact gesture detection accuracy.
- Fast or dynamic gestures may not be captured effectively every time.
- The gesture vocabulary can be expanded in future versions to support a wider range of gestures for enhanced versatility.

8. CONCLUSION

The Real-Time Hand Gesture Recognition System does a great job regarding the transition that hand gesture recognition goes through this system, it is a friendly approach for communication specifically to the deaf and mute people. Through implementing rigorous machine learning algorithms and real-time video analysis, the system even acquires a high recognition rate for multiple different gestures of American Sign Language (ASL). Hand detection algorithms should be effectively designed and combined with stability and performance guarantees, and real-time feedback should be applied to improve user interactivity. Thus, during the recent testing, the system was able to prove its responsiveness to the different types of users and environmental settings providing consistently high results of the recognition. There is an intuitive design that enables the user to engage with it easily and use it by any group of people they may be. However, there are still some problems, for example, instability when working with different levels of illumination and tiredness of the user's eyes during work. These limitations provide prospects for the future development and improvement of the approach. In sum, the objectives of accuracy, user interactivity, and capacity characterizing the Real-Time Hand Gesture Recognition System are well achieved. The proposed system is useful in enhancing interaction and reducing barriers in various contexts. Its success provides evidence of how gesture recognition technology can be used as a tool to improve the interaction between humans and computers.

REFERENCES

[1]. Devira Anggi Maharani, "Hand Gesture Recognition using K-Means Clustering and Support Vector Machine", 978-1-5386-8, 2018

[2]. Vina Ayumi, "Performance evaluation of support vector machine algorithm for human gesture recognition", IJSRSET, 2394-4099, 2020

[3]. Meng Wu, "Gesture Recognition based on Deep Learning", School of Physics and Information Engineering, Jiangsu Second Normal University, 2024

[4]. Rubén E. Nogales * and Marco E. Benalcázar, "Hand Gesture Recognition Using Automatic Feature Extraction and Deep Learning Algorithms with Memory", Artificial Intelligence and Computer Vision Research Lab, 2023

[5]. Abhishek B. Kanya Krishi, Meghana M, Mohammed Daaniyaal, Anupama H, "Hand gesture recognition using machine learning algorithms", BMS Institute of Technology, Bangalore, India, 2722-3221, 2020

[6]. Mr. Pratik Annaramchandra Yadav, Mr. R. S. Patil, "Hand gesture recognition using machine learning", Department of E&TC, DBATU University Ashokrao Mane Group of Institutions, Wathar, India, 2582-7421, 2021

[7]. Hung-Yuan Chung, Yao-Liang Chung, Wei-Feng Tsai, "An Efficient Hand Gesture Recognition System Based on Deep CNN", the University of Texas Rio Grande Valley, 978-1-5386-6376-9/19, 2019

[8]. Vinayasree, P., Mallikarjuna Reddy, A. Blockchain-Enabled Hyperledger Fabric to Secure Data Transfer Mechanism for Medical Cyber-Physical System: Overview, Issues, and Challenges, EAI Endorsed Transactions on Pervasive Health and Technology, 2023, 9(1)

[9]. Mamidisetti, S., Reddy, A.M. Enhancing Depression Prediction Accuracy Using Filter and Wrapper-Based Visual Feature Extraction, Journal of Advances in Information Technology, 2023, 14(6), pp. 1425–1435

[10]. Mamidisetti, S., Reddy, A.M. A Stacking-based Ensemble Framework for Automatic Depression Detection using Audio Signals, International Journal of Advanced Computer Science and Applications, 2023, 14(7), pp. 603–612

[11]. Padmanabhuni, S.S., Reddy, B.S., Reddy, A.M., Reddy, K.S. Identification of Bacterial Diseases in Plants Using Re-Trained Transfer Learning in Quantum Computing Environment, Evolution and Applications of Quantum Computing, 2023, pp. 207–232

[12]. Padmavathi, V., Sujatha, C.N., Sitharamulu, V., Reddy, K.S., Reddy, A.M. Introduction to Quantum Computing, Evolution and Applications of Quantum Computing, 2023, pp. 1–14.

[13]. S. K.Sarangi ,R.Panda & Manoranjan Dash," Design of 1-D and 2-D recursive filters using crossover bacterial foraging and cuckoo search techniques", Engineering Applications of Artificial Intelligence, Elsevier Science,vol.34, pp.109-121,May 2014.

[14]. Manoranjan Dash, N.D. Londhe, S. Ghosh, et al., "Hybrid Seeker Optimization Algorithm-based Accurate Image Clustering for Automatic Psoriasis Lesion Detection", Artificial Intelligence for Healthcare (Taylor & Francis), 2022, ISBN: 9781003241409

[15]. Manoranjan Dash, Design of Finite Impulse Response Filters Using Evolutionary Techniques - An Efficient Computation, ICTACT Journal on Communication Technology, March 2020, Volume: 11, Issue: 01

[16]. Manoranjan Dash, "Modified VGG-16 model for COVID-19 chest X-ray images: optimal binary severity assessment," International Journal of Data Mining and Bioinformatics, vol. 1, no. 1, Jan. 2025, doi: 10.1504/ijdmb.2025.10065665.

[17]. Manoranjan Dash et al.," Effective Automated Medical Image Segmentation Using Hybrid Computational Intelligence Technique", Blockchain and IoT Based Smart Healthcare Systems, Bentham Science Publishers, Pp. 174-182,2024

[18]. Manoranjan Dash et al.," Detection of Psychological Stability Status Using Machine Learning Algorithms", International Conference on Intelligent Systems and Machine Learning, Springer Nature Switzerland, Pp.44-51, 2022.

[19]. Samriya, J. K., Chakraborty, C., Sharma, A., Kumar, M., & Ramakuri, S. K. (2023). Adversarial ML-based secured cloud architecture for consumer Internet of Things of smart healthcare. IEEE Transactions on Consumer Electronics, 70(1), 2058-2065.

[20]. Ramakuri, S. K., Prasad, M., Sathiyanarayanan, M., Harika, K., Rohit, K., & Jaina, G. (2025). 6 Smart Paralysis. Smart Devices for Medical 4.0 Technologies, 112.

[21]. Kumar, R.S., Nalamachu, A., Burhan, S.W., Reddy, V.S. (2024). A Considerative Analysis of the Current Classification and Application Trends of Brain–Computer Interface. In: Kumar Jain, P., Nath Singh, Y., Gollapalli, R.P., Singh, S.P. (eds) Advances in Signal Processing and Communication Engineering. ICASPACE 2023. Lecture Notes in Electrical Engineering, vol 1157. Springer, Singapore. https://doi.org/10.1007/978-981-97-0562-7_46.

[22]. R. S. Kumar, K. K. Srinivas, A. Peddi and P. A. H. Vardhini, "Artificial Intelligence based Human Attention Detection through Brain Computer Interface for Health Care Monitoring," 2021 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON), Dhaka, Bangladesh, 2021, pp. 42-45, doi: 10.1109/BECITHCON54710.2021.9893646.

[23]. Vytla, V., Ramakuri, S. K., Peddi, A., Srinivas, K. K., & Ragav, N. N. (2021, February). Mathematical models for predicting COVID-19 pandemic: a review. In Journal of Physics: Conference Series (Vol. 1797, No. 1, p. 012009). IOP Publishing.

[24]. S. K. Ramakuri, C. Chakraborty, S. Ghosh and B. Gupta, "Performance analysis of eye-state charecterization through single electrode EEG device for medical application," 2017 Global Wireless Summit (GWS), Cape Town, South Africa, 2017, pp. 1-6, doi:10.1109/GWS.2017.8300494.

[25]. Gogu S, Sathe S (2022) autofpr: an efficient automatic approach for facial paralysis recognition using facial features. Int J Artif Intell Tools. https://doi.org/10.1142/S0218213023400055

[26]. Rao, N.K., and G. S. Reddy. "Discovery of Preliminary Centroids Using Improved K-Means Clustering Algorithm", International Journal of Computer Science and Information Technologies, Vol. 3 (3), 2012, 4558-4561.

[27]. Gogu, S. R., & Sathe, S. R. (2024). Ensemble stacking for grading facial paralysis through statistical analysis of facial features. Traitement du Signal, 41(2), 225–240.

[28]. Daniel, G. V., Chandrasekaran, K., Meenakshi, V., & Paneer, P. (2023). Robust Graph Neural-Network-Based Encoder for Node and Edge Deep Anomaly Detection on Attributed Networks. Electronics, 12(6), 1501. https://doi.org/10.3390/electronics12061501

[29]. Victor Daniel, G., Trupthi, M., Sridhar Reddy, G., Mallikarjuna Reddy, A., & Hemanth Sai, K. (2025). AI Model Optimization Techniques. Model Optimization Methods for Efficient and Edge AI: Federated Learning Architectures, Frameworks and Applications, 87-108.

[30]. Lakshmi, M.A., Victor Daniel, G., Srinivasa Rao, D. (2019). Initial Centroids for K-Means Using Nearest Neighbors and Feature Means. In: Wang, J., Reddy, G., Prasad, V., Reddy, V. (eds) Soft Computing and Signal Processing . Advances in Intelligent Systems and Computing, vol 900. Springer, Singapore. https://doi.org/10.1007/978-981-13-3600-3_3