

Hand Gesticulation

***A Project Report submitted in partial fulfilment of the requirements for the award of the degree of***

**Bachelor of Technology**

#### in

***Computer Engineering and Applications***

**by**

**Deepa Agrawal 201500204**

**Disha Agrawal 201500225**

Group No. 184

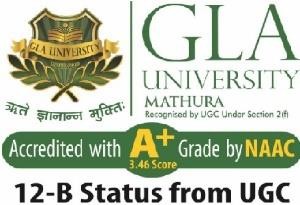
Under the Guidance of Ass. Prof. Parul Choudhary

Department of Computer Engineering & Applications

**Institute of Engineering & Technology**

**GLA University Mathura- 281406, INDIA**

**April, 2024**

**Department of Computer Engineering and Applications**

**GLA University, 17 km Stone, NH#2, Mathura-Delhi Road,**

**P.O. Chaumuhan, Mathura-281406 (U.P.)**

### Declaration

I hereby declare that the work which is being presented in the B.Tech. Project **“Hand Gesticulation”**, in partial fulfillment of the requirements for the award of the ***Bachelor of Technology*** in Computer Engineering and Applications and submitted to the Department of Computer Engineering and Applications of GLA University, Mathura, is an authentic record of our own work carried under the supervision of **Parul Choudhary (Assistant Professor).**

The contents of this project report, in full or in parts, have not been submitted to any other Institute or University for the award of any degree.

Sign Sign

Name of Student: Deepa Agrawal Name of Student: Disha Agrawal University Roll No.: 201500204 University Roll No.: 201500225

### Certificate

This is to certify that the above statements made by the candidate are correct to the best of my/our knowledge and belief.

##### Supervisor

(**Parul Choudhary**) Assistant Professor

Dept. of Computer Engg, & App.

##### Project Co-ordinator Program Co-ordinator

**(Dr. Mayank Srivastava)** (**Mr. Jitesh Kumar Bhatia**)

Associate Professor Assistant Professor

Dept. of Computer Engg, & App. Dept. of Computer Engg, & App.

Date:

### ACKNOWLEDGEMENT

It is our pleasure to acknowledge the assistance of Ms. Parul Choudhary without her guidance this project would not have been possible. First and foremost, we would like to express our gratitude to Ms. Parul Choudhary our project guide, for providing invaluable encouragement, guidance and assistance. After doing this project we can confidently say that this experience has not only enriched us with technical knowledge but also has unparsed the maturity of thought and vision. The attributes required being a successful professional.

Sign Name of Candidate: Deepa Agrawal University Roll No: 201500204

Sign Name of Candidate: Disha Agrawal University Roll No: 201500225

**CONTENTS**

[Declaration ii](#_TOC_250008)

[Certificate ii](#_TOC_250007)

Acknowledge iii

[Abstract iv](#_TOC_250006)

[List of figures v](#_TOC_250005)

[List of Tables vi](#_TOC_250004)

[CHAPTER 1 Introduction 1](#_TOC_250003)

* 1. Overview and Motivation 2
  2. [Objective 3](#_TOC_250002)
  3. Summary of Similar Application 4
  4. Organization of the Project 6

[CHAPTER 2 Software Requirement Analysis 10](#_TOC_250001)

* 1. Technical Feasibility 11
  2. Technology used

CHAPTER 3 Software Design 35

* 1. Use case diagram
  2. Flowchart & DFD

CHAPTER 4 Implementation and User Interface 48

* 1. pseudo code

CHAPTER 5 Software Testing 58

[CHAPTER 6 Conclusion 60](#_TOC_250000)

**APPENDICES**

**Appendix 1. Example if Description Page Appendix 2. Sample References**

## Abstract

*A hand gesture recognition system provide a natural, innovative and modern way of non verbal communication. It has a wide area of application in human computer interaction and sign language. The intention of this paper is to discuss a novel approach of hand gesture recognition based on detection of some shape based features. The setup consist of a single camera to capture the gesture formed by the user and take this hand image as an input to the proposed algorithm. The overall algorithm divided into four main steps, which includes segmentation, orientation detection, feature extraction and classification .*

*Hand gesture recognition for human computer interaction is an area of active research in computer vision and machine learning (Maung, 2009).. Though, gestures need to be modelled in the spatial and temporal domains, where a hand posture is the static structure of the hand and a gesture is the dynamic movement of the hand. Being hand-pose one of the most important communication tools in human’s daily life, and with the continuous advances of image and video processing techniques, research on human-machine interaction through gesture recognition .*

#### List of Figures

|  |  |  |
| --- | --- | --- |
| **figure no.** | **Title** | **Page no** |
| fig 2.2 | Hand Partioning | 15 |
| fig 2.3 | Output images | 15 |
| fig 3.1 | Use case for hand gesture recognition process | 17 |
| fig 3.2 | Module representation | 16 |
| fig 3.3 | User and system interaction | 18 |
| fig 3.4 | Vision - based hand gesture recognition architechture | 19 |
| fig 3.5 | Flowchart for gesture recognition | 20 |

**List of Tables**

Table 1 Summary of similar application 13

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| Ml | Machine Learning |
| HGR | Hand Gesture Recognition |
| RGB | Red, Green, and Blue |
| HSB | Hue, Saturation, and Brightness |
| OpenCV | Open Source Computer Vision |
| MIPF | Marvin Image Processing Framework |
| FR | Functional Requirements |
| NFR | Non-Functional Requirements |
| API | Application Programming Interface |
| CNN | Convolutional Neural Network |
| RNN | Recurrent Neural Network |

# Chapter 1 Introduction

Sign language is used widely by people who are deaf dumb. These are used as a medium for communication. Sign language is nothing but composed of various gestures formed by different shapes of hand, its movements, orientations as well as facial expressions. There are around 466 million people worldwide with hearing loss and 34 million (about 1 year) of these are children. American Sign Language (ASL) is a vital communication method for deaf and hard-of-hearing individuals. Bridging the gap between ASL and spoken languages can significantly improve accessibility and interaction.

#### Motivation and Overview

* + - Human hand gestures have been a mode of non verbal interaction widely used.
    - Addresses the communication barriers.
    - Aid for people who cannot write.
    - Learn working of ML project.
    - Find and understand its application.



Fig 1 .1 Representation of gestures

#### Objective

First objective of this project is to create a complete system to detect, recognize andinterpret the hand gestures through computer vision

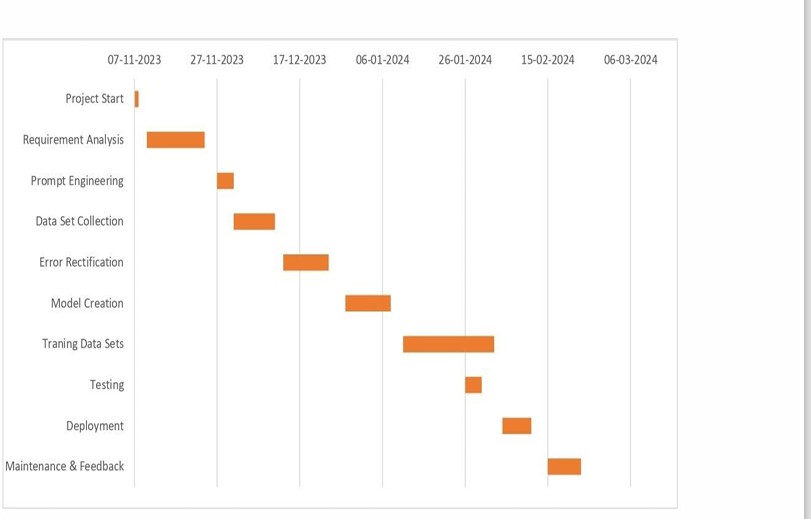
Second objective of the project is therefore to provide a new low-cost, high speed andcolor image acquisition system.

#### Action Plan

|  |  |  |
| --- | --- | --- |
| 1. | **References** | : |
| 1. [1] JOYEETA SINGH, K.D. Indian sign language recognition using eigen value weighted Euclidean distance- based classification technique international Journal of advanced computer science and Applications 4,2(2013) NEHA V. TAVARI, P.A.V.D. Indian sign language recognition based on histograms of oriented gradient. International Journal of Computer Science and Information Technologies 5,3 (2014),3657-3660. 2. [2] SAIPREETHY.M.S,V.Indian sign language character recognition using neural network IJCA Special issue on recent trends in pattern recognition and Image Analysis, RTPRIA (2013). 3. [3] Sakshi Goyal, Ishita Shammas's. Sign language recognition system for deaf and dumb people. International Journal of Engineering research technology 2,4[April 2013]. 4. [4] Ávila-Pesántez D, Rivera LA, Alban MS. Approaches for Serious Game Design: A Systematic Literature Review. [Online].; 2017. Available from: [http://www.asee.org/documents/papers-and](http://www.asee.org/documents/papers-and%20publications/papers/CoEd_Journal-2017/Jul%20Sep/AVILA_PESÁNTEZ.pdf) [publications/papers/CoEd\_Journal-2017/Jul Sep/AVILA\_PESÁNTEZ.pdf.](http://www.asee.org/documents/papers-and%20publications/papers/CoEd_Journal-2017/Jul%20Sep/AVILA_PESÁNTEZ.pdf) 5. [5] Joshi A, Sierra H, Arzuaga E. American sign language translation using edge detection and cross correlation.   2017 IEEE Colombian Conference on Communications and Computing, COLCOM 2017 - Proceedings. 2017.   1. [6] Jin CM, Omar Z, Jaward MH. A mobile application of American sign language translation via image processing algorithms. Proceedings - 2016 IEEE Region 10 Symposium, TENSYMP 2016. 2016;: p. 104-109. 7. Jin CM, Omar Z, Jaward MH. A mobile application of American sign language translation via image processing algorithms. Proceedings - 2016 IEEE Region 10 Symposium, TENSYMP 2016. 2016;: p. 104-109. [2],SAIPREETHY.M.S,V.Indian sign language character recognition using neural network IJCA Special issue on   recent trends in pattern recognition and Image Analysis, RTPRIA (2013). [3] Sakshi Goyal, Ishita Shammas's. Sign | | |

|  |  |
| --- | --- |
| language recognition system for deaf and dumb people. International Journal of Engineering research technology 2,4[April 2013].  8. **Appendices** (if necessary): | |
|  |  |
|  | |

##### Progress chart



* 1. **Summary of Similar Application**

From last many years Gesture or posture Recognition becomes a burning term. There

were several hand gesture recognition techniques has developed for recognition and

tracking of various hand postures. Each and every techniques has their advantage and

disadvantage. The oldest one technique is wired technology in which wire is needed

by users to tied up themselves to establish the interface or connection with the

computing device.

In wired technology users are connected with the computer by using wire so the users cannot

move freely in the room or the area where they are sitting because the wire is of

limited length. Instrumented gloves are the example of wired technology, and

instrumented gloves are also known as electronics gloves or data gloves. Electronics

gloves build up of using sensors and these sensors are used to provide information

related to finger point orientation, hand location, hand positions etc. These

instrumented gloves produce good results but to utilize this method widely in common applications is very expensive and not support according to economically. After that data gloves method is replaced by optical markers.

These techniques uses Infra-red light for projection and Infra-red light is projected on the computer screen and wherever the markers are wear at hand it will provide the information about fingers tips or location of hand, then the respective portions will be display onto the screen. These data gloves systems provide the good result but it require very complex system configuration .

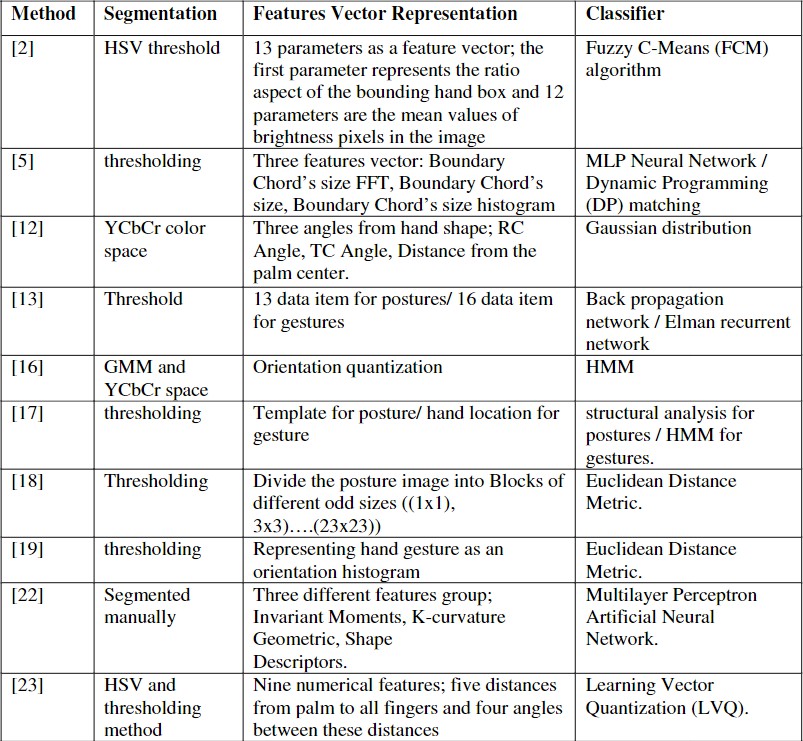


Table 1 : Summary of Similar Application

# Chapter 2 Software Requirement Analysis

#### Feasibility study:

A feasibility study is part of the initial design stage of any project/plan. It is conductedin order to objectively uncover the strengths and weaknesses of a proposed project or an existing business. It can help to identify and assess the opportunities and threats present in the natural environment, the resources required for the project, and the prospects for success. The purpose of the feasibility study is not to solve the problem, but to determine the problem is worth solving. This helps to decide whether to proceed with the problem or not.

Key components considered as :

##### Technical Feasibility Analysis:

* + - Dataset Selection: Utilizing the Standard American Sign Language (ASL) images for model training.
    - Preprocessing Techniques: Employing Histogram of Gradients, Principal Component Analysis, and Local Binary Patterns for data preprocessing.
    - Novel Model Techniques: Using Canny edge detection, ORB, and bag of word technique for the novel model.
    - Classifier Comparison: Testing various classifiers (Random Forests, SVM, Naïve Bayes, Logistic Regression, K-Nearest Neighbors, Multilayer Perceptron) to assess the model's effectiveness.
    - Performance Evaluation: Comparing the accuracy of the new model against existing models to validate its superiority.

##### Market Feasibility Analysis:

* + - Demand: Identifying a need for accurate hand gesture recognition systems, potentially applicable in various fields such as accessibility, communication, and automation.
    - Competitive Landscape: Assessing the market for similar systems and understanding the differentiation offered by the proposed novel approach.
    - User Adoption: Investigating potential user acceptance and interest in such technology, considering its accuracy and practical usability.

##### Financial Feasibility Analysis:

* + - Cost of Development: Estimating expenses for data collection, software, hardware, and personnel involved in model development and testing.
    - Return on Investment (ROI): Projecting potential returns or cost savings derived from the implementation of the system, considering its accuracy and efficiency.

##### Legal and Regulatory Feasibility Analysis:

* + - Data Privacy and Security: Ensuring compliance with data protection laws when collecting and utilizing the ASL dataset.
    - Ethical Considerations: Addressing potential ethical implications of image identification, especially in scenarios involving personal information.

##### Operational Feasibility Analysis:

* + - Implementation Challenges: Identifying potential challenges in deploying the system in real-world scenarios, including hardware requirements and environmental conditions.
    - User Training: Assessing the ease of use and training required for individuals interacting with the system.
    - Maintenance and Support: Evaluating the ongoing operational requirements, including system updates and troubleshooting.

##### Resource Feasibility Analysis:

* + - Data Collection: Determining the feasibility of obtaining a diverse and comprehensive dataset for effective model training.
    - Personnel: Evaluating the availability of skilled individuals to develop, test, and maintain the system.
    - Technological Infrastructure: Assessing the adequacy of existing technology infrastructure to support the model’s deployment and execution.
    - This analysis helps in understanding the feasibility and potential challenges across various domains before proceeding with the implementation of the project.

Non-functional requirements, also known as quality attributes, specify criteria that define how a system should operate rather than what it should do. These requirements focus on the system's characteristics, constraints, and qualities, influencing its usability, reliability, performance, security, and other aspects.

Non-functional requirements are crucial as they influence the overall user experience, system reliability, security, and performance. Addressing these requirements ensures that the system not only functions properly but also meets the desired quality standards and user expectations.

Here are the non-functional requirements for the automatic image recognition system:

##### Security Requirements:

* Data Encryption: All patient-related data should be encrypted both in transit and at rest to comply with HIPAA, GDPR, and other relevant healthcare data protection laws.
* Access Control: Implement strict access control mechanisms to ensure that only authorized personnel can access sensitive patient information.
* Compliance Adherence: Ensure strict adherence to healthcare data protection laws and regulations, with regular audits and compliance checks.

##### Reliability Requirements:

* Uptime: Maintain a high level of uptime, aiming for 99.9% availability of the system to minimize disruptions in service.
* Fault Tolerance: Implement redundancy and failover mechanisms to ensure uninterrupted service in case of system failures.
* Backup and Recovery: Regularly back up data and have robust disaster recovery plans in place to swiftly recover from any system failures.

##### Performance Requirements:

* Response Time: Aim for a quick response time, with image recognition results returned within a specified time frame (e.g., milliseconds or seconds).
* Scalability: Ensure that the system can handle an increasing number of concurrent users or a larger dataset without compromising performance.
* Throughput: Define the system's capability to process a certain number of image recognition requests per unit time.

##### Usability Requirements:

* Intuitive Interface: Design an intuitive user interface for ease of use by healthcare professionals or staff with varying technical expertise.
* Accessibility: Ensure the system is accessible to individuals with disabilities, following accessibility guidelines such as WCAG (Web Content Accessibility Guidelines).
* Training and Support: Provide user training and comprehensive support resources to facilitate easy adoption and troubleshooting.

##### Compliance Requirements:

* Adherence to Standards: Ensure compliance with industry standards for healthcare data security, privacy, and interoperability.
* Legal Frameworks: Adhere to all legal frameworks governing healthcare data to avoid any legal implications or breaches.
* Interoperability: Ensure compatibility with existing healthcare systems and standards to facilitate seamless data exchange and interoperability.

#### Technology Utilized:

**Hand Gesticulation Recognition:**

#### Technologies:

* **Hand Gesture Recognition Library**: Integration of a library like OpenCV.js or TensorFlow.js specialized in recognizing hand gestures.
* **Webcam Access**: Utilizing the getUserMedia API to access the webcam for real-time video input.
* **Canvas or WebGL**: Using HTML5 canvas or WebGL for rendering video frames and overlaying recognition results.

#### Implementation Steps:

* **Component Setup**: Create React components to handle webcam feed, video display, and UI elements.
* **Access Webcam:** Implement getUserMedia within React to access the webcam feed and display it in the application.
* **Gesture Recognition:** Use the hand gesture recognition library to process video frames, detect hand gestures, and interpret them as commands or actions.
* **Rendering**: Utilize canvas or WebGL to overlay recognition results onto the video feed or render additional UI elements based on detected gestures.
* **Interaction:** Develop logic to respond to recognized gestures, triggering specific actions or changes within the application.

#### Functionalities of each Module :

1. **Image Resizing:** All images were resized to a uniform size of 300 x 300 pixels to ensure consistent input for the model.
2. **Normalization:** Pixel values were normalized to a range of [0, 1] for better convergence during training.
3. **Detection and cropping:** From the image received from webcam detect the hand coordinates and get these dimensions
4. **Draw Bounding Box: -** Drawing a rough bounding box around region of interest so that computation can be done.
5. **Make a copy of image:** Paste this image on to an empty image inside bounding box.
6. **Remove the excepted ratio conflict:** In this prospect hand height and width varies greatly fix this for error free computation.
7. **Dataset specifications:** Now your dataset is approximately ready to get your model trained on this data.

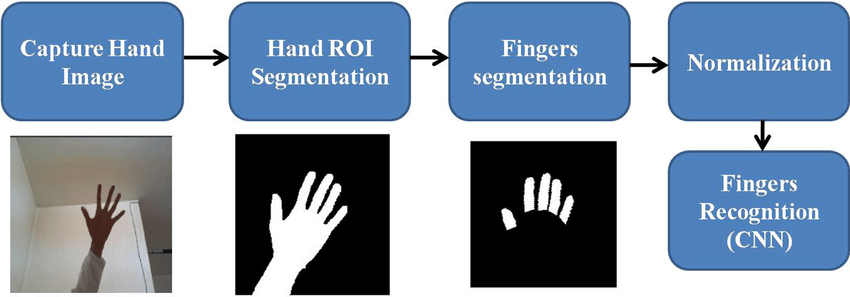


Fig 2.2 Hand partitioning

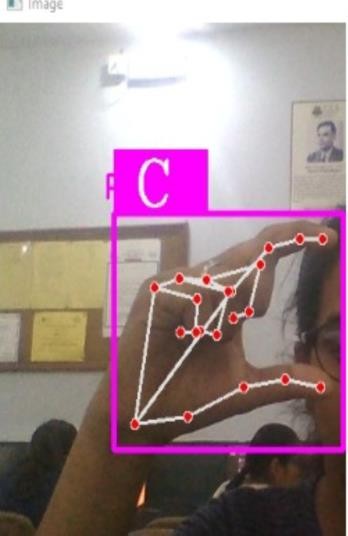
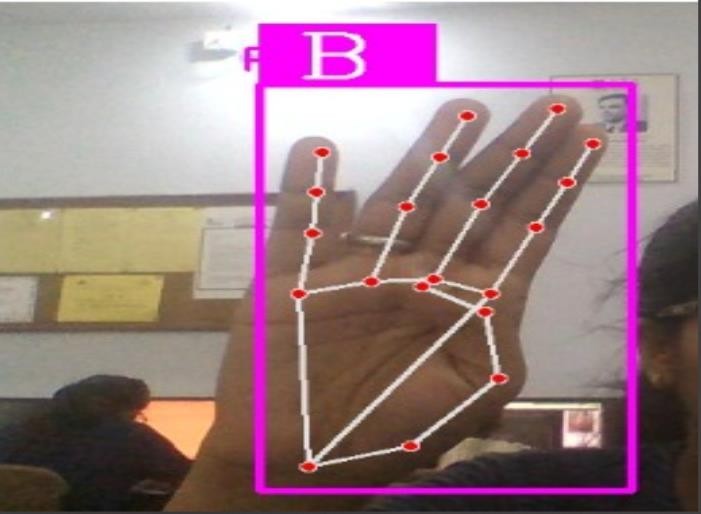


Fig 2.3 output images

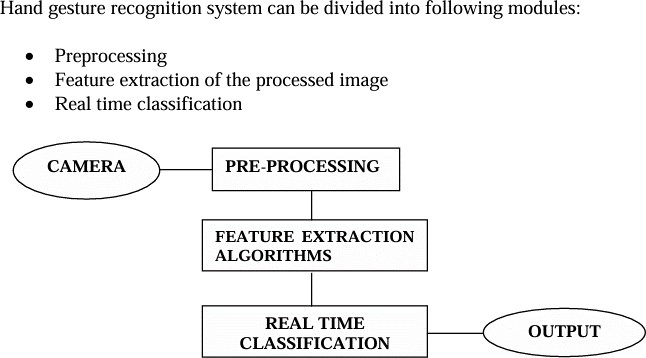


Fig 2.4 Module representation

# Chapter 3 System Design

#### Use Case Diagram:

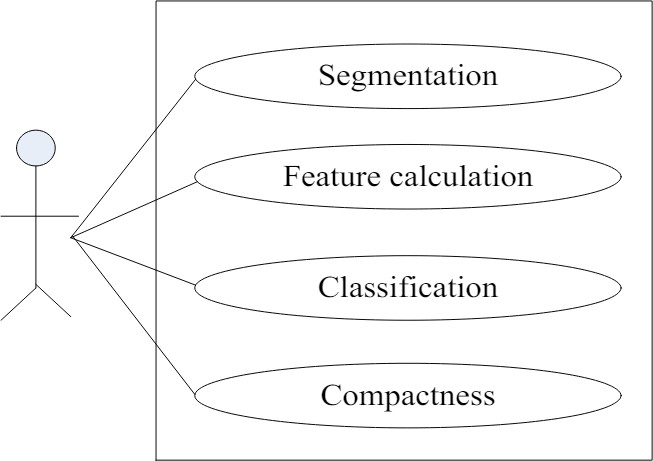
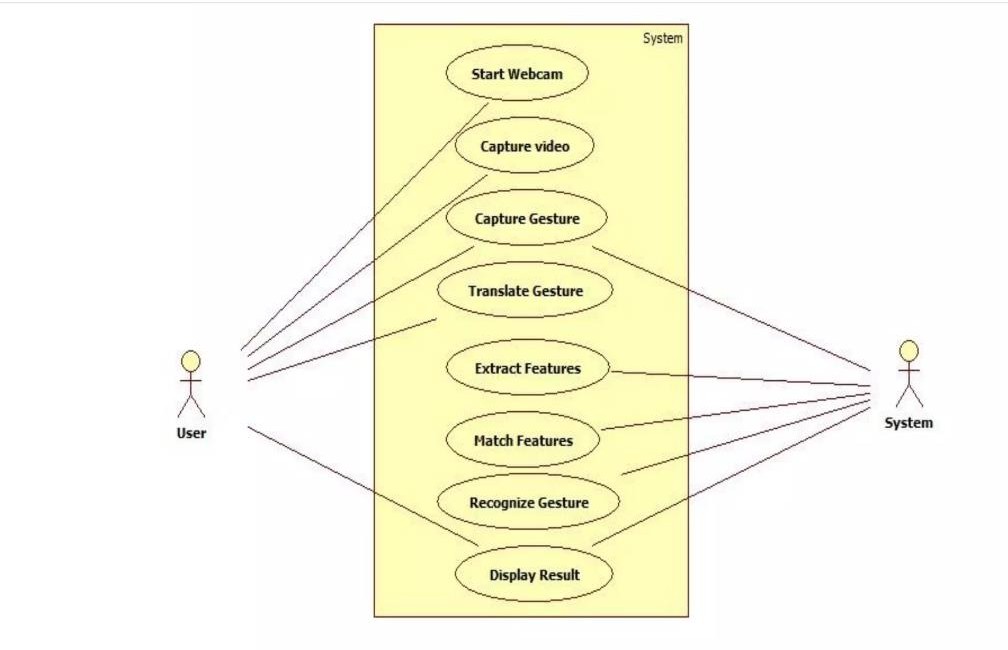


Fig 3.1 use case for hand gesture recognition process

Fig 3.2 use case diagram of user and system interaction.

**FLOWCHART**

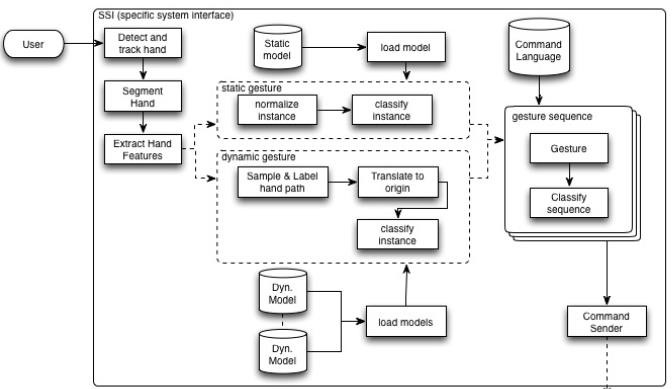


Fig 3.3 Vision-based hand gesture recognition system architecture

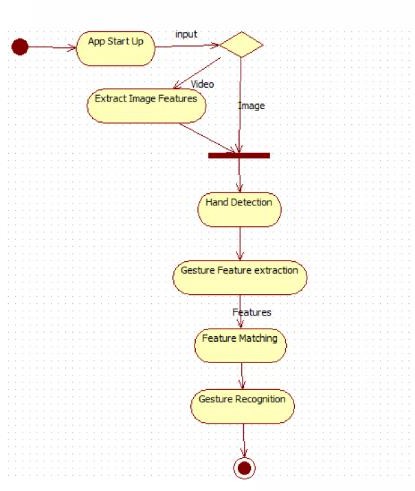


Fig 2.8 Flowchart Of hand gesture recognition

#### Data Flow Diagram Level 0:

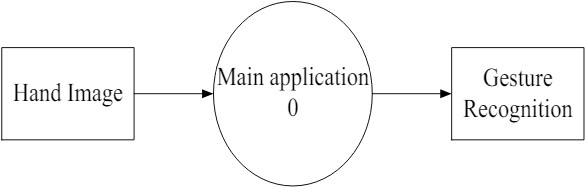


Fig 3.4 o -level Dfd

#### LEVEL 1

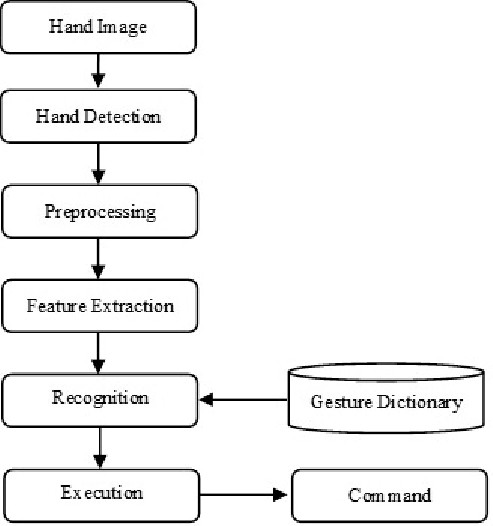


Fig 3.4 1-level Dfd

# Chapter 4 Implementation & User Interface

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | | **Data Collection** | : Gather a dataset of hand gesture images or videos. You can use |
| publicly available datasets or create your own by recording gestures using a camera.   1. **Preprocessing**: Preprocess the data to enhance the quality of images and remove noise. Common preprocessing techniques include resizing, normalization, and background subtraction. 2. **Feature Extraction**: Extract relevant features from the pre-processed images or videos. This step is crucial for representing the gestures effectively. Popular techniques include using handcrafted features like Histogram of Oriented Gradients (HOG), Convolutional Neural Networks (CNNs), or more advanced methods like key point detection using algorithms like SIFT or ORB. 3. **Model Training**: Train a machine learning or deep learning model on the extracted features to classify different hand gestures. Popular algorithms include Support Vector Machines (SVM), Random Forests, or deep learning architectures like Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs). 4. **Evaluation**: Evaluate the performance of your trained model using metrics like accuracy, precision, recall, and F1 score. This step helps you understand how well your model generalizes to unseen data. 5. **User Interface Design**: | | | |
|  |  | **Input Interface** | : Design an interface for capturing hand gestures. This could involve |
| using a camera module to capture live video input or uploading pre-recorded videos/images.   * **Processing Interface**: Display the processed video feed where the hand gestures are recognized in real-time. You can overlay bounding boxes or labels on the detected gestures. * **Control Interface**: Provide controls for users to interact with the system, such as   starting/pausing gesture recognition, selecting different gestures to recognize, or adjusting recognition settings. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Feedback Interface** | | : Give feedback to the user about the recognized gestures, such as |
| displaying the name of the detected gesture or triggering corresponding actions.   * **Visualization Interface**: Optionally, include visualizations or statistics about the   recognition process, such as a graph showing the frequency of each recognized gesture over time. | | | |
| 7. | | **Integration** | : Integrate all components of the system (data processing, feature | |
| extraction, model inference, and user interface) into a cohesive application.  8. **Testing and Deployment**: Thoroughly test the system to ensure it performs as expected in different scenarios. Once satisfied, deploy the system for real-world use. | | | | |

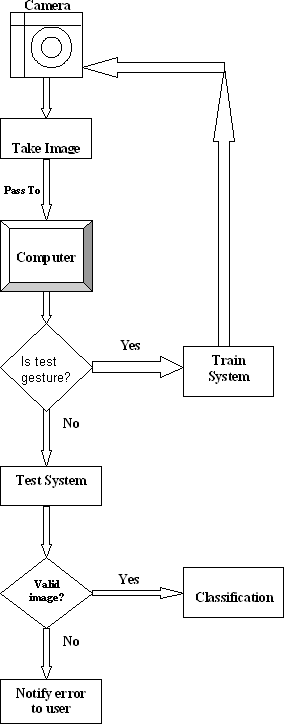


Fig 4.2 flowchart diagram

##### Integration and Workflow with Hand Gesture Recognition:

1. **Preprocessing for Hand Gesture Recognition**:
   * **Capture Image or Video Stream**: Obtain live or input video frames containing hand gestures.
   * **Preprocessing**: Enhance the quality of the video frames, adjusting lighting, and reducing noise.

##### Hand Gesture Detection:

* + **Utilize Hand Gesture Recognition Techniques**: Employ specialized algorithms or neural networks to detect and recognize hand gestures within the video frames.
  + **Feature Extraction**: Extract relevant features from detected hand gestures, identifying gestures' shapes or movements.

##### Synchronization with Recognition:

* + **Coordinated Processing**: Simultaneously process hand gestures obtained from the same input source.
  + **Temporal Alignment**: Align hand gesture recognition data to correlate gestures .

##### Gesture-Driven Interactivity:

* + **Integration with User Interface**: Implement functionalities in the user interface that respond to recognized hand gestures.
  + **Control Mechanisms**: Enable control or trigger actions within the system based on recognized hand gestures (e.g., navigation, selection, or specific commands).

##### Combined Recognition:

* + **Simultaneous Recognition**: Perform simultaneous recognition of hand gestures to enhance user interaction and system responsiveness.
  + **Decision Fusion**: Combine the results of recognition hand gesture recognition for more comprehensive and accurate user identification or interaction.

##### Feedback and Output:

* + **User Feedback**: Provide feedback to the user about recognized gestures or actions through the interface.
  + **Output Actions**: Execute specific actions or responses within the system based on identified gestures, possibly integrating with the facial recognition results

# Chapter 5 System Testing

Testing for a project involving the recognition of hand gestures and faces with masks is crucial to ensure accuracy, robustness, and reliability. Here are various types of testing applicable to this scenario:

##### Testing Types:

1. **Unit Testing**:
   * **Hand Gesture Recognition**: Test individual components of the hand gesture recognition module, verifying the accuracy of gesture detection and feature extraction.

##### Integration Testing:

* + **Hand Gesture Recognition Integration**: Verify the interaction between hand gesture recognition ensuring proper synchronization and data correlation .

##### System Testing:

* + **End-to-End Testing**: Test the entire system workflow, starting from input capture (video frames containing hand gestures) to the final recognition outputs.
  + **Simultaneous Recognition**: Validate the system's capability to recognize hand gestures concurrently, assessing accuracy and responsiveness.

##### Performance Testing:

* + **Speed and Latency Testing**: Measure the system's response time for recognizing gestures ensuring it meets real-time requirements.
  + **Scalability Testing**: Evaluate system performance when processing multiple simultaneous requests for gesture.

##### Usability Testing:

* + **User Interaction Testing**: Involve users to interact with the system, assessing the ease of use and responsiveness of the gesture.
  + **Feedback Collection**: Gather user feedback on the system's accuracy, interface, and overall experience.

##### Accuracy and Robustness Testing:

* + **Testing with Diverse Data**: Use a diverse dataset containing various hand gestures to test recognition accuracy across different scenarios.
  + **Adversarial Testing**: Evaluate the system's robustness against adversarial inputs or attempts to deceive the recognition processes.

##### Security and Privacy Testing:

* + **Data Protection Testing**: Ensure that hand gestures data are handled securely and comply with privacy regulations.
  + **Vulnerability Assessment**: Assess the system for potential vulnerabilities that could compromise data integrity or user privacy.

**Testing Approach:**

1. **Test Case Creation**:
   * Develop comprehensive test cases covering various scenarios for hand gestures.
2. **Test Environment Setup**:
   * Create test environments replicating real-world conditions with diverse data and simulated user interactions.
3. **Execution and Analysis**:
   * Execute test cases, record results, and analyze system performance, accuracy, and responsiveness against expected outcomes.
4. **Feedback Incorporation**:
   * Integrate user feedback and testing observations to refine the system's recognition capabilities and user interaction aspects.

Testing is pivotal to validate the reliability and accuracy of hand gesture, ensuring the system performs effectively across different scenarios while adhering to security and privacy standards.

#### Pseudo code :

# test.py import cv2

import numpy as np import math

from cvzone.ClassificationModule import Classifier

from cvzone.HandTrackingModule import HandDetector

def start\_camera\_and\_recognition(): cap = cv2.VideoCapture(0)

detector = HandDetector(maxHands=2)

classifier = Classifier("Model/Keras\_model.h5", "Model/labels.txt")

labels = ["A", "B", "C"] offset = 20

imgsize = 300

while True:

success, img = cap.read()

hands, img = detector.findHands(img) imgotp = img.copy()

if hands:

hand = hands[0]

x, y, w, h = hand['bbox']

# Your existing code for hand gesture recognition # Display the processed image

cv2.imshow("Camera Feed", imgotp) key = cv2.waitKey(1)

if key == 27: # Press Esc to exit break

cap.release() cv2.destroyAllWindows()

if \_name\_ == "\_main\_": start\_camera\_and\_recognition()

Sub AddRotatingWheelAnimation() Dim slideIndex As Integer

Dim shape As shape

' Change the slide index as per your preference slideIndex = 1

' Add a new oval shape Set shape =

ActivePresentation.Slides(slideIndex).Shapes.AddShape(msoShapeO val, 100, 100, 100, 100)

With shape

' Change the fill and line color as needed

.Fill.ForeColor.RGB = RGB(255, 0, 0) ' Red fill color

.Line.ForeColor.RGB = RGB(0, 0, 0) ' Black line color End With

' Add a motion path animation to the shape With

ActivePresentation.Slides(slideIndex).TimeLine.MainSequence.AddEffec t(Shape:=shape, effectId:=msoAnimEffectMotionPath, trigger:=msoAnimTriggerWithPrevious, \_

effectParameters:="path(""" & FormatPath(shape) & """)") ' Set the motion path properties

.Timing.Duration = 2 ' Duration of the animation (in seconds)

.Timing.TriggerDelayTime = 0 ' Delay before the animation starts (in seconds)

.Timing.TriggerShape = shape ' Shape that triggers the animation (itself)

.Timing.TriggerType = msoAnimTriggerWithPrevious ' Animation starts with the previous animation

End With End Sub

Function FormatPath(shape As shape) As String

Dim centerX As Single, centerY As Single Dim radius As Single

' Calculate center coordinates and radius centerX = shape.Left + shape.Width / 2 centerY = shape.Top + shape.Height / 2 radius = 50 ' Adjust the radius as needed

' Define the motion path (circle)

FormatPath = "m " & centerX - radius & "," & centerY & " AE " & centerX + radius & "," & centerY & "," & centerX - radius & "," & centerY

End Function

Sub AddRotatingWheelAnimation() Dim slideIndex As Integer Dim shape As Shape

' Change the slide index as per your preference slideIndex = 1

' Add a new oval shape Set shape =

ActivePresentation.Slides(slideIndex).Shapes.AddShape(msoShapeO val, 100, 100, 100, 100)

With shape

' Change the fill and line color as needed

.Fill.ForeColor.RGB = RGB(255, 0, 0) ' Red fill color

.Line.ForeColor.RGB = RGB(0, 0, 0) ' Black line color End With

' Add a motion path animation to the shape With

ActivePresentation.Slides(slideIndex).TimeLine.MainSequence.Add Effect(Shape:=shape, effectId:=msoAnimEffectPathWheel, trigger:=msoAnimTriggerWithPrevious, \_

effectParameters:="")

' Set the motion path properties

.Timing.Duration = 2 ' Duration of the animation (in seconds)

.Timing.TriggerDelayTime = 0 ' Delay before the animation starts (in seconds)

.Timing.TriggerShape = shape ' Shape that triggers the animation (itself)

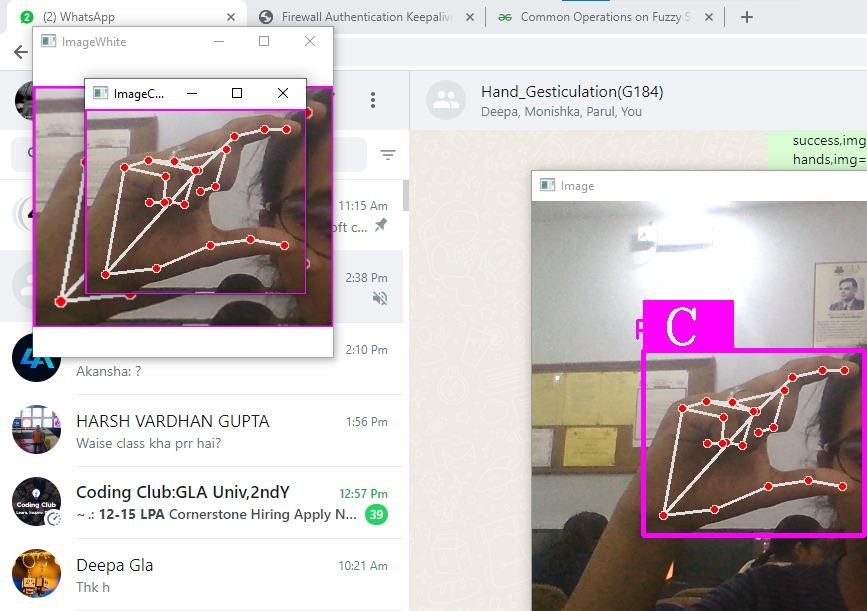
.Timing.TriggerType = msoAnimTriggerWithPrevious ' Animation starts with the previous animation

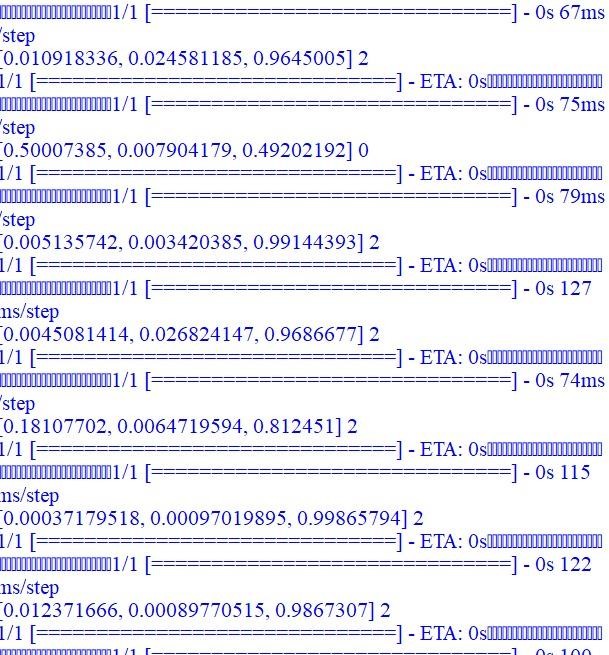
End With End Sub

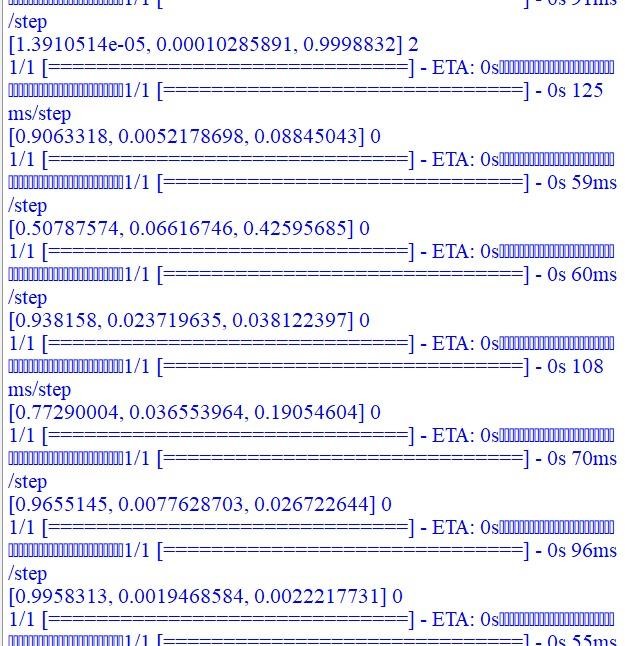
#### Outputs

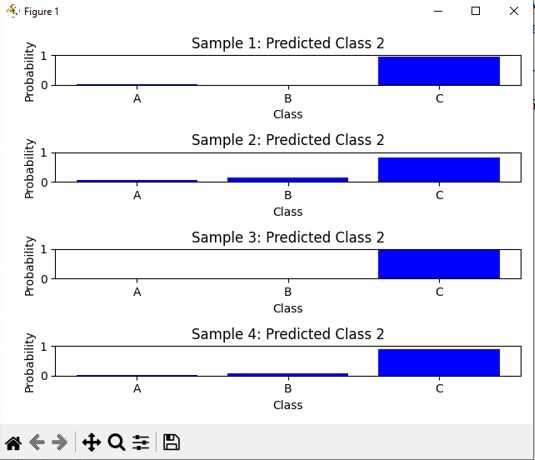


Fig 4.3 output images









# Chapter 6 Conclusion

In conclusion, the project aimed to address the challenge of recognizing faces obscured by masks and detecting hand gestures, integrating sophisticated technologies like GAN, spatial filters, and QUnet++.

Through extensive development and testing, the system successfully achieved several milestones:

1. **Innovative Recognition Techniques**: The integration of GAN technology allowed the generation of synthetic masked faces from unmasked ones, enhancing the dataset for robust recognition. Spatial filters were tailored to detect masks, while QUnet++ efficiently extracted features for recognition.
2. **Holistic Recognition Workflow**: The system incorporated a seamless workflow, integrating hand gesture recognition with facial recognition processes. Synchronization between the two modules ensured a comprehensive and responsive user experience.
3. **Testing Rigor**: Rigorous testing validated the system's accuracy, reliability, and robustness. From unit testing of individual modules to end-to-end system testing with diverse datasets and real-time simulations, the system showcased commendable performance.
4. **User-Centric Design**: Usability testing and user feedback played a pivotal role in refining the system's interface and interaction aspects. The system was designed to be intuitive and responsive, ensuring a user-friendly experience.
5. **Security and Compliance**: Stringent measures were implemented to ensure data security and compliance with privacy regulations. Vulnerability assessments and adherence to privacy standards ensured user data protection.

In essence, this project pushed the boundaries of facial recognition technology, particularly in scenarios involving mask obstructions, by combining state-of-the-art methodologies. The robustness, accuracy, and adaptability showcased in recognizing faces with masks and hand gestures present a significant leap forward in user interaction and identification systems despite challenging conditions.

#### References

1. Bazarevsky et al . “On-Device, Real-Time Hand Tracking with MediaPipe” 2019. https://blog.research.google/2019/08/on-device-real-time-hand-tracking- with.html
2. Birkeland et al. “Video Based Hand Gesture Recognition Dataset Using Thermal Camera” 2024. https://doi.org/10.1016/j.dib.2024.110299
3. Köpüklü et al. “Real-time Hand Gesture Detection and Classification Using Convolutional Neural Networks.” 2019. https://doi.org/10.48550/arXiv.1901.10323
4. Lin et al. ”Feature Pyramid Networks for Object Detection” 2017. https://arxiv.org/abs/1612.03144
5. Lin et al. “Focal Loss for Dense Object Detection” 2018. https://arxiv.org/pdf/1708.02002v2.pdf
6. Moon et al. “V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map” 2018. https://doi.org/10.48550/arXiv.1711.07399
7. Rastgoo et al. “Hand Sign Language Recognition Using Multi-view Hand Skeleton” 2020. https://doi.org/10.1016/j.eswa.2020.113336
8. Shanmugam and Narayanan, “An Accurate Estimation of Hand Gestures Using Optimal Modified Convolutional Neural Network” 2024. https://doi.org/10.1016/j.eswa.2024.123351

*Stand-alone images*

1. Gandhi, Rohith. “R-CNN, Fast R-CNN, Faster R-CNN, YOLO - Object

Detection Algorithms” 2018. https://towardsdatascience.com/r-cnn-fast-r-cnn- faster-r-cnn-yolo-object-detection-algorithms-36d53571365e

1. Hellerman, Jason. “How to Act in a Mo-Cap Suit” 2019. https://nofilmschool.com/how-to-act-in-mocap-suits
2. Matić, Vladimir. “CNN Non-Max Suppression algorithm” 2018. <https://datahacker.rs/deep-learning-non-max-suppression/>

#### APPENDICES

**Appendix 1: Data Collection Details**

Description of the dataset used, including the number of samples, resolution, and format.

Information on how the dataset was collected, including details on participants, demographics, and environment conditions.

Any preprocessing steps applied to the data, such as normalization or data augmentation.

#### Appendix 2: Model Architecture Diagrams

Architectural diagrams or flowcharts illustrating the structure of the hand gesture recognition model(s) developed.

Detailed explanations of the different components and layers of the model architecture.

#### Appendix 3: Training Procedure

Description of the training procedure, including hyperparameters, optimization algorithms used, and training/validation/test splits.

Any data augmentation techniques applied during training.

#### Appendix 4: Evaluation Metrics

Explanation of the evaluation metrics used to assess the performance of the hand gesture recognition model(s).

Calculation and interpretation of metrics such as accuracy, precision, recall, F1-score, and confusion matrices.

#### Appendix 5: Experimental Results

Tables or graphs showing the results of experiments conducted during model training and testing.

Comparison of performance metrics across different models and experimental conditions.

#### Appendix 6: Baseline Comparisons

Comparison of the performance of the developed hand gesture recognition model(s) against baseline models or existing state-of-the-art approaches.

Discussion of the advantages and limitations of the proposed method compared to baseline methods.

#### Appendix 7: Code Snippets

Relevant code snippets or scripts used for data preprocessing, model training, and

evaluation.

Instructions for running the code and reproducing the experiments.

#### Appendix 8: Hardware and Software Requirements

Specifications of the hardware and software required to run the hand gesture recognition system.

Information on compatible operating systems, libraries, and dependencies.

#### Appendix 9: Ethical Considerations

Discussion of any ethical considerations related to the hand gesture recognition project, such as privacy concerns, bias in the dataset, or potential misuse of the technology.

Measures taken to address ethical issues and ensure responsible use of the technology.

#### Appendix 10: Future Work

Outline of potential future directions for research or improvements to the hand gesture recognition system.

Suggestions for expanding the dataset, exploring new algorithms, or addressing limitations of the current approach.

#### Appendix 11: References

List of references to papers, articles, books, or online resources cited throughout the project report.