Visvesvaraya Technological University, Belagavi – 590018

PROJECT REPORT
ON

CLOUD-BASED SMART MONITORING SYSTEM FOR BABY HEALTH AND SAFETY

Submitted in partial fulfillment for the award of degree of

BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE & ENGINEERING

Submitted by

Aaron Tauro	4SO21CS002
Abhik L Salian	4SO21CS004
Akhil Shetty M	4SO21CS013
H Karthik P Nayak	4SO21CS058

Under the Guidance of

Dr Sridevi Saralaya
Professor, Department of CSE

DEPT. OF COMPUTER SCIENCE AND ENGINEERING
ST JOSEPH ENGINEERING COLLEGE
An Autonomous Institution
(Affiliated to VTU Belagavi, Recognized by AICTE, Accredited by NBA)
Vamanjoor, Mangaluru - 575028, Karnataka

2024-25

ST JOSEPH ENGINEERING COLLEGE

An Autonomous Institution

(Affiliated to VTU Belagavi, Recognized by AICTE, Accredited by NBA)

Vamanjoor, Mangaluru - 575028, Karnataka

DEPT. OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project work entitled **"Cloud-Based Smart Monitoring System for Baby Health and Safety"** carried out by

Aaron Tauro	**4SO21CS002**
Abhik L Salian	**4SO21CS004**
Akhil Shetty M	**4SO21CS013**
H Karthik P Nayak	**4SO21CS058**

the bonafide students of VIII semester Computer Science & Engineering in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2024-2025. It is certified that all corrections/suggestions indicated during Internal Assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said degree.

Dr Sridevi Saralaya	**Dr Sridevi Saralaya**	**Dr Rio D'Souza**
Project Guide	HOD-CSE	Principal

External Viva:

Examiner's Name	Signature with Date
1.	....
2.	....

ST JOSEPH ENGINEERING COLLEGE

An Autonomous Institution

(Affiliated to VTU Belagavi, Recognized by AICTE, Accredited by NBA)

Vamanjoor, Mangaluru - 575028, Karnataka

DEPT. OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We hereby declare that the entire work embodied in this Project Report titled **"Cloud-Based Smart Monitoring System for Baby Health and Safety"** has been carried out by us at St Joseph Engineering College, Mangaluru under the supervision of **Dr Sridevi Saralaya,** for the award of **Bachelor of Engineering** in **Computer Science & Engineering**. This report has not been submitted to this or any other University for the award of any other degree.

Aaron Tauro USN:4SO21CS002

Abhik L Salian USN:4SO21CS004

Akhil Shetty M USN:4SO21CS013

H Karthik P Nayak USN:4SO21CS058

Acknowledgement

We dedicate this page to acknowledge and thank those responsible for the shaping of the project. Without their guidance and help, the experience while constructing the dissertation would not have been so smooth and efficient.

We sincerely thank our Project guide **Dr Sridevi Saralaya**, Professor, Computer Science and Engineering for his guidance and valuable suggestions which helped us to complete this project. We also thank our Project coordinators **Ms Supriya Salian** and **Dr Saumya Y M**, Dept of CSE, for their consistant encouragement.

We owe a profound gratitude to **Dr Sridevi Saralaya**, Head of the Department, Computer Science and Engineering, whose kind support and guidance helped us to complete this work successfully

We are extremely thankful to our Principal, **Dr Rio D'Souza**, Director, **Rev. Fr Wilfred Prakash D'Souza**, and Assistant Director, **Rev. Fr Kenneth Rayner Crasta** for their support and encouragement.

We would like to thank all faculty and staff of the Deprtment of Computer Science and Engineering who have always been with us extending their support, precious suggestions, guidance, and encouragement through the project.

We also extend our gratitude to our friends and family members for their continuous support.

Abstract

The need for reliable infant monitoring systems has grown due to the high demands of modern parenting and the importance of ensuring infant safety. This project presents a "Cloud-Based Smart Monitoring System for Baby Health and Safety," which monitors key health metrics such as body temperature, heart rate, room temperature, humidity, and posture. By providing real-time notifications and alerts, the system offers parents peace of mind and enhances infant safety.

Recent advancements in non-contact health monitoring utilize technologies like remote photoplethysmography and computer vision for detecting health parameters. However, existing systems often lack comprehensive capabilities or rely on contact-based sensors that may cause discomfort to infants. This project overcomes these challenges by integrating contactless sensors and machine learning techniques, creating a holistic and user-friendly monitoring solution.

The methodology involves developing a mobile application that interacts with a cloud-based system and sensors to analyze infant health data in real time. The system employs computer vision algorithms to monitor baby posture and detect unsafe positions, such as tummy sleeping, potentially preventing sudden infant death syndrome (SIDS). Experimental results confirm the system's reliability and accuracy under various environmental conditions, providing immediate alerts during abnormalities.

This work significantly enhances infant safety by reducing the need for constant parental monitoring while offering peace of mind. The system demonstrates a valuable contribution to infant health care by combining advanced technology with practical usability.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The health and safety of infants are critical concerns for parents, particularly when they are unable to provide constant supervision due to other responsibilities. One of the major risks to infants during sleep is Sudden Infant Death Syndrome (SIDS), which can occur if the baby unknowingly assumes an unsafe sleeping posture. In addition to posture, environmental factors like temperature, humidity, and the baby's health indicators—such as body temperature and heart rate—can have significant impacts on the baby's well-being. The lack of real-time, comprehensive monitoring systems makes it difficult for parents to detect these risks in time. This project, Cloud-Based Smart Monitoring System for Baby Health and Safety, is designed to bridge this gap by leveraging advanced software algorithms and cloud-based solutions to provide real-time monitoring of a baby's health and surroundings. With the integration of multiple sensors and a camera, the system ensures that any abnormalities, such as unsafe sleeping postures or sudden health changes, are detected and immediately communicated to the parents through a mobile application, helping to prevent potential health risks.

With the advancement of technology, there has been a growing interest in creating smart monitoring systems that go beyond simple video surveillance, incorporating health data analytics. This project aims to build on existing systems by introducing an innovative, software-focused approach that can simultaneously monitor and process multiple parameters, such as the baby's posture, heart rate, and environmental conditions. Using cloud computing for real-time data processing and alerts, the system will allow parents to track their child's well-being from any location, ensuring both the baby's safety and the parents' peace of mind. The focus on cloud infrastructure also allows scalability, enabling the system to be expanded with additional features and updates as needed.

1.2 Problem statement

To develop a cloud-based smart monitoring system that addresses the challenges parents face in continuously monitoring their infants, particularly when away from home. The system will use real-time data from sensors and video feeds to detect unsafe sleeping postures, abnormal body temperature, irregular heart rate, and environmental factors such as humidity. By using software-driven algorithms for analysis and alerting, the system will notify parents instantly of any concerns, thus preventing risks like Sudden Infant Death Syndrome (SIDS) and ensuring the infant's health and safety.

1.3 Objectives

The objectives of the proposed project work are:

1. To develop a mobile app that collects the body temperature of the baby and room temperature from the cloud, which is transmitted from the monitoring device.
2. To integrate computer vision technology to detect unsafe sleeping positions of the baby.
3. To create a user-friendly interface that allows parents to easily monitor real-time temperature readings regardless of the distance.
4. To deliver actionable notifications through app alerts when abnormal readings or unsafe sleeping position is detected.

1.4 Scope

The Cloud-Based Smart Monitoring System for Baby Health and Safety aims to provide a comprehensive, software-driven solution for real-time monitoring of a baby's health, environment, and movements. The project's scope includes the development of advanced algorithms to detect unsafe sleeping postures using computer vision, as well as the integration of sensor data from temperature, humidity, and heart rate monitors. The software will process this data in real-time through a cloud infrastructure, delivering instant alerts to parents via a mobile application whenever abnormalities are detected, such as a sudden change in the baby's sleeping position, body temperature, or crying. This monitoring will be continuous and remote, ensuring that parents receive timely notifications even when they are away from home.

The project is highly relevant in today's fast-paced world, where parents are often unable to supervise their children around the clock. The system can be applied in homes, daycares, or hospitals, giving caregivers real-time insight into the baby's well-being. By focusing on software for analyzing health and environmental data, this project addresses a significant gap in traditional baby monitors, which are often limited in functionality. The use of cloud technology ensures scalability, allowing for future enhancements such as the addition of more sensors or features, thereby making the system adaptable to evolving needs in infant care and monitoring, as well as regardless of the distance between the parent and the child, the vitals of the child can be monitored by the parents from any location.

Chapter 2

Literature Survey

2.1 IoT Based Smart Baby Monitoring System with Emotion Recognition Using Machine Learning

Identified Problem: This paper addresses the challenges faced by working parents in continuously monitoring their babies, particularly regarding environmental conditions and emotional states[1].

Methodology: The authors propose an IoT-based system that integrates various sensors to monitor room temperature, humidity, and emotional recognition through facial detection. Data is transmitted to the Blynk server, allowing real-time monitoring via a mobile application.

Implementation: The system employs a combination of IoT sensors and machine learning algorithms to detect a baby's cry and facial emotions. Notifications are sent to parents if abnormal conditions are detected.

Results: The implementation demonstrated effective monitoring capabilities, allowing parents to manage their time efficiently while ensuring their child's well-being.

Inference from Results: The system significantly alleviates the burden on parents by providing timely notifications and insights into their child's emotional state.

Limitations/Future Scope: While the system shows promise, it requires further development in terms of data security and privacy, as well as enhancing the accuracy of emotion recognition algorithms

2.2 IOT Based Baby Monitoring System

Identified Problem: This research focuses on creating an efficient and cost-effective monitoring system for infants that can operate in real-time[2].

Methodology: The authors utilize NodeMCU as the main control unit, integrating various sensors to monitor temperature, humidity, and crying. Data is uploaded to the AdaFruit BLYNK server for remote access.

Implementation:A prototype was developed that includes features like automatic cradle swaying when a baby cries and live video surveillance through an external webcam.

Results: The prototype proved effective in monitoring vital parameters, demonstrating simplicity and cost-effectiveness.

Inference from Results: The system's design allows for easy implementation in various settings, making it accessible for many families.

Limitations/Future Scope: Future improvements could focus on enhancing sensor accuracy and expanding functionalities to include more health parameters.

2.3 Internet of Things in Pregnancy Care Coordination and Management

Identified Problem: This systematic review highlights gaps in existing literature regarding IoT applications in pregnancy and neonatal care[3].

Methodology: The authors conducted a thorough review of IoT systems used in healthcare, focusing on their application in monitoring pregnant women and newborns.

Implementation: The review synthesizes findings from various studies to identify trends and challenges in IoT applications for maternal and infant health.

Results: It emphasizes the growing importance of IoT in healthcare but also points out significant limitations related to data security and sensor accuracy.

Inference from Results: The findings suggest that while IoT has transformative potential in healthcare, there are critical gaps that need addressing for effective implementation.

Limitations/Future Scope: Future research should focus on improving security protocols and enhancing user experience with IoT devices.

2.4 Development of an IoT based Smart Baby Monitoring System with Face Recognition

Identified Problem: This study tackles the issue of parental anxiety regarding infant safety by proposing an advanced monitoring system[4].

Methodology: The authors developed a system that combines face recognition technology with environmental monitoring sensors to provide comprehensive oversight of infants' conditions.

Implementation: The system utilizes machine learning algorithms for face recognition alongside traditional environmental sensors for temperature and humidity monitoring.

Results: The proposed solution showed high accuracy in recognizing faces and effectively monitored environmental conditions.

Inference from Results: This dual approach enhances parental confidence by providing real-time updates on both the child's identity and environmental safety.

Limitations/Future Scope: Challenges remain in ensuring robust performance under varying lighting conditions for facial recognition.

2.5 IOT Based Baby Monitoring System Smart Cradle

Identified Problem: This paper addresses the need for automated solutions in baby care, particularly for parents who cannot be physically present at all times[5].

Methodology: A smart cradle was designed using IoT technology to monitor key parameters such as crying, temperature, and humidity automatically.

Implementation: The cradle employs a microcontroller for automation, integrating sensors that trigger actions like swaying when a baby cries.

Results: Testing confirmed that the system effectively monitored environmental parameters while providing automated responses to crying.

Inference from Results: The design significantly reduces parental workload by automating basic care functions.

Limitations/Future Scope: Enhancements could include integrating more advanced health monitoring features such as heart rate tracking.

2.6 Smart Infant Baby Monitoring System Using IoT

Identified Problem: This paper highlights the alarming rates of Sudden Infant Death Syndrome (SIDS) attributed to inadequate monitoring of infants' health parameters during sleep. It emphasizes the necessity for a reliable system that can alert parents to potential dangers[6].

Methodology: The authors developed an IoT-based monitoring system utilizing Raspberry Pi along with various sensors designed to track temperature, heart rate, and sound detection. This multifaceted approach enables comprehensive monitoring of the infant's environment and health status.

Implementation: Data collected by the sensors is transmitted via SMS notifications to parents whenever abnormalities are detected. The system is designed for ease of use, ensuring parents can receive alerts without needing to constantly check their devices.

Results: The study reported a significant reduction in SIDS risk due to continuous monitoring capabilities. Parents expressed high satisfaction levels with the system's reliability and responsiveness, which provided peace of mind during nighttime hours.

Inference from Results: By allowing parents to monitor their infants remotely, this system enhances overall safety and reduces anxiety associated with infant care. The results underline the importance of real-time data access in preventing health emergencies.

Limitations/Future Scope: Future research directions include integrating advanced analytics capabilities that could predict health issues based on historical data patterns, thereby further enhancing preventive measures against SIDS.

2.7 Development of RTOs Based Internet Connected Baby Monitoring System

Identified Problem: Parents often lack real-time access to critical health metrics concerning their infants due to fragmented monitoring systems. This paper addresses this issue by proposing an integrated solution[7].

Methodology: The authors developed an internet-connected baby monitoring system that leverages various sensors for tracking environmental conditions such as temperature and humidity while also monitoring motion patterns of the baby.

Implementation: Data collected from multiple sensors is stored in a cloud database where it can be accessed by caregivers via a mobile application designed for user-friendly interaction. Alerts are generated when readings fall outside safe ranges.

Results: The study demonstrated reliable data transmission capabilities along with effective alert systems for abnormal readings, significantly improving parental engagement with their infants' health data.

Inference from Results: By providing continuous access to essential health metrics, this system empowers parents with information necessary for timely interventions during potential emergencies.

Limitations/Future Scope: Recommendations for future research include enhancing user interface design for better accessibility and exploring options for integrating additional sensors that could monitor more complex health indicators such as sleep quality or respiratory rates.

2.8 Smart Caregiving Support Cloud Integration Systems

Identified Problem: Current baby monitoring solutions often operate independently without sufficient integration between different functionalities leading towards fragmented experiences for parents trying to keep track of multiple aspects related towards child care[8].

Methodology: This paper discusses developing an intelligent baby monitoring system leveraging cloud computing technologies aimed at seamlessly connecting various sensor outputs into one cohesive platform accessible via mobile applications—allowing caregivers easy access whenever needed.

Implementation: Utilizing advanced cloud technologies ensures data collected from multiple sensors—including temperature monitors & motion detectors—are aggregated into one interface where alerts can be generated if any parameter deviates from established norms—ensuring comprehensive oversight at all times.

Results: Achieved better synchronization in data reporting led directly towards improved parental response times during emergencies—demonstrating how integration can enhance overall effectiveness significantly compared against fragmented approaches previously available on market spaces focused solely around single-functionality devices lacking holistic integration capabilities.

Inference from Results: This analysis highlights importance developing integrated systems capable delivering holistic insights rather than isolated metrics—ultimately fostering better decision-making processes among caregivers regarding child safety/wellbeing.

Limitations/Future Scope: Future work should focus on enhancing scalability options alongside exploring further integrations between different types of devices available today aimed at improving overall user experiences across diverse contexts.

2.9 Real time infant health monitoring system for hard of hearing parents

Identified Problem: Parents often lack immediate access to critical health metrics concerning their infants due to traditional monitoring methods being either too manual or inefficient at providing timely updates about changing conditions[9].

Methodology: This study proposes a real-time health monitoring system utilizing various IoT technologies capable of capturing vital signs along with environmental conditions present within the baby's room.

Implementation: Data collected from multiple sensors is processed in real-time before being made accessible through an intuitive mobile interface designed specifically for ease-of-use among caregivers.

Results: The prototype demonstrated effective performance by providing continuous updates about key indicators related directly towards overall infant wellbeing—allowing quick intervention when necessary.

Inference from Results: Real-time insights empower parents with knowledge needed during critical moments—significantly enhancing overall child safety measures taken within homes today.

Limitations/Future Scope: Future research directions may include exploring integration possibilities between healthcare providers' systems alongside existing frameworks aimed at ensuring comprehensive support mechanisms available whenever required.

Table 2.1: Comparison of Existing Work

Paper Title	Identified Problem	Methodology	Key Features	Limitations
IoT Based Smart Baby Monitoring System with Emotion Recognition[1]	Continuous monitoring challenges for parents	IoT sensors + ML	Emotion detection, notifications	Data security concerns
IOT Based Baby Monitoring System[2]	Need for real-time monitoring	NodeMCU + sensors	Automatic cradle swaying	Sensor accuracy issues
Internet of Things in Pregnancy Care Coordination[3]	Gaps in literature on IoT applications	Systematic review	Comprehensive analysis of existing works	Lack of usability studies
Development of an IoT based Smart Baby Monitoring System with Face Recognition[4]	Parental anxiety over infant safety	Face recognition + sensors	Real-time updates on identity & environment	Performance under varying conditions
IOT Based Baby Monitoring System Smart Cradle[5]	Automation needs in baby care	Microcontroller + sensors	Automated responses to crying	Limited health tracking features
Smart Infant Baby Monitoring System Using IoT[6]	High SIDS Rates; Inadequate Monitoring	Raspberry Pi + Sensors; SMS Notifications	Significant reduction in SIDS incidents; High parent satisfaction	Advanced analytics needed; Predictive capabilities
Development of RTOs Based Internet Connected Baby Monitoring System[7]	Lack of Real-Time Data Access	Multiple Sensors + Cloud Storage; User-Friendly App Design	Reliable alerts; Enhanced parental engagement reported	UI enhancements suggested; Additional sensor integrations
Smart Caregiving Support Cloud Integration Systems[8]	Lack integration between functionalities	Cloud computing technologies + sensor outputs; Mobile app	Data from sensors and cloud is aggregated into one interface	Scalability enhancment suggested
Real time infant health monitoring system for hard of hearing parents[9]	Lack immediate access to critical health metrics	IoT technologies capture vital signs	Effective performance by providing continuous updates	Suggested exploring integration with healthcare providers' systems

2.10 Proposed system

The Cloud-Based Smart Monitoring System for Baby Health and Safety is designed to ensure the well-being of infants through real-time monitoring of critical health parameters and environmental conditions. It integrates various IoT sensors to measure the baby's body temperature, room temperature, humidity, heart rate, and blood oxygen saturation (SpO2). A camera captures live video feeds, enabling the detection of unsafe sleeping positions using advanced computer vision algorithms. The system is powered by a Raspberry Pi, which collects and processes data, transmitting it to a cloud infrastructure for storage and analysis. A mobile application serves as the user interface, providing parents with real-time access to health data and alerts.

The system offers numerous benefits, including enhanced safety through continuous monitoring and immediate alerts for potential health issues, which can be crucial for preventing incidents such as Sudden Infant Death Syndrome (SIDS). Its intuitive mobile application ensures easy access to vital information, allowing parents to monitor their baby's health from anywhere. Additionally, the cloud-based architecture facilitates scalability for future enhancements. Overall, this proposed system significantly advances the intersection of technology and infant care, promoting a safer and more responsive environment for parents and their babies.

2.10.1 Importance of chosen project

The chosen project addresses a critical need for continuous monitoring of infants, a task that is particularly challenging for parents, especially when they are away from home. Infants are vulnerable to health risks that can arise unexpectedly, making it essential for caregivers to have reliable systems in place to monitor their well-being. By integrating various health sensors, video feeds, and cloud-based real-time analysis, the system ensures that parents are immediately alerted to potential health risks. This proactive approach to monitoring can help prevent life-threatening conditions such as Sudden Infant Death Syndrome (SIDS) and other critical health issues, ultimately providing parents with much-needed peace of mind. The project's significance is underscored by the growing demand for technology that can support busy families in maintaining the safety and health of their infants.

2.10.2 Novelty in Proposed project

The novelty of the proposed project lies in its comprehensive integration of multiple health parameters—such as body temperature, heart rate, SpO2, humidity, and more—with video-based posture detection. This data is processed in real-time using advanced cloud technology. Unlike most existing systems, which tend to focus on isolated health metrics or lack the functionality for real-time monitoring, this system offers a unified platform that simultaneously tracks various aspects of a baby's health. The use of machine learning algorithms for video analysis further distinguishes this project, as it allows for automatic detection of unsafe sleeping positions. This combination of features provides a more holistic approach to infant monitoring that is not commonly found in current market solutions.

2.10.3 Advancement of State-of-the-Art

The project advances the state-of-the-art by merging health monitoring with video-based posture analysis through the application of artificial intelligence techniques. Current solutions typically operate in silos, either relying solely on health sensors or focusing on video surveillance without integrating the two. This system's multi-faceted approach ensures comprehensive monitoring, as it not only tracks vital health parameters but also observes the baby's physical position. Additionally, leveraging cloud technology allows for scalable and remote access to the monitoring data, making the system more robust and future-proof. By employing cutting-edge technologies, this project aims to set new standards in the realm of infant health monitoring.

2.10.4 Differentiation from Existing Works

This project differentiates itself from existing works that focus solely on either wearable sensors or video-based monitoring by integrating both components into a single, cohesive system. This comprehensive approach makes it more effective in providing thorough monitoring of infants. The incorporation of deep learning techniques for posture detection, combined with real-time alerts, sets this system apart from those that rely merely on sensor-based monitoring. Furthermore, the use of a cloud infrastructure ensures seamless access to data from remote locations, allowing busy parents to stay informed about their baby's health at all times. This combination of features not only enhances the practicality of the system but also addresses the evolving needs of modern families.

Chapter 3

Software Requirements Specification

3.1 Functional Requirements

3.1.1 User Management

- Users will have the ability to create an account using email and password authentication. Account management features will include options to update personal information, reset passwords, and delete accounts.
- The application must implement secure login and logout procedures to protect user data, including multi-factor authentication for added security.

3.1.2 Data Monitoring

- The system will continuously fetch and display real-time data for baby parameters such as body temperature, room temperature, humidity, heart rate, and SpO2 levels through the integration of various IoT sensors.
- Users will have access to live video feeds from a camera connected to the Raspberry Pi, allowing them to visually monitor their baby at all times.

3.1.3 Alerts and Notifications

- The system will monitor predefined thresholds for all critical parameters, and users will receive immediate notifications (via push notifications or SMS) if any parameter, such as temperature or heart rate, exceeds safe levels.
- Utilizing computer vision algorithms, the system will analyze the baby's posture and send alerts if it detects unsafe sleeping positions, particularly if the baby is at risk of sleeping on their tummy.

3.1.4 Historical Data Access

- The application will allow users to access historical data and trends for all monitored parameters over selectable time periods. Users can view graphs and statistics to understand trends in their baby's health.
- Users will have the capability to export their data in various formats (e.g., CSV, PDF) for personal records or sharing with healthcare professionals.

3.2 Non-Functional Requirements

3.2.1 Usability

- The application must be designed with a user-friendly interface, ensuring that even non-technical users can navigate easily. Help sections should be readily available for guidance.
- The application should be usable even by parents with limited digital literacy.

3.2.2 Reliability

- The system should provide 99.9% uptime to ensure continuous monitoring, especially during critical periods when parents are not home.

- Implement data backup strategies to prevent loss of critical monitoring data and ensure that the system can recover from failures seamlessly.

3.2.3 Security

- Implement secure authentication protocols (e.g., OAuth2) to safeguard user accounts and personal information.

3.2.4 Scalability

- The system architecture must support scalability, enabling it to handle an increasing number of users and devices without compromising performance.
- Utilize cloud services that can easily scale resources such as multiple user handling and catering multiple devices at the same time based on demand.

3.3 User Interface Design

3.3.1 Layout

- The main dashboard will present an overview of the baby's current health metrics, including temperature, heart rate, and other relevant data, and it should be easy to interpret, with clear visual indicators.
- Users should be able to navigate effortlessly between different sections of the application, such as historical data views, alert logs, and settings.

3.3.2 Color Scheme and Branding

- The application should use a soothing color palette (e.g., soft blues and greens) to create a calming atmosphere for users, promoting comfort and ease.

- All branding elements, including logos and fonts, should be consistently applied throughout the app to enhance brand recognition.

3.3.3 Accessibility

- The design should ensure that users with disabilities can easily navigate the app, incorporating features such as screen reader compatibility and adjustable text sizes.

3.4 Hardware & Software Requirements

3.4.1 Hardware Requirements

- A Raspberry Pi (minimum Model 3B) will be used for processing video feeds and interfacing with IoT sensors, with a compatible camera module for video input.
- The system will include multiple sensors such as temperature sensors (e.g., DHT22), humidity sensors, heart rate sensors (e.g., MAX30100), and SpO2 sensors.
- A Raspberry Pi camera module will be utilized for live monitoring.

3.4.2 Software Requirements

- The system will run on Raspbian or any compatible Linux-based OS for the Raspberry Pi to support necessary libraries and applications.
- React Native will be used for mobile app development, allowing cross-platform functionality for both Android and iOS[10].
- Firebase by Google Cloud Platform will serve as the cloud service backend for real-time data storage and user authentication, providing scalability and ease of integration[11][12].
- OpenCV will be utilized for computer vision tasks, specifically for detecting unsafe baby sleeping positions, and MQTT will be used as the messaging protocol to facilitate communication between the Raspberry Pi and the cloud[13][14].

3.5 Performance Requirements

3.5.1 Response Time

- The application should provide real-time updates for health metrics with a maximum latency of 2 seconds to ensure timely alerts and monitoring.
- Live video feeds should load within 3 seconds to provide parents with immediate visual access to their baby.

3.5.2 Data Processing

- The system should continuously process and analyze health data to ensure timely alerts and notifications, maintaining performance even with multiple concurrent users.

3.6 Design Constraints

3.6.1 Technical Constraints

- The system must operate within the processing capabilities and memory limits of the selected hardware (Raspberry Pi).
- Must ensure that data storage and processing comply with relevant data protection regulations, such as GDPR or HIPAA, depending on the target market.

3.6.2 Environmental Constraints

- The system must function effectively in varying home environments, considering factors like Wi-Fi signal strength, which could impact data transmission and monitoring capabilities.

3.7 Other Requirements

3.7.1 Compliance Requirements

- The system must comply with health and safety regulations applicable to baby monitoring devices, ensuring that all hardware components are safe for use around infants.

3.7.2 Documentation

- Comprehensive user manuals must be provided to assist users in setting up and using the monitoring system effectively.
- Detailed technical documentation should be created for future maintenance and potential upgrades, outlining system architecture and component specifications.

Chapter 4

System Design

This chapter outlines the system design for the Cloud-Based Smart Monitoring System for Baby Health and Safety, detailing the system's architecture, functionality, control flow, access layers, and user interface design. Each section includes design diagrams, descriptions, and an explanation of how they apply to the project.

4.1 Abstract Design

4.1.1 Architectural diagram

The architectural diagram in Figure 4.1 outlines the Cloud-Based Smart Monitoring System for Baby Health and Safety, showcasing how various components interact to provide a comprehensive baby monitoring solution.

1. **User Interaction and Mobile Interface:** The user initiates the monitoring process by accessing the mobile application interface, which includes a live camera feed. The user can request the recording of the baby's health metrics, which initiates data collection from multiple sensors and a real-time video feed. This interface also displays alerts and processed information regarding the baby's health.
2. **Sensing System:** The sensing system consists of multiple sensors, including Heart rate sensor, Baby temperature sensor, SpO2 (oxygen saturation) sensor, Humidity sensor, Camera sensor.

These sensors collect real-time data from the baby's physical environment, covering both health metrics and environmental factors. This raw data is then sent to the Processor for initial handling.

3. **Processor (Raspberry Pi):** Acting as an intermediary, the Raspberry Pi receives the data from the sensors and forwards relevant data to the **ML Model (Posture Detector)** to analyze the baby's sleeping posture. The processor handles the computational load and ensures efficient data processing before sending responses back to the mobile interface. If unsafe postures or abnormal metrics are detected, an alert is generated.

4. **ML Model and Database (Firebase):** The ML model, hosted externally as a pre-trained model, plays a crucial role in posture detection. Once the Raspberry Pi sends the data to the ML model, it processes the data and generates a response. Additionally, all data is stored and managed within a **Firebase** database, which securely holds historical health metrics and retrieves information as required by the user. Firebase also triggers alerts and notifications based on the ML model's responses and alerts from the sensor data.

5. **Alerts and Notifications:** The system is designed to provide timely alerts. When abnormal readings or unsafe sleeping postures are detected, the mobile app immediately notifies the user, allowing them to take quick action. The database and ML model work together to process and extract data, ensuring that alerts are accurate and meaningful.

6. **Summary:** This architecture ensures seamless data flow from sensing, processing, and alerting to final user interaction. This integrated approach enhances baby monitoring, helping caregivers make informed decisions to ensure the baby's safety and well-being.

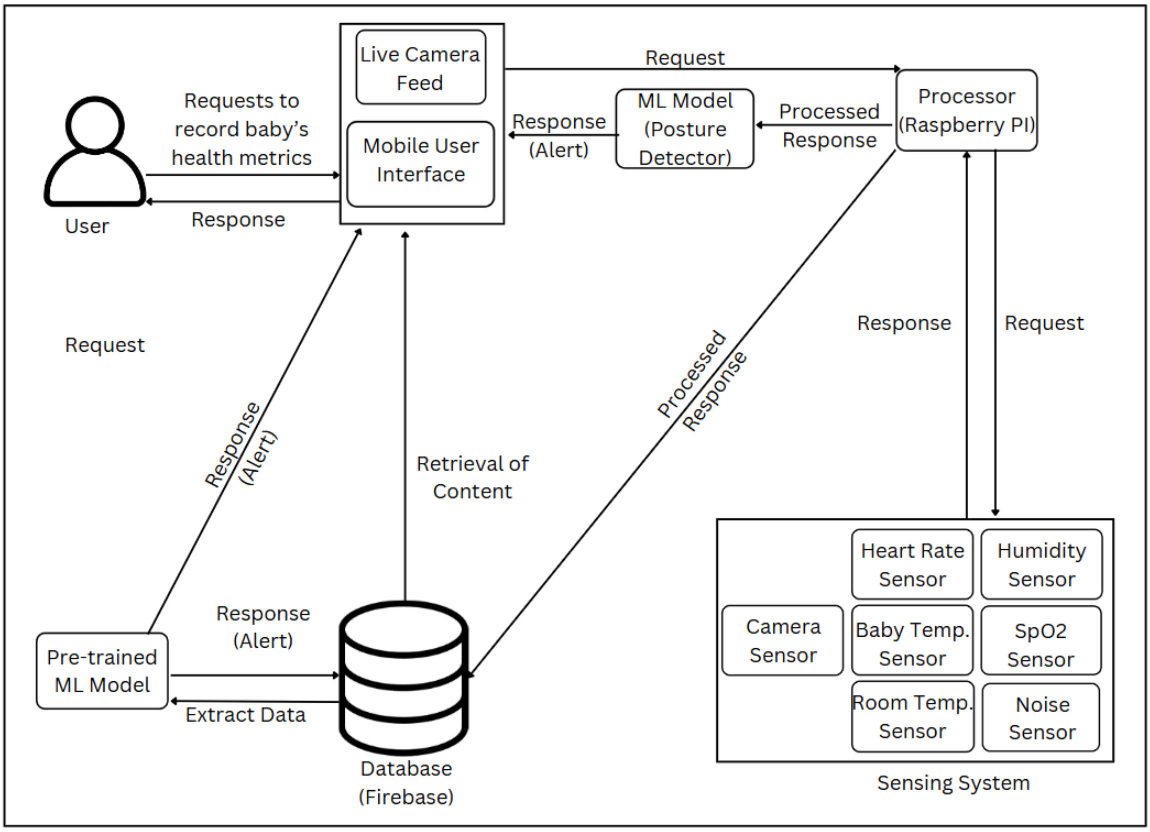


Figure 4.1: Architectural Diagram showing the interaction of various entities of the baby monitoring system.

4.1.2 Use Case Diagram

This use case diagram in 4.2 outlines the product designed to monitor a baby's health and safety through interactions with a **User (Caregiver)**, a **Sensing System**, and a **Medical Practitioner**.

1. **User:** The User can view a live camera feed and request the recording of health metrics, which are stored in the database. If any unsafe conditions (like risky postures) are detected, the system sends alerts to the User for immediate action.
2. **Sensing System:** The Sensing System collects health data (e.g., posture) and sends it to the monitoring system. The system processes this data to detect potential safety risks, triggering alerts when necessary.
3. **Medical Practitioner:** The Medical Practitioner reviews health

metrics stored in the system and provides insights or recommendations, which the system relays to the User to support safe caregiving.

Overall, the system combines real-time monitoring, data storage, and expert feedback to ensure the baby's health and safety effectively.

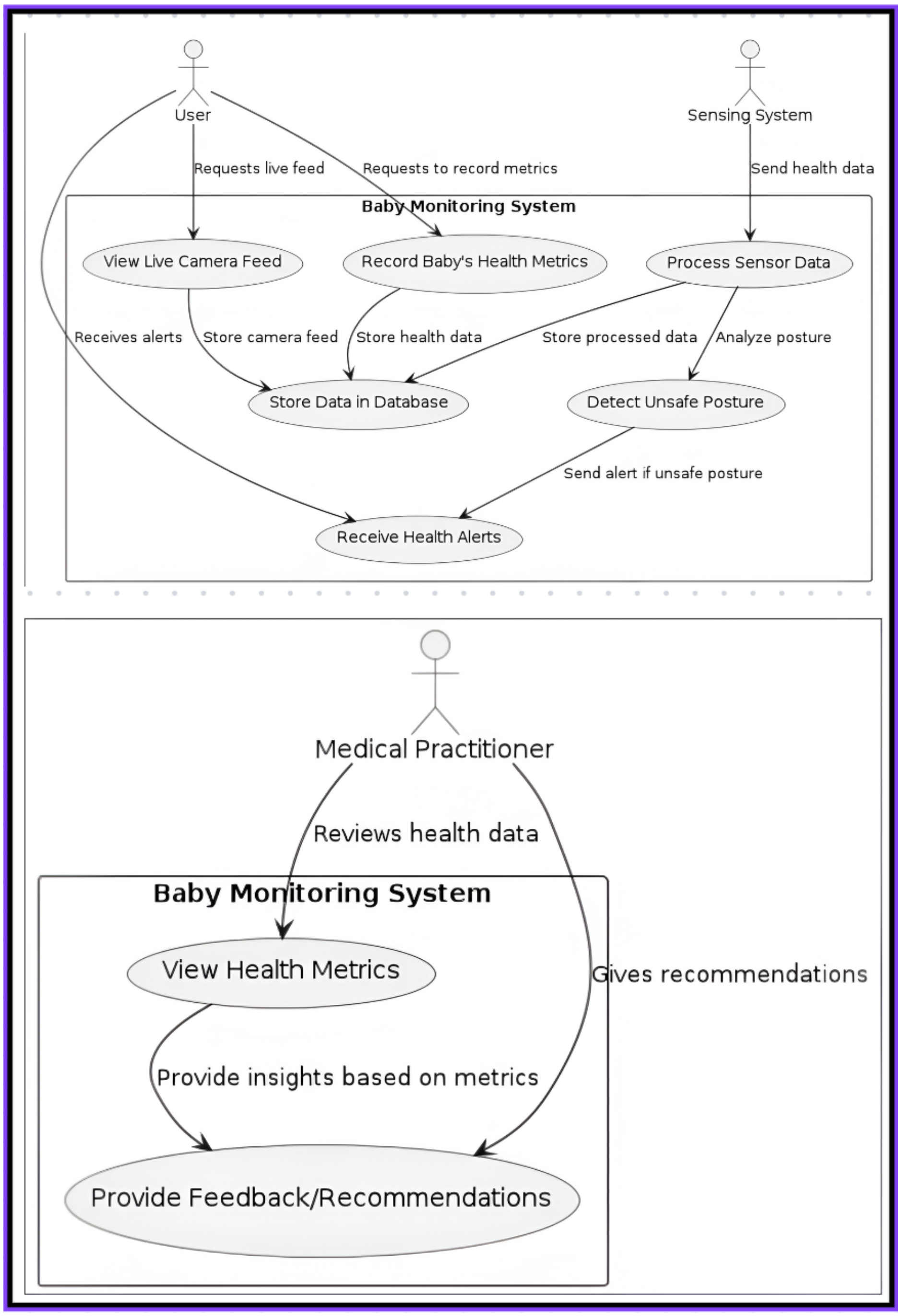


Figure 4.2: Use Case Diagram showing the interaction between users and the system.

4.2 Functional Design

4.2.1 Sequence diagram

The sequence diagram shown in Figure 4.3 illustrates the workflow of a Cloud-Based Smart Monitoring System for Baby Health and Safety. Here's a breakdown of the interactions among various components in the system:

User Interaction: The user begins by opening the mobile application. This app is designed to fetch real-time data regarding the baby's health and environmental factors, helping to monitor the baby's well-being effectively.

Mobile Application Requests: Upon initialization, the mobile application communicates with the Sensing System to retrieve various health parameters. These include the baby's heartbeat, humidity levels, body and room temperature, SpO2 (oxygen saturation), and a live video feed.

Sensing System Data Collection: The sensing system gathers these metrics from the physical environment where the baby is located. It captures vital data such as heart rate, humidity, body and room temperature, and oxygen levels, along with the live video feed. This data is sent for processing.

Data Processing by ML Model: The data collected is then passed to the Machine Learning (ML) Model for analysis. The ML model processes the data to identify any potential risks, such as unsafe sleeping positions or abnormal readings that may indicate health concerns.

Data Storage: The processed data, including any alerts or historical records, is stored in a Database. This storage enables the system to maintain a log of health metrics over time, allowing caregivers to review past records and track trends.

Alerts and Notifications: If the ML model detects unsafe sleeping positions or abnormal values in the sensed parameters, it triggers an alert. This alert is sent back to the mobile application to notify the user, ensuring immediate awareness of any potential risks to the baby's health.

Viewing Results: Finally, the user can view real-time data and historical readings from within the mobile app. This user-friendly interface provides caregivers with comprehensive insight into the baby's health, allowing for timely intervention if needed.

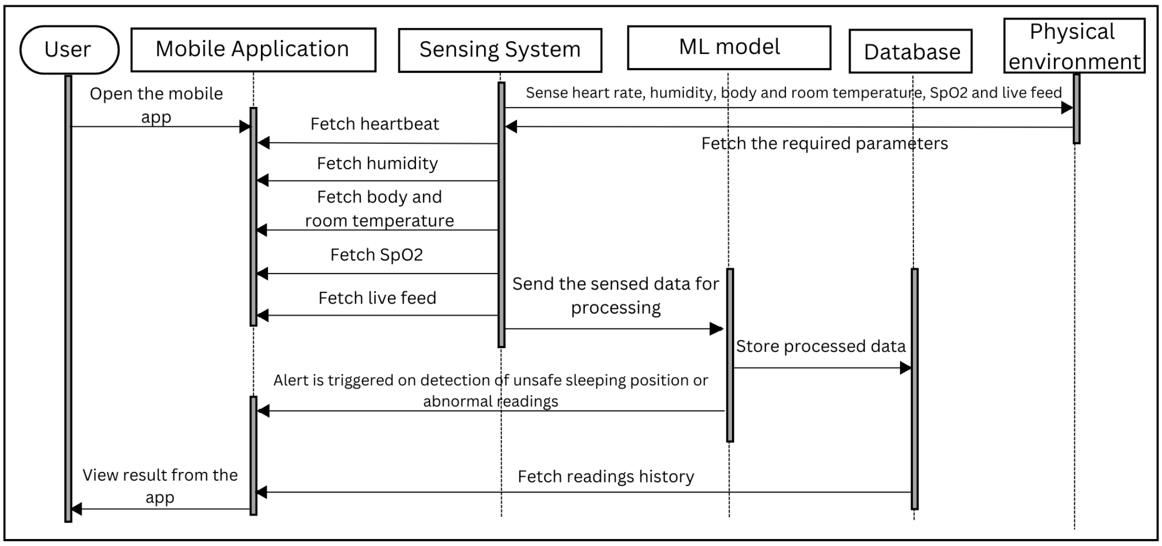


Figure 4.3: Sequence diagram showing the timeline of interaction between different entities in the system

4.3 Presentation Layer Design

4.3.1 User Interface Flow Design

The user interface flow for the application (Figure 4.4) is designed to provide parents with an intuitive and seamless experience for monitoring their baby's health and environment. The following steps describe the flow from the onboarding process to the profile management:

1. **Onboarding Screen:** Upon launching the app, the user is welcomed with the onboarding screen, which introduces the app's primary function, ensuring peace of mind for every parent by offering baby health monitoring in the palm of their hand. This screen includes an image carousel showcasing key features like monitoring baby movements, temperature, humidity, and more. The user is prompted to continue with email to proceed further.

2. **Sign-Up Screen:** New users are taken to the Sign-Up screen after the onboarding. Here, they can create an account by providing a username, email, and password. A simple and user-friendly form guides

the registration process. After filling in the details, they can tap Sign Up. If they already have an account, they are given the option to switch to the login screen.

3. **Login Screen:** Existing users access the Login screen, where they can sign in by entering their email and password. There's also a Forgot password option for those who might need help recovering their credentials. After entering valid credentials, the user taps Log In to access the app's features.

4. **Home Screen:** After logging in, the user is taken to the Home screen. This screen displays key baby health data, including real-time temperature tracking, humidity levels and SpO2 levels.

5. **Stats Screen:** The Stats screen gives a detailed view of the baby's health trends over time.

6. **Profile Screen:** The Profile screen offers personalized features for the user. They can call a doctor directly from the app if any issues are detected, access emergency contacts quickly for immediate assistance, modify details such as contact information, health preferences, and other personal settings to ensure the app is tailored to their needs.

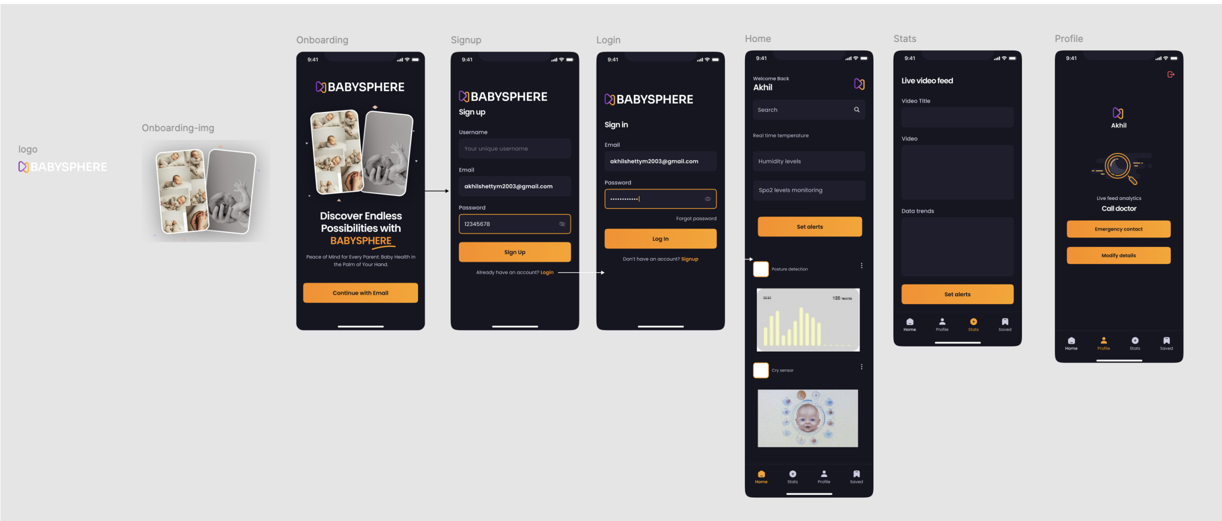


Figure 4.4: User Interface Flow Design of the mobile app designed using Figma

Chapter 5

Implementation

The implementation phase of the project "Cloud-Based Smart Monitoring System for Baby Health and Safety" was executed in a systematic manner, incorporating both hardware and software components. Below is a detailed explanation of the steps and methodologies adopted during this phase.

5.1 System Architecture Design

The system architecture was designed to ensure seamless integration of hardware and software components, focusing on real-time data processing and user-friendly interfaces. The architecture consisted of:

- **Hardware Module:**
  - Sensors for baby body temperature, room temperature, humidity, and heart rate.
  - A camera module integrated with Raspberry Pi for video streaming and posture detection.
- **Software Module:**
  - A React Native mobile application for parent notifications.
  - Firebase for real-time data storage and cloud integration.
  - Computer vision algorithms for unsafe posture detection using Mediapipe and OpenCV.

The architecture ensured data flow from the hardware module to the software application, with alerts generated based on predefined thresholds.

5.2 Hardware Implementation

The hardware implementation involved integrating various sensors and a camera module to collect real-time data from the baby's environment.

- **Sensors:**
  - MLX90614 IR Temperature sensor was used to monitor the body temperature of the baby.
  - MAX30102 heart rate ans SpO2 sensor was employed for continuous monitoring of the baby's health.
  - Data from these sensors were sent to a microcontroller unit for preprocessing.
- **Camera Module:**
  - A Raspberry Pi camera was used to capture live video for posture analysis.
  - The video feed was sent to Render server through a websocket connection and was processed for unsafe posture detection using Mediapipe library in Python.
- **Connectivity:**
  - The Raspberry Pi was configured to send the processed data to Firebase over a Wi-Fi connection.

5.2.1 CAD representation of the hardware setup

Shown in Figure 5.1 is the 3D pictorial representation of the hardware setup which consists of Raspberry Pi, power adaptor and all the sensors along with the camera.

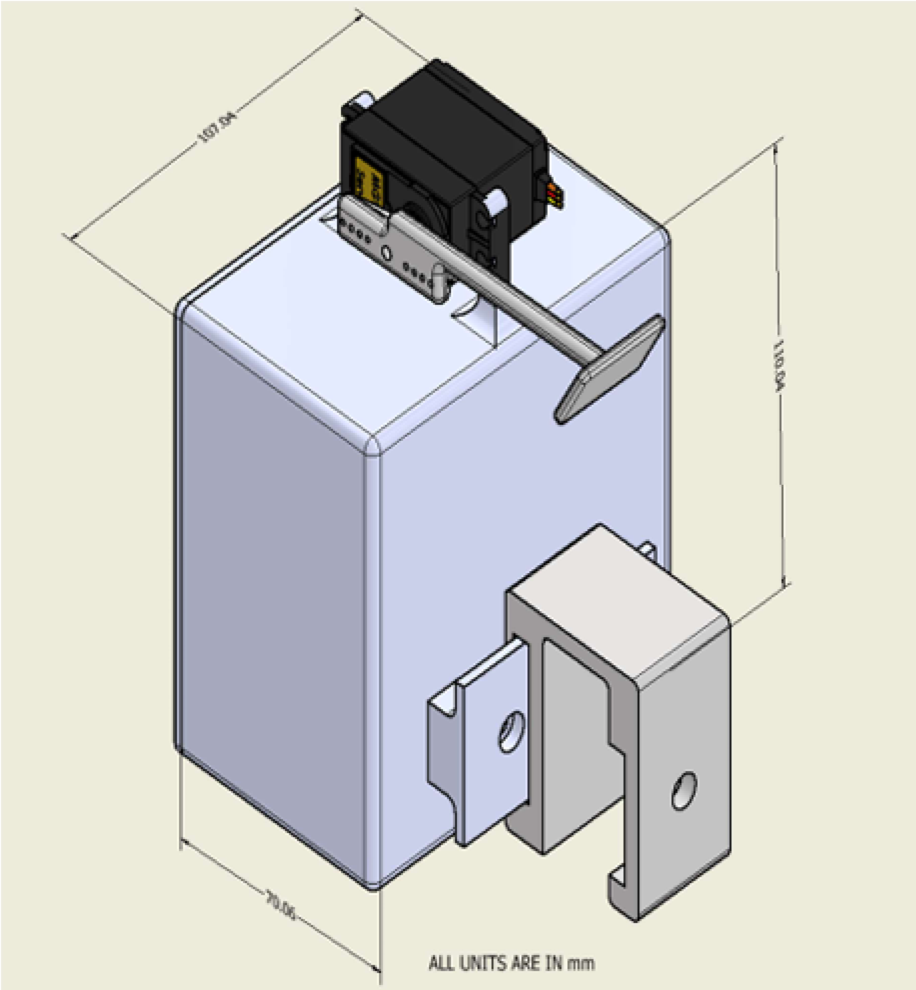


Figure 5.1: CAD Model of the hardware setup designed using Autodesk Inventor

5.3 Software Implementation

The software implementation focused on developing the front-end application, backend integration, and computer vision-based posture detection.

- **Mobile Application Development:**
  - Built using React Native for cross-platform compatibility.
  - Features included real-time alerts for abnormal conditions (e.g., fever, unsafe sleeping postures).
  - Integrated Firebase to fetch live data from the hardware module and display it on the user interface.
- **Backend Integration:**

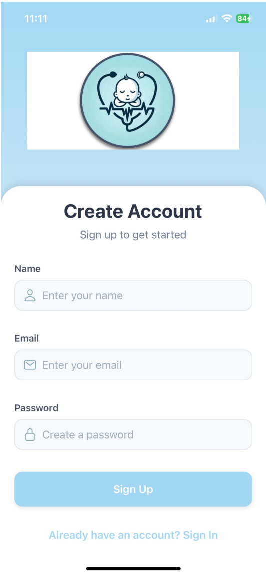
  - Firebase Realtime Database was configured for data storage and retrieval.
  - Cloud Functions in Firebase were used to process and trigger alerts based on predefined thresholds.

- **Posture Detection Algorithm:**
  - Implemented using Mediapipe and OpenCV.
  - Analyzed the baby's posture by detecting key landmarks (e.g., shoulders, eyes, nose).
  - Algorithms to identify tummy-sleeping and side-sleeping positions were fine-tuned based on testing data.

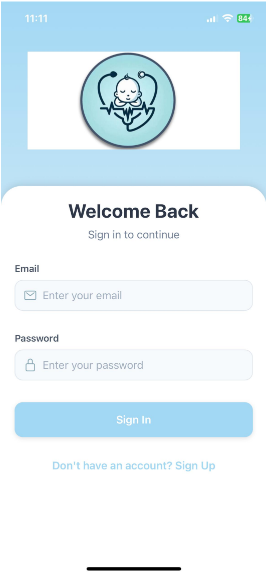
5.4 Screenshots of the mobile app

5.4.1 Sign Up and Sign In Pages

The Sign Up page (Figure 5.2a) allows new users to create an account by entering their details, while the Sign In page (Figure 5.2b) authenticates existing users using their email and password.



(a) Sign Up page

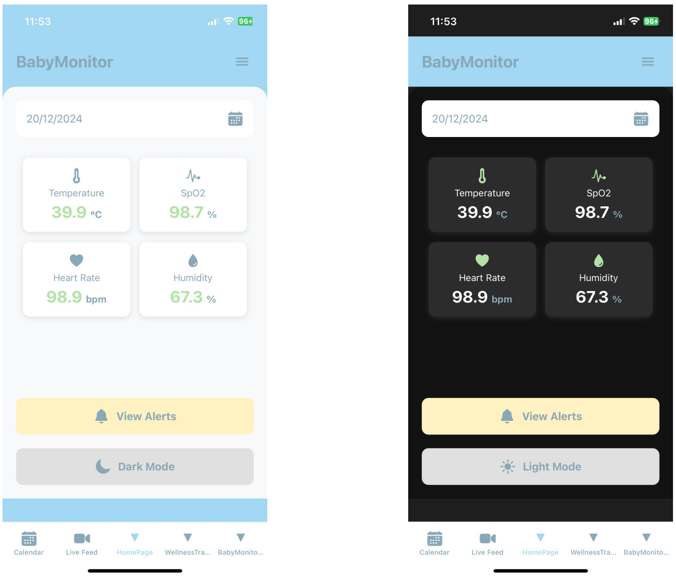


(b) Sign In page

Figure 5.2: Sign Up and Sign In pages of the baby monitoring app for user registration and user authentication.

5.4.2 Dashboard (Live Vitals)

The dashboard given in Figure 5.3 provides a real-time display of the baby's vital signs, including body temperature, heart rate, SpO2, and humidity.



(a) Dashboard - Light theme

(b) Dashboard - Dark theme

Figure 5.3: Comprehensive real-time display of vital health parameters.

5.4.3 Graph Page (Vitals History)

Figure 5.4 visualizes the historical fluctuations in the baby's vitals through interactive graphs. Parents can track trends over time to identify patterns or anomalies in the baby's health.

Figure 5.4: Graph page for detailed visualization of historical health data trends.

5.4.4 Live-feed Page

The live-feed page in Figure 5.5 streams video from the camera in real time, allowing parents to monitor the baby's movements and posture. This feature helps ensure the baby's safety by detecting abnormal sleeping positions or other concerns.

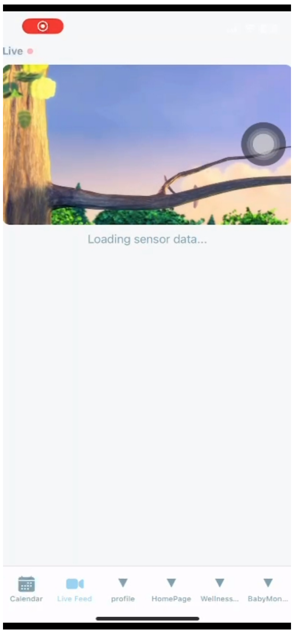


Figure 5.5: Live-feed page for seamless live video monitoring for enhanced baby safety.

5.4.5 Profile Page

Figure 5.6 contains personal information about the user and the baby. It includes details like the baby's name, date of birth, and other relevant data, offering a personalized experience.

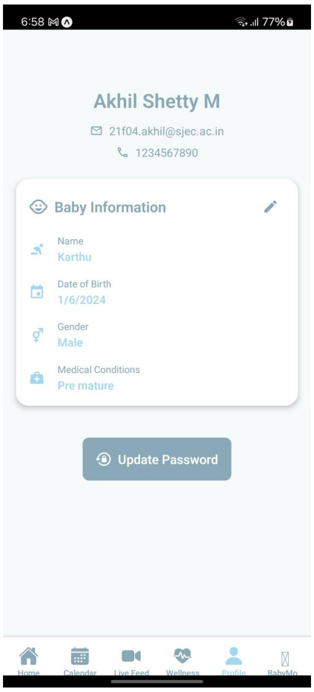
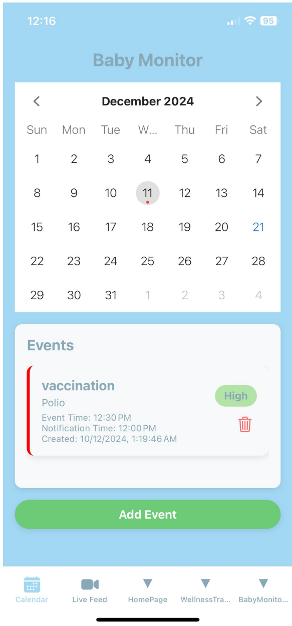


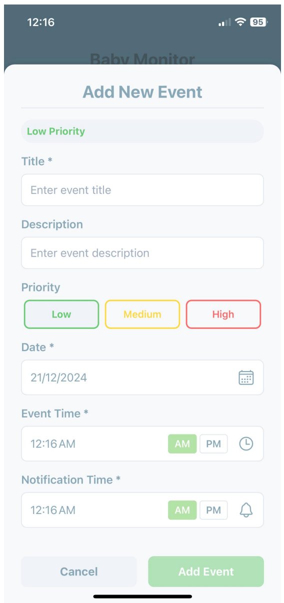
Figure 5.6: Profile page for consolidated user and baby information at a glance.

5.4.6 Calendar

The calendar page (Figure 5.7) allows parents to log and view important dates such as vaccination schedules, doctor appointments, or milestones. It serves as a central planner for managing the baby's care routine. Add event page in Figure 5.7b enables parents to add custom events to the calendar, such as reminders for medical check-ups, birthdays, or specific tasks. It helps streamline the baby's schedule efficiently.



(a) Calendar



(b) Adding events to calendar

Figure 5.7: Calendar for efficient management of critical dates and events.

Chapter 6

Implementation

6.1 Audio Extraction

Audio extraction is the process of isolating and extracting the audio content from a multimedia source, such as a video file. It involves separating the audio track from the accompanying video or other elements to obtain a standalone audio file representing the sound present in the source material.

Figure 6.1: code snippet for audio extraction

6.2 Speech Separation

SpeechBrain is an open-source framework

6.2.1 Sepformer

SepFormer is an algorithm for speech separation that utilizes self-attention mechanisms. It employs a transformer-based architecture to capture long-range dependencies and model the relationships between time-frequency points in the audio mixture, enabling the separation of multiple speech sources from the mixture.

Figure 6.2: code snippet for speech separation

6.3 Speech Enhancement

6.3.1 Lite Audio Visual Speech Enhancement

Lite AVSE algorithm is used for the separation and enhancement of the speech. The system includes two visual data compression techniques and removes the visual feature extraction network from the training model, yielding better online computation efficiency. As for the audio features, short-time Fourier transform (STFT) is calculated of 3-second audio segments. Each time-frequency (TF) bin contains the real and imaginary parts of a complex number, both of which used as input. Power-law compression used to prevent loud audio from overwhelming soft audio. The same processing is applied to both the noisy signal and the clean reference signal.

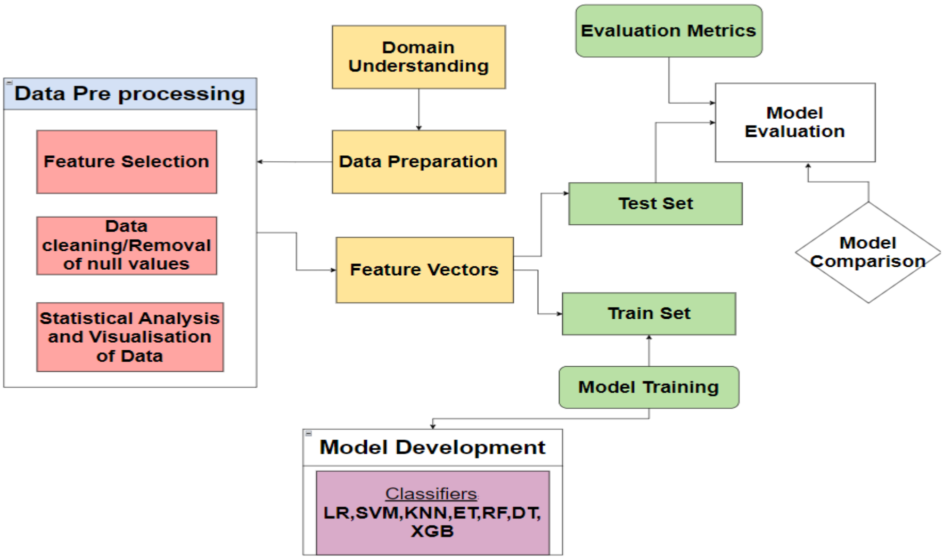


Figure 6.3: code snippet for speech enhancement using LAVSE

6.3.2 Spectral Subtraction

Spectral subtraction is a technique used in audio signal processing to reduce background noise from an audio signal. It involves estimating the noise spectrum from a noisy signal and subtracting it from the noisy spectrum to enhance the desired signal. The resulting spectrum is then transformed back into the time domain to obtain a cleaner audio signal as in Figure 6.4

Figure 6.4: code snippet for speech enhancement using spectral subtraction

6.4 Speaker Detection

The cv2 functions provide methods to load the pre-trained models, apply them to images or video frames, and draw bounding boxes around the detected faces. By leveraging cv2's face detection capabilities, you can automate tasks such as facial recognition, emotion analysis, or face tracking in various applications like surveillance, biometrics, or augmented reality.

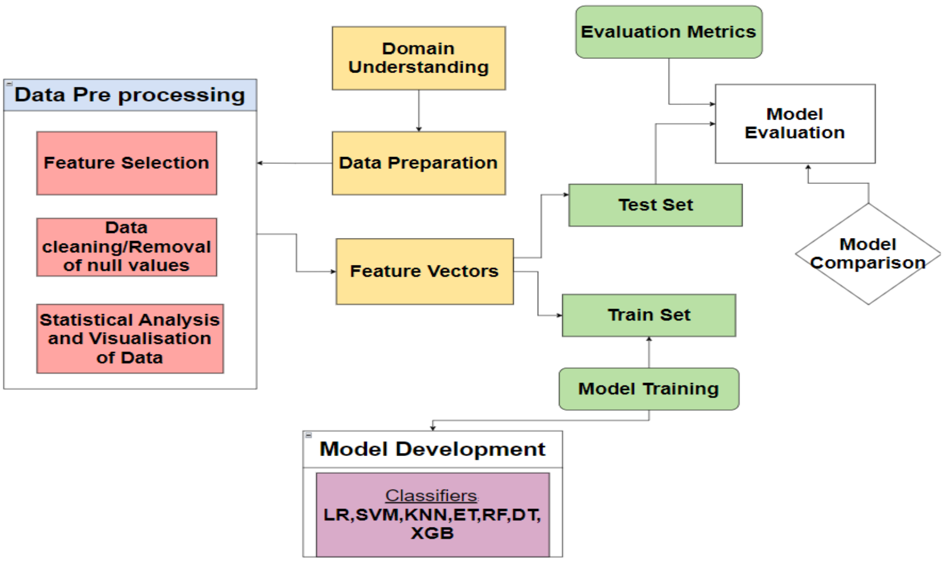


Figure 6.5: code snippet for speaker detection

Chapter 7

System Testing

Testing is a procedure of executing the program with unequivocal intension of **ref4** discovering mistakes, assuming any, which makes the program, fall flat. This stage is an essential piece of improvement.

It plays out an exceptionally basic part for quality affirmation and for guaranteeing unwavering quality of programming. It is the way toward finding the mistakes and missing operation and furthermore an entire confirmation to decide if the targets are met the client prerequisites are fulfilled.

The objective of testing is to reveal prerequisites, outline or coding blunders in the projects. Therefore, unique levels of testing are utilized in programming frameworks. The testing results are utilized amid upkeep. The testcases are shown in Figure 7.1

7.1 Testing Objectives

This area manages the points of interest in the various classes of the test which should be directed to approve capacities, imperatives and execution. This can be accomplished fundamentally by using the methods for testing, which assumes a crucial part in the improvement of a product.

7.2 Types of Testing conducted

The structure of the program is not being considered in useful testing. Test cases are exclusively chosen on the premise of the prerequisites or particulars of a program or module of program but the internals of the module or the program are not considered for determination of experiments**ref1**.

Figure 7.1: testcases

The program to be tried is executed with an arrangement of experiments and the yield of the program for the experiments is assessed to decide whether the program is executing not surprisingly. The accomplishment of testing in uncovering mistakes in projects depends basically on the experiments. There are two fundamental ways to deal with testing Black Box or functional Testing and White Box or structural testing. Table 7.1 shows the workflow.

Table 7.1: Work Flow

Sl No	**Work**	**Duration(in Weeks)**
1	Audio Extraction	1
2	Audio Enhancement using LAVSE	4
3	Audio Separation using Speechbrain	3
4	Noice Reduction using spectral subtraction	2
5	Image segmentation	3
6	Speaker Identification	5

Chapter 8

Results and Discussion

8.1 Face detection

Figure 8.1: Face detection

Above Figure 8.1 shows initial face detection process using opencv and dlib. It convert the image to grayscale, apply the model using cv2. detectMultiScale(), and draw bounding boxes around the detected faces using cv2.rectangle(). Display or save the result using cv2.imshow() or cv2.imwrite().

8.2 Speaker recognition

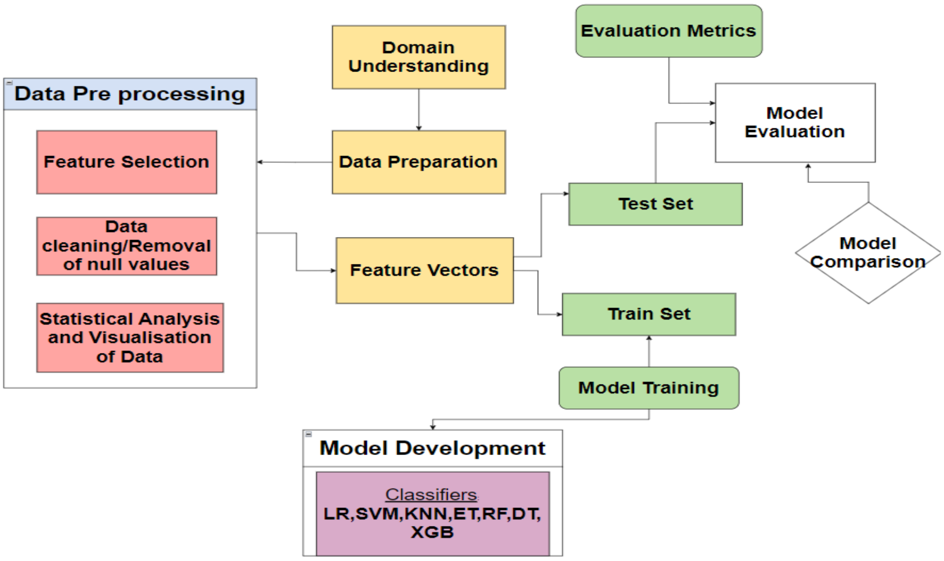


Figure 8.2: Speaker recognition 1,person 1

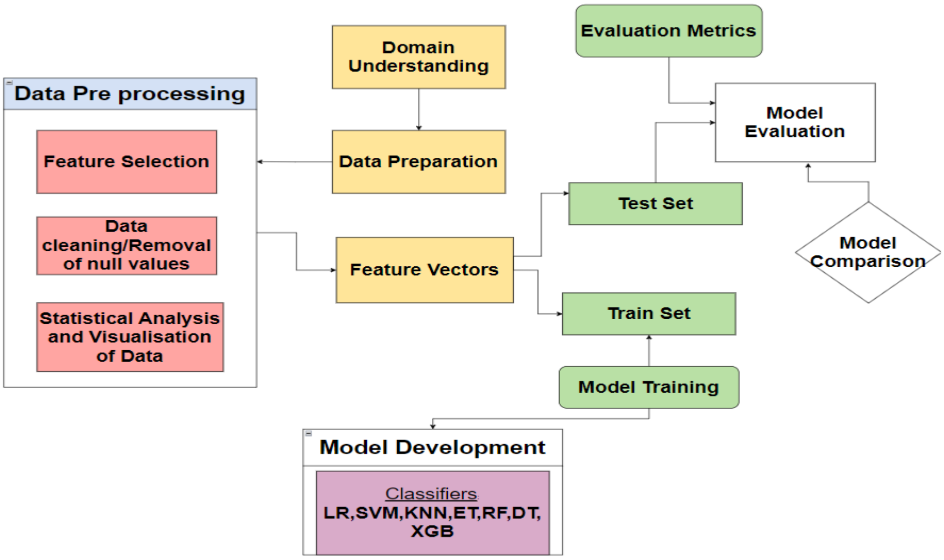


Figure 8.3: Speaker recognition 1,person 2

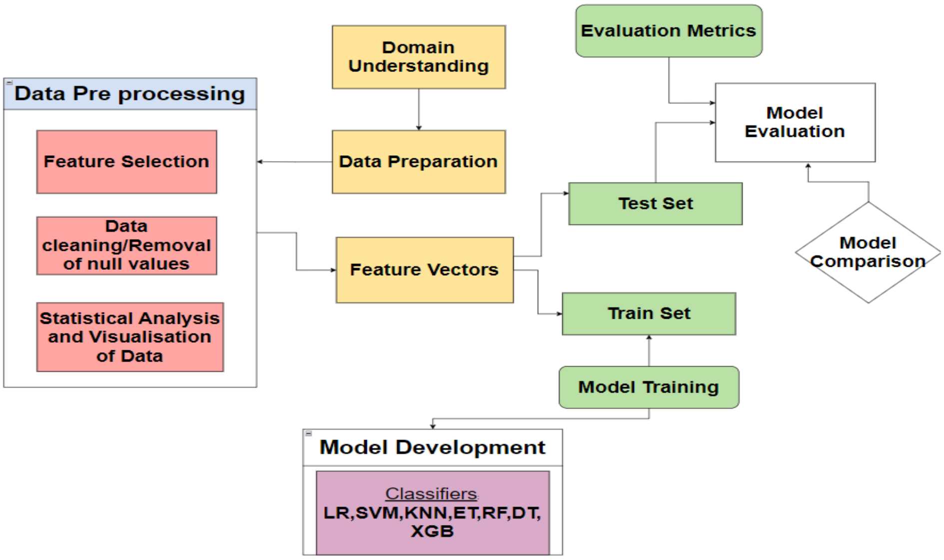


Figure 8.4: Speaker recognition 2,person 1

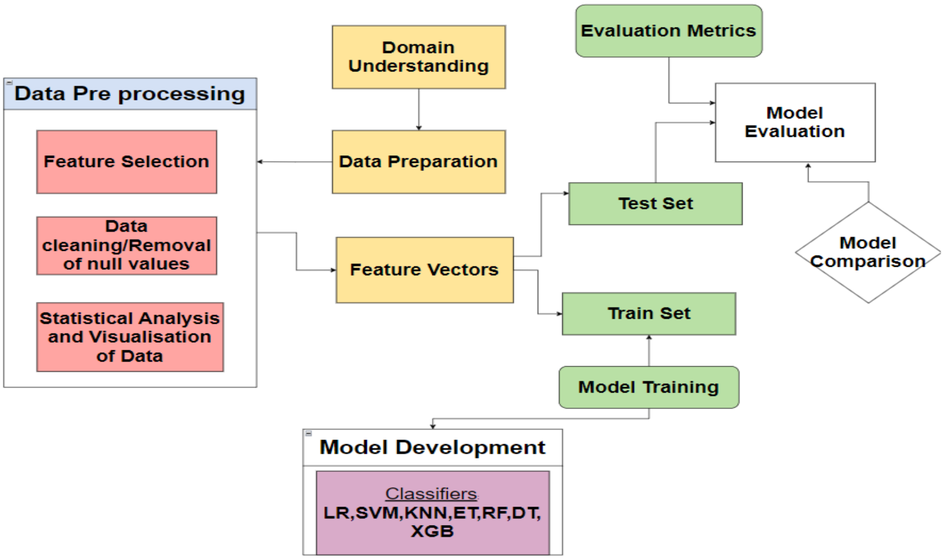


Figure 8.5: Speaker recognition 2,person 2

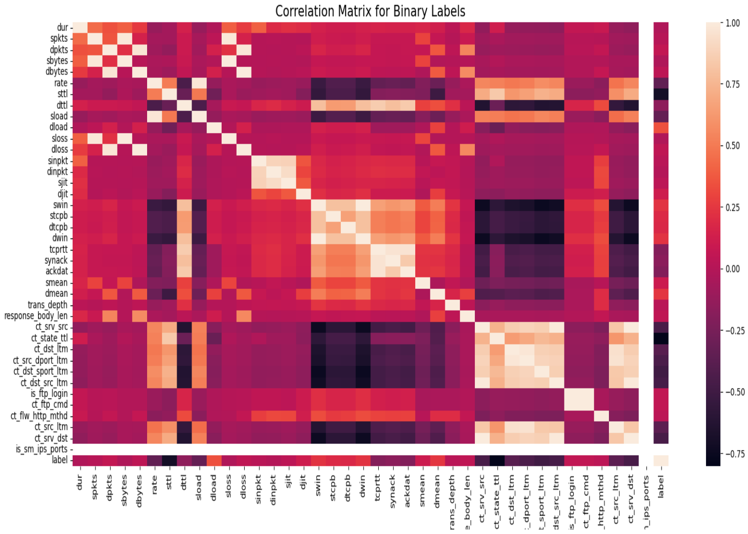


Figure 8.6: Speaker recognition 3,person 1

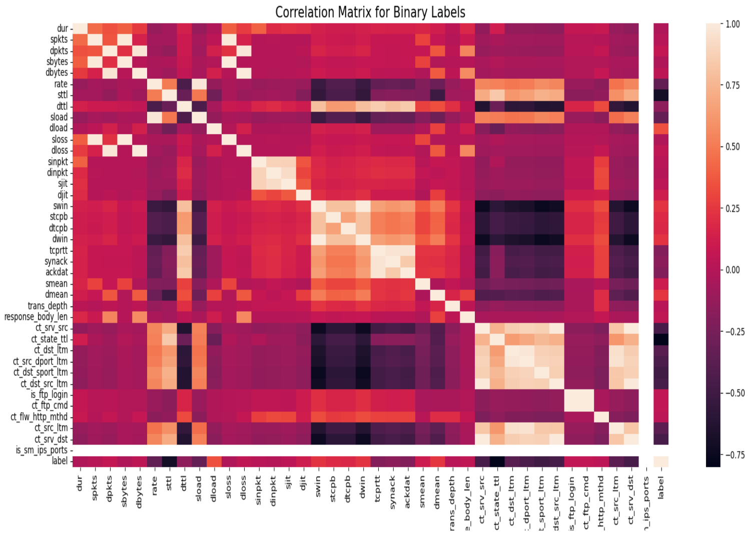


Figure 8.7: Speaker recognition 3,person 2

Above figures from 7.2 to 7.7 shows speaker recognition process using opencv and dlib.Speaker detection using cv2 and dlib involves utilizing dlib's pre-trained models along with cv2 functions to detect and locate human faces. By combining face detection with additional techniques such as audio analysis or lip movement tracking, speaker detection can be achieved in various applications like video conferencing or surveillance.

Chapter 9

Conclusion and Future work

The Project will help in narrowing the imprecise communication problem in real-time data using speech separation and speaker identification technique by Deep Learning and Image Processing algorithms. This will impact the communication and security sectors in a greater extent. Overall, this project aims to develop an application or method that can help to separate the audio-visual speech and enhance it based on speaker identification.
This project can be further developed as: • By incorporating more real-world testing and gathering feedback from individual units. • The system can be connected with communication devices or services to enable the users to communicate with others with ease.

This project has a great potential to make a positive impact on communication and security situations. Its continuous improvement will be important to make this impact even greater

References

[1] Hu, G., Yang, Y., Yi, D., Kittler, J., Christmas, W.J., Li, S., & Hospedales, T.M. "When Face Recognition Meets with Deep Learning: An Evaluation of Convolutional Neural Networks for Face Recognition," 2015 IEEE International Conference on Computer Vision Workshop (ICCVW), pp. 384–392, Dec. 2015, doi: 10.1109/iccvw.2015.58.

[2] Parkhi, Omkar, Andrea Vedaldi, and Andrew Zisserman. "Deep face recognition." In BMVC 2015-Proceedings of the British Machine Vision Conference 2015. British Machine Vision Association, 2015.

[3] Levitin, Anany. "Introduction to design and analysis of algorithms", 2/E. Pearson Education India, 2008.

[4] Prabhu, "Understanding of Convolutional Neural Network (CNN) — Deep Learning", URL: https://medium.com/RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148. Accessed on 23/07/2023

[5] L. Blanger and A. R. Panisson, "A Face Recognition Library using Convolutional Neural Networks," International Journal of Engineering Research and Science, vol. 3, no. 8, pp. 84–92, Aug. 2017, doi: 10.25125/engineering-journal-ijoer-aug-2017-25.

[6] R. Khedgaonkar, K. Singh, and M. Raghuwanshi, "Local plastic surgery-based face recognition using convolutional neural networks," Demystifying Big Data, Machine Learning, and Deep Learning for Healthcare Analytics, pp. 215–246, 2021, doi: 10.1016/b978-0-12-821633-0.00001-5.

[7] P. J. Phillips, "A Cross Benchmark Assessment of a Deep Convolutional Neural Network for Face Recognition," 2017 12th IEEE Inter-

national Conference on Automatic Face &; Gesture Recognition (FG 2017), pp. 705–710, May 2017, doi: 10.1109/fg.2017.89.

[8] Z. Huang, J. Zhang, and H. Shan, "When Age-Invariant Face Recognition Meets Face Age Synthesis: A Multi-Task Learning Framework," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7278–7287, Jun. 2021, doi: 10.1109/cvpr46437.2021.00720.

[9] Z. Huang, J. Zhang, and H. Shan, "When Age-Invariant Face Recognition Meets Face Age Synthesis: A Multi-Task Learning Framework," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7278–7287, Jun. 2021, doi: 10.1109/cvpr46437.2021.00720.

References

[1] H. Alam *et al.*, "Iot based smart baby monitoring system with emotion recognition using machine learning," *Wireless Communications and Mobile Computing*, vol. 2023, no. 1, p. 1 175 450, 2023.

[2] Y. Singh, "Iot based baby monitoring system," *International Journal for Research in Applied Science and Engineering Technology*, vol. 9, no. 12, pp. 2184–2190, Dec. 2021, ISSN: 2321-9653. DOI: `10.22214/ijraset.2021.39699`. [Online]. Available: `http://dx.doi.org/10.22214/ijraset.2021.39699`.

[3] M. M. Hossain *et al.*, "Internet of things in pregnancy care coordination and management: A systematic review," *Sensors*, vol. 23, no. 23, p. 9367, Nov. 2023, ISSN: 1424-8220. DOI: `10.3390/s23239367`. [Online]. Available: `http://dx.doi.org/10.3390/s23239367`.

[4] H. M. Ishtiaq Salehin, Q. R. Anjum Joy, F. T. Zuhra Aparna, A. T. Ridwan, and R. Khan, "Development of an iot based smart baby monitoring system with face recognition," in *2021 IEEE World AI IoT Congress (AIIoT)*, 2021, pp. 0292–0296. DOI: `10.1109/AIIoT52608.2021.9454187`.

[5] S. Joseph, A. Gautham.J, A. Kumar, and M. Harish Babu, "Iot based baby monitoring system smart cradle," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2021, pp. 748–751. DOI: `10.1109/ICACCS51430.2021.9442022`.

[6] M. R. Kumar, D. Harshitha, A. S. Prathyusha, C. Sangeetha, C. D. Rao, and D. Nitesh, "Smart infant baby monitoring system using iot," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 4, pp. 3003–3008, Apr. 2023, ISSN: 2321-9653. DOI: `10.22214/ijraset.2023.50764`. [Online]. Available: `http://dx.doi.org/10.22214/ijraset.2023.50764`.

[7] S. Mishra, "Development of rtos based internet connected baby monitoring system," *Indian Journal of Public Health Research Development*, vol. 9, no. 2, p. 345, 2018.

[8] D. T. P. Hapsari, Y. Nataliani, I. Sembiring, and T. Wahyono, "Smart caregiving support cloud integration systems," in *2024 International Conference on Smart Computing, IoT and Machine Learning (SIML)*, 2024, pp. 167–173. DOI: `10.1109/SIML61815.2024.10578217`.

[9] F. Aktaş, E. Kavuş, and Y. Kavuş, "Real time infant health monitoring system for hard of hearing parents," in *2016 Medical Technologies National Congress (TIPTEKNO)*, IEEE, 2016, pp. 1–4.

[10] *Introduction · React Native — reactnative.dev*, `https://reactnative.dev/docs/getting-started`, [Accessed 21-10-2024].

[11] *Cloud Computing Services — Google Cloud — cloud.google.com*, `https://cloud.google.com/?hl=en`, [Accessed 20-10-2024].

[12] *Firebase Documentation — firebase.google.com*, `https://firebase.google.com/docs`, [Accessed 20-10-2024].

[13] G. Singh, A. R. Shekhar, X. Yu, and J. Saniie, "Smart infant monitoring system using computer vision and ai," in *2023 IEEE International Conference on Electro Information Technology (eIT)*, 2023, pp. 1–6. DOI: 10.1109/eIT57321.2023.10187295.

[14] A. Okuno, T. Ishikawa, and H. Watanabe, "Rollover detection of infants using posture estimation model," in *2020 IEEE 9th Global Conference on Consumer Electronics (GCCE)*, 2020, pp. 490–493. DOI: 10.1109/GCCE50665.2020.9292052.

The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

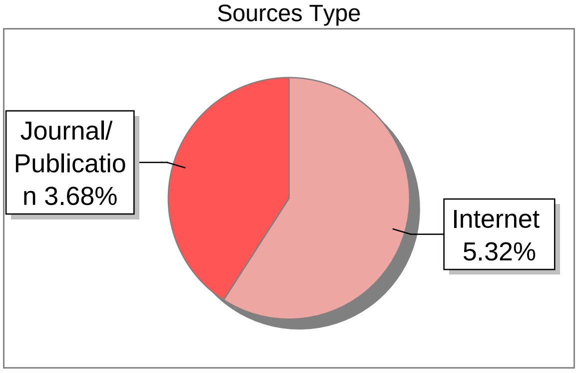
Author Name	Saumya
Title	Gesture-Enhanced Presentation Control for Education
Paper/Submission ID	3504295
Submitted by	saumyam@sjec.ac.in
Submission Date	2025-04-15 14:59:07
Total Pages, Total Words	6, 4458
Document type	Article

Result Information

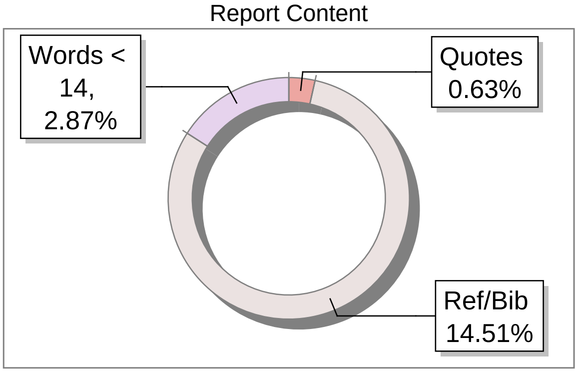
Similarity **9 %**

1 10 20 30 40 50 60 70 80 90

Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

Gesture-Enhanced Presentation Control for Education

Saumya Y M
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
saumyam@sjec.ac.in

JaishmaKumari B
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
jaishmab@sjec.ac.in

Colin Christon DCruz
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
colinchriston@gmail.com

Austin Dsouza
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
austindsz21@gmail.com

Daniel Loy Braggs
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
danielloy675@gmail.com

Elwin Jason Pereira
Dept. of CSE
St Joseph Engineering College
Vamanjoor, India
elwinjpereira02@gmail.com

***Abstract*—Presentation skills are vital in many areas of life. Giving presentations is probably a common experience for anyone, whether they are a worker, student, business owner, or employee of an organisation. The requirement to manage and manipulate the slides with a keyboard or other specialised device might make presentations seem tedious at times. Enabling users to control the slideshow with hand gestures is the aim of this work. Gestures have become increasingly common in human-computer interaction in recent years. Several PowerPoint functionalities have been attempted to be controlled by hand movements by the system. This system maps motions using multiple Python modules and uses machine learning to identify motions with minute variances. Creating the perfect presentation is becoming increasingly difficult due to a number of aspects, including the slides, the keys to switching the slides, and the audience's composure. An intelligent presentation system that is based on hand gestures makes it simple to update or modify the slides. Allowing viewers to explore and manipulate the slideshow with hand movements is the technology's main objective. The technique recognises various hand motions for a variety of tasks using machine learning. A means of recognition opens up a line of communication between people and machines.**

Index Terms*—Gesture, Gesture Recognition, Human Computer Interaction, Presentation, Annotation, Slide change.

I. Introduction

In today's ever-evolving education landscape, traditional classroom presentations are under- going a digital transformation. As digital learning gains prominence, there is a pressing need for a more dynamic and engaging means of controlling presentations. The existing tools not only limit the interactive potential of educators but also create accessibility challenges, particularly for those with physical disabilities. These issues hinder the effectiveness of teaching and can disrupt the flow of lessons [1]. Educators seek innovative ways to engage students using technology, making the "Gesture-Enhanced Presentation Control for Education" a highly relevant task.

979-8-3503-5059-3/24/$31.00 ©2024 IEEE

In AR/VR, hand tracking is essential for facilitating natural engagement and communication [10], and it has been a subject of intense discussion in the field of study. For many years, research has been conducted on vision-based hand pose estimation [2].The slides are editable by users. The interactive presentation system creates a more useful and approachable user interface for manipulating presentation displays by utilising state-of-the-art human-computer interaction techniques. When these hand gesture choices are used in place of a traditional mouse and keyboard control, the presentation experience is substantially improved. Nonverbal communication refers to the use of body language and gestures to convey a certain message. The Python framework was primarily employed in the construction of the system, together with NumPy, MediaPipe, openCV, and CV zone technologies. The goal of this approach is to improve presentations' usefulness and efficiency [5].

II. Literature Review

The paper [6] offers a thorough examination of computer vision based hand gesture recognition system. From mathematical algorithms like the row vector to machine learning approaches, it critically analyzes strengths and limitations. By scrutinizing methods such as edged image analysis and vector passing, the survey identifies research gaps and showcases deficiencies. This foundation justifies the chosen techniques, providing vital background and directionality. It aids in positioning the paper's contributions by benchmarking current challenges and showcasing field deficiencies, delivering crucial insights for hand gesture recognition understanding and system improvement. They employed the Row Vector Algorithm, the Diagonal Sum Algorithm, the Mean and Standard Deviation of the Edged Image, and the Edging and Row Vector Passing Algorithm.

This paper [2] introduces a system for controlling PowerPoint slides through hand gestures using a combination of a thermal camera and a webcam for robust hand tracking.

The methodology covers illumination invariant hand region extraction, gesture recognition through skin segmentation and SVM classification, and slide control mappings. Experiments demonstrate high accuracy in classifying gestures like swipe left and right to switch slides. The literature review analyzes existing research in gesture recognition and hand tracking, identifying challenges in accuracy and processing lag. The conclusion sums up key innovation, potential applications in interactive presentations, and limitations like small gesture vocabulary, suggesting enhancements through multidimensional dynamic time warping. The methodology included OpenCV, Haar Cascade Classifier, Skin Color Segementation, and Gesture Recognition.

This paper [3] introduces an innovative vision based hand gesture recognition system designed for PowerPoint presentation control, encompassing both static and dynamic gestures. The literature review scrutinizes existing methodologies, highlighting limitations in accessibility and vocabulary across various approaches. Leveraging a sophisticated 7-layer convolutional neural network (CNN) built upon the 20BN-Jester baseline, the system extracts spatial and temporal features from dynamic hand gesture video frames, significantly improving accuracy. The training process, conducted on the 20BN-Jester dataset using PyTorch on a GPU system, results in a highly accurate model capable of real-time classification. The methodology involves multi-phase processing, from opening a PowerPoint presentation to capturing live webcam video of hand gestures, transformed into a 20-frame image array for classification. Python is the chosen programming language, with Tkinter for GUI, PyTorch for deep learning, OpenCV for computer vision, and PyAutoGUI for simulating virtual keyboard keypresses, ensuring a robust and versatile integration of functionalities. The system recognizes diverse gestures contributing to enhanced accessibility and user-friendliness in PowerPoint presentations. Their methodology used Convolutional Neural Network Architecture, 20BN-Jester dataset, multi-phase approach, PyTorch and PyAutoGU.

A hand gesture-controlled virtual mouse system for seamless human-computer interaction is presented in this paper [4]. Using a webcam to record the user's hand movements, the system uses computer vision and machine learning models to detect and identify pointing and clicking actions in real-time. These predicted hand poses are seamlessly translated into virtual cursor operations, allowing touchless spatial control.The literature review traces the evolution of gesture recognition techniques from initial glove-based tools to modern solutions, analyzing pros and cons of past approaches. The proposed methodology addresses limitations like hardware restrictions and system lag by blending MediaPipe, speech recognition, and natural language processing for an efficient and responsive interface. The Algorithms and Tools used here include Google's MediaPipe, Single Shot Detector model, Hand Landmark model.

This paper [5] presents a system for controlling presentations using hand gestures, built using OpenCV and Google's MediaPipe framework. A webcam captures video input of the user's hand gestures, which are recognized by MediaPipe. Specific gestures like raising different numbers of fingers are then mapped to control commands for the presentation - changing between slides, accessing a pointer to draw on slides, and erasing drawings. The main technical challenge discussed is accurately recognizing gestures with background noise and variations in lighting. The system is designed to provide an intuitive hands free way of controlling presentations that could be used in real-world scenarios with basic hardware. Key libraries utilized include OpenCV for image processing and frame detection, MediaPipe for gesture recognition, and NumPy for numeric computing to transform the inputs into outputs. Overall, it demonstrates a practical application of computer vision and gesture recognition to facilitate more natural human-computer interaction. The Algorithms and Tools used here are BlazePalm, Hand Landmark Model, Hidden Markov Models (HMM), K-means clustering, Fast Fourier Transform (FFT), Non-maximum suppression and Encoder-decoder models.

The so-called Virtual Whiteboard, which is based on electronic pens and sensors, is given in the paper [8] and may offer an alternative to contemporary electronic whiteboards. With the tool in hand, the user can write, draw, and manipulate the contents of the whiteboard with just his or her hands. It is not necessary to have extra equipment like infrared diodes, infrared cameras, or cyber gloves. Dynamic hand gesture recognition is the foundation for user interaction with the Virtual Whiteboard computer application. When examining a video feed from a webcam connected to a multimedia projector that displays content from a whiteboard, gestures are identified. Kalman filtering helps to track the positions of hands in the image. In the paper the hardware and software of the Virtual Whiteboard is discussed with a special focus on applying Kalman filters for prediction of successive hand locations. The effectiveness of Kalman filter-supported recognition was evaluated for the motions used to manage the contents of the whiteboard, and the efficiency without filtering is provided.

The problem of estimating the entire 3D hand shape and pose from a single RGB image is a new and difficult one that is tackled in this study [7]. The majority of existing techniques for 3D hand analysis from monocular RGB images are limited to guessing the 3D positions of hand keypoints; they are unable to accurately convey the 3D shape of the hand. On the other hand, the research describes an approach based on Graph Convolutional Neural Networks (Graph CNNs) that can reconstruct a complete 3D mesh of the hand surface, which includes more detailed information on the 3D shape and attitude of the hand. They provide a large-scale synthetic dataset comprising both 3D postures and ground truth 3D meshes in order to train networks under complete supervision. Using the depth map as a weak supervision in training, the researcher presented a weakly supervised method for fine-tuning the networks using real-world datasets without 3D ground truth. Through rigorous evaluations on their suggested new datasets and two public datasets, proposed research indicate that proposed technique can build accurate and reasonable 3D hand mesh and can accomplish superior 3D hand pose estimate accuracy when compared with state-of-the-art methods. The difficulties faced by patients receiving physical therapy are discussed in the paper [9], with a focus on the boredom of repeating exercises that may cause patients to lose enthusiasm. It offers a remedy in the shape of hand rehabilitation software, which makes use of hand gesture detection and recognition technologies to enhance patient engagement and enjoyment during rehabilitation. The MediaPipe Hands algorithm is used by the system to recognise gestures and detect hands.

The study [11] uses morphological processing and YCbCr thresholding to accomplish efficient gesture recognition for PowerPoint presentation control. The Hidden Markov Model is used to classify the gestures that have been identified. HMM is a statistical model that works well for tasks involving the recognition of patterns over extended periods of time.

The purpose of the paper [12] is to enable gesture-based control of PowerPoint presentations, and it does so by using multiple techniques. Machine learning algorithms are used in the study to identify and categorise hand gestures. By training the model to distinguish minor changes in movements, the system can accurately map these motions to specific actions, such as advancing or reversing slides. The Python programming language is used to implement the system, making use of Mediapipe and OpenCV packages.

The paper [13] provides a new way for controlling PowerPoint presentations using static hand gestures. This technique uses a webcam to record hand motions, making it a useful and user-friendly solution. The thinning method, a method for processing and analysing hand forms, is introduced in this study. The number of elevated fingers is determined by using the hand form parameters that are extracted using this procedure. This novel method improves gesture recognition precision. The fact that the suggested approach doesn't need any extra gear, like gloves, markers, or other gadgets, is one of its best qualities. This improves the system's accessibility and usability by enabling users to interact with their presentations using just their hands.

III. System Design

A. Architectural Diagram

The architectural design for gesture-enhanced presentation control as displayed in the above Fig. 1 begins with the webcam capturing the user's hand gestures, serving as the primary input method. OpenCV processes the video feed, extracting critical details such as hand position and shape. These details are then analyzed by MediaPipe, which employs sophisticated algorithms to recognize specific gestures based on predefined patterns. Following recognition, the identified gestures are relayed back to the presentation software, where they are interpreted into actions such as navigating slides or activating multimedia elements. This process involves several intermediary steps, including video capturing, framing, and hand detection as well as frames filtering to enhance accuracy. Feature extraction distills relevant information from the recognized gestures, which are then classified into predefined actions. The architecture further encompasses gesture mapping, where these classified gestures are matched with corresponding presentation control functions. Ultimately, the presentation control component interfaces seamlessly with the software, executing the mapped functions based on the recognized gestures. Throughout this interaction, the user plays a central role, activating the webcam input device and performing hand gestures within its view to control the presentation. Feedback mechanisms such as audible or visual signals confirm gesture recognition and execution ensuring a smooth and intuitive user experience.

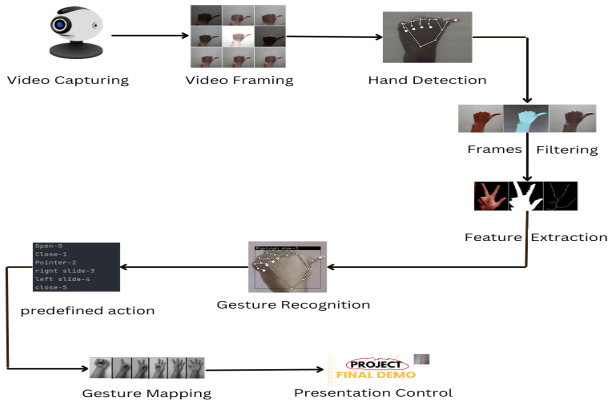


Fig. 1. Architecture Diagram

B. Phases of Gesture Recognition

- **Phase 1: Video Acquisition (Webcam):** This phase involves capturing the video stream from a webcam or any other camera input device. The quality and resolution of the captured video are crucial for accurate hand gesture recognition. The video stream serves as the input for subsequent phases in the gesture recognition system.
- **Phase 2: Video Pre-processing:** Video pre-processing is essential for preparing the captured video stream for hand gesture recognition. This phase typically includes several tasks such as:
    1) **Frame Extraction:** The continuous video stream is divided into individual frames for analysis.
    2) **Background Removal:** Removing the background from each frame helps isolate the hands from the rest of the scene, reducing interference and improving accuracy.
    3) **Hand Region Detection:** Identifying and delineating the regions of interest containing the hands within each frame. This can involve techniques like skin tone detection or background subtraction to locate the hands within the frame accurately.
- **Phase 3: Feature Extraction:** In this phase, relevant features are filtered and extracted from the detected hand region. These features provide the basis for identifying and interpreting different hand gestures. Frame filtering tasks may include:

1) **Frame Rate Reduction:**The system can sample the video at a reduced frame rate, such as every second or third frame, to focus on key points in time where meaningful gestures occur. This eliminates redundant data and allows the model to concentrate on frames that contain significant hand movements.
2) **Background Subtraction:**Background subtraction techniques are applied to isolate the hand region from the background. This helps to filter out irrelevant objects and noise, ensuring that only the hand gesture is processed. Common methods include Gaussian Mixture Models (GMM) or simple thresholding techniques that detect motion in the foreground.
3) **Blurring and Smoothing:**Applying blurring or smoothing filters (such as Gaussian blur) to the frames can help remove minor noise or irregularities in the video feed. This step enhances the quality of the input image, making the subsequent feature extraction process more robust.
4) **Skin Detection and Masking:** Skin detection algorithms can be applied to identify regions of the frame corresponding to human skin tones, focusing specifically on hand regions. This creates a mask that highlights the hand while ignoring non-skin areas, leading to more precise hand detection and feature extraction.

Feature extraction tasks may include:

1) **Hand Pose Estimation:** Determining the orientation and configuration of the hand(s) in the frame, including finger positions and hand shape. These landmarks provide a detailed map of the hand's orientation, configuration, and shape. Algorithms like MediaPipe Hands can accurately detect these landmarks in real time.
2) **Finger Tracking:** Tracking the movement and position of individual fingers within the hand region. By tracking the movement of each finger over time, the system can recognize dynamic gestures, such as a finger swipe or a specific finger motion sequence.
3) **Motion Trajectory Analysis:** Analyzing the trajectory of hand movements over time to detect gestures involving motion, such as swipes or gestures with directional components.Recognizing when a gesture starts and ends is essential for temporal gestures. This involves detecting the initial movement and when the hand comes to rest or returns to a neutral position.

- **Phase 4: Gesture Recognition:** Gesture recognition involves two main steps:
  1) **Gesture Classification:** Classifying the extracted features into specific gesture classes or commands. This can be achieved using machine learning models such as convolutional neural networks (CNNs) or rule-based algorithms.
  2) **Gesture Mapping:** Once gestures are classified, they are mapped to corresponding presentation control functions. For example, a specific hand pose or motion trajectory might correspond to commands like next slide, previous slide, or activate pointer mode.
- **Phase 5: Presentation Control:** In this final phase, the recognized gestures are used to control presentation software such as PowerPoint or Google Slides. This includes executing the mapped presentation control functions based on the recognized gestures, enabling seamless interaction with the presentation content. Presentation control functions may include slide navigation, pointer control, annotation or drawing tools activation, and other interactive features. The system interfaces with the presentation software through appropriate APIs or communication protocols to facilitate these actions.

IV. Implementation

The implementation of Gesture-Enhanced Presentation system commenced with the pivotal task of data collection and preprocessing. This is followed by selection and training the model for gesture recognition.

A. *Data Collection and Preprocessing:*

- **Data Collection:** A diverse set of hand gesture images representing various presentation commands like ”Next Slide”, ”Previous Slide”, ”Start Presentation” and ”Stop Presentation” is collected. Ensuring diversity, the dataset covers a wide range of hand shapes, positions, and lighting conditions. Quality assurance was maintained throughout the process to minimize noise and ensure clarity for effective model training.
- **Data Labeling:** Each image underwent a labeling process, where the images were manually annotated with the corresponding gesture it represents, including gestures like ”Next Slide”, ”Previous Slide” and others. Key points within the hand gestures, such as the position of the index finger, were also labeled to provide crucial information for model training. Consistency in labeling was paramount to avoid confusion during model training and evaluation.
- **Data Preprocessing:** Before training the model, the collected dataset was preprocessed to enhance image quality and remove noise. This involved resizing images to a standard size, normalizing pixel values for consistent brightness and contrast, and applying various data augmentation techniques to increase dataset diversity. Relevant features, such as identifying landmarks or keypoints like the coordinates of the index finger, were extracted for effective gesture recognition.
  - **Model Selection:** For the classification task TensorFlow's Keras API was opted. This choice was driven by the availability of pre-built deep learning models tailored for gesture recognition. By utilizing this framework, the model was able to efficiently

utilize computational resources and simplify the development process.

- **Training Data:** After preprocessing, the dataset was divided into training and validation sets. The goal here was to ensure a fair distribution of samples across different gesture classes, which is crucial for the model to generalize well to new, unseen data.
- **Model Training:** With TensorFlow as the training platform, a structured approach was followed. Techniques like transfer learning and fine-tuning of pre-trained models was applied to optimize the model's performance, especially given constraints such as limited training data or computational resources.
- **KeyPoint Classifier:** To facilitate the gesture recognition, the KeyPointClassifier class was implemented. This supported the deployment of a TensorFlow Lite interpreter, specifically configured with the chosen model file and thread specifications. With this setup, the landmark coordinates could be taken as input resulting in the accurate prediction of the class index, enabling precise classification of hand gestures based on these key points.

B. *Real-time Gesture Recognition:*

- **Camera Integration:** Integrate a camera module (e.g., webcam) with the system to capture real-time video input.
- **Gesture Detection:** Utilized libraries like OpenCV and MediaPipe to detect and track hand gestures in real-time video streams. Apply the trained gesture recognition model to classify detected gestures.
- **Feedback Mechanism:** Provide visual feedback to the user in real-time, indicating the recognized gesture and corresponding action.

C. *GUI Development:*

- **Graphical User Interface:** A user-friendly GUI was implemented using Tkinter framework, featuring an intuitive interface that enables users to interact with the presentation software using hand gestures.
- **Control Elements:** Implemented control elements such as buttons or sliders for common presentation functions (e.g., next slide, previous slide, start/stop presentation).
- **Integration with Gesture Recognition:** Integrated the gesture recognition module with the GUI, ensuring seamless interaction between gesture input and presentation control.

D. *System Integration and Testing:*

- **Component Integration and Function Testing:** All the components of the system, including gesture recognition, GUI, and presentation control logic are integrated. This is followed by a thorough testing to ensure the system functions as expected in different scenarios and environments. Test for accuracy, responsiveness, and robustness to variations in lighting conditions and hand gestures are performed.

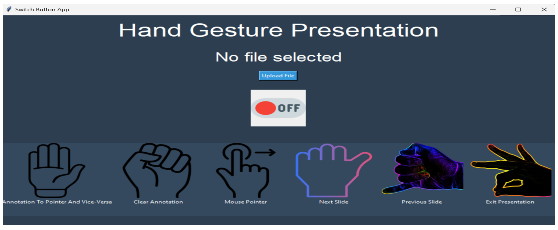


Fig. 2. GUI without input pptx file

V. Results and Discussion

The results depict the Graphical User Interface (GUI) of the proposed system as displayed in the Fig. 2, illustrating the initial state with no file selected. Additionally, it showcases an array of recognized hand gestures and their corresponding actions seamlessly integrated into the interface.

Users can seamlessly navigate through slides, move left or right, annotate slides with a red-colored line, and erase annotations as needed, providing a dynamic and engaging presentation experience.

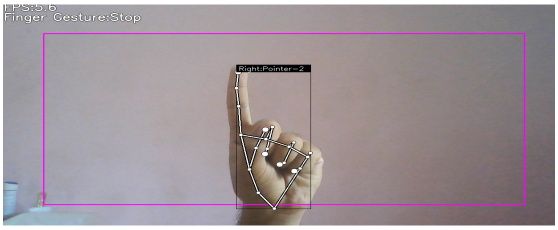


Fig. 3. Pointer to move cursor

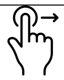
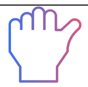
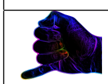
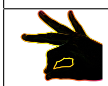
In Fig. 3, the background image detection with landmarks is showcased, highlighting the system's ability to accurately detect gestures and provide real-time feedback to the user. This functionality empowers programmers with a flexible approach to working with gestures, ensuring precise recognition and seamless integration into presentation control. TABLE I provides a comprehensive overview of the different gestures supported by the proposed system. The table outlines the specific actions associated with each gesture, ensuring users understand how to utilize them. This detailed description facilitates a deeper comprehension of each gesture's functionality and its intended role within the proposed system.

VI. Future Work

Despite the successful implementation of the Gesture-Enhanced Presentation system, there are several avenues for future work and enhancements:

1) **Integration with Voice Commands:** Expanding the system to support voice commands alongside hand

TABLE I
GESTURES SUPPORTED BY THE PROPOSED SYSTEM

Gesture	Action Performed	Description
	Switch Between Annotation and Pointer	This gesture allows users to smoothly transition between annotation and pointer modes, as well as back and forth from pointer to annotation mode.
	Clear Annotation	This gesture serves the purpose of clearing annotations made by the user previously.
	Mouse Pointer	This gesture serves a dual purpose: first, it enables users to navigate the cursor during presentations, facilitating seamless control and highlighting of key areas. Additionally, it empowers users to make annotations, jot down notes, and mark significant sections within the presentation, enhancing engagement and interaction.
	Next Slide	This gesture enables users to seamlessly transition to the next slide during presentations.
	Previous Slide	This gesture enables users to seamlessly transition to the previous slide during presentations.
	Exit Presentation	This gesture provides a way to the user to exit the presentation.

gestures would provide users with additional control options and further enhance the user experience. Integrating voice recognition technology would enable presenters to navigate slides and execute commands using natural language.

2) **Enhanced Gesture Recognition:** Continuously improving the accuracy and robustness of gesture recognition algorithms is crucial for ensuring reliable performance across different environments and hand poses. Further research and development in this area could involve exploring advanced machine learning techniques and leveraging larger datasets for training.
3) **Multi-Modal Interaction:** Exploring the integration of multiple modalities such as hand gestures, voice commands and facial expressions could lead to more immersive and interactive presentation experiences. By combining different input modalities one can create a more versatile and adaptable system that caters to a wider range of user preferences and abilities.

VII. Conclusion

In conclusion, the development of the Gesture-Enhanced Presentation system for education has been a significant endeavor aimed at revolutionizing the way educators and students interact with presentation materials. By leveraging hand gesture recognition technology, a user-friendly interface that allows presenters to control presentation slides seamlessly using intuitive gestures is proposed. This system offers an innovative and engaging approach to delivering educational content, enhancing the learning experience for both presenters and audiences. Through rigorous testing and iterative design improvements, it is ensured that the system meets the requirements of educational settings and delivers reliable performance.

References

[1] Bobo Zeng, Guijin Wang, Xinggang Lin. ”A Hand Gesture Based Interactive Presentation System Utilizing Heterogeneous Cameras ”. TSINGHUA SCIENCE AND TECHNOLOGY ISSNll1007-0214ll15/18llpp329-336 Volume 17, Number 3, June 2012.

[2] Rida Zahra , Afifa Shehzadi , Muhammad Imran Sharif , Asif Karim , Sami Azam , Friso De Boer , Mirjam Jonkman , Mehwish Mehmood.”Camera-based interactive wall display using hand gesture recognition”,Computational Intelligence and Neuroscience, 2022.

[3] Muhammad Idrees , Ashfaq Ahmad , Muhammad Arif Butt , and Hafiz Muhammad Danish. ”Controlling Power Point using Hand Gestures in Python”. PWebology (ISSN: 1735-188X) Volume 18, Number 6, 2021.

[4] M. Kasar, P. Kavimandan, T. Suryawanshi, and S. Abbad, “AI-based real-time hand gesture-controlled virtual mouse,” Australian Journal of Electrical and Electronics Engineering, pp. 1–10, Feb. 2024, doi: 10.1080/1448837x.2024.2313818.

[5] Hajeera Khanum, Dr. Pramod H B.” Smart Presentation Control by Hand Gestures Using Computer Vision and Google’s MediaPipe”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 09 Issue: 07 — July 2022 .

[6] Munir Oudah , Ali Al-Naji and Javaan Chahl “Hand Gesture Recognition Based on Computer Vision: A Review of Techniques”. IEEE conference on computer vision - 23 July 2020

[7] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. ”3D Hand shape and Pose Estimation from a Single RGB image”. In Proceedings of the IEEE conference on computer vision and pattern recog- nition, pages 10833–10842, 2019.

[8] M. Lech, B. Kostek, and A. Czyzewski, “Virtual Whiteboard: A gesture-controlled pen-free tool emulating school whiteboard,” Intelligent Decision Technologies, vol. 6, no. 2, pp. 161–169, Feb. 2012, doi: 10.3233/idt-2012-0132.

[9] Yeng, Angelina Chow Mei, et al. ”Hand Gesture Controlled Game for Hand Rehabilitation.” International Conference on Computer, Information Technology and Intelligent Computing (CITIC 2022). Atlantis Press, 2022.

[10] H.S. Shrisha, V. Anupama, ”NVS-GAN: Benefit of generative adversarial network on novel view synthesis”, International Journal of Intelligent Networks, Volume 5, 2024, 184-195,doi.org/10.1016/j.ijin.2024.04.002.

[11] Cahya, Rahmad., Arief, Prasetyo., Riza, Awwalul, Baqy. ”PowerPoint slideshow navigation control with hand gestures using Hidden Markov Model method.” 12 (2022).:7-18. doi: 10.31940/matrix.v12i1.7-18

[12] K., P., Kumari., Bandaram, Bharath, Goud., Kalvakuntla, Sumana., Bathula, Naresh., Bellamkonda, Harish. ”Automated Gesture Controlled Presentation Using Machine Learning.” International Journal For Science Technology And Engineering, 10 (2022).:1248-1251. doi: 10.22214/ijraset.2022.47517

[13] Savitha, M. ”Static Hand Gesture Recognition for PowerPoint Presentation Navigation using Thinning Method.” International Journal on Recent and Innovation Trends in Computing and Communication, 6 (2018).:187-189.