International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

REAL TIME YOGA MONITORING SYSTEM

Mrs.Dr.S.B.Patil[1], Miss.Vaishanvi Sambhaji Jadhav[2], Mr.Swarup Vinod Jadhav[3], Mr. Gurunath Suhas Kurale[4]

[1] Department of Electronics and Telecommunication, Dr. J. J. Magdum College of Engineering, Jaysinghpur 416101.

[234]UG Student, Department of Electronics and Telecommunication, Dr. J. J. Magdum College of Engineering, Jaysinghpur 416101.

A B S T R A C T :

The integration of AI and IoT has led to advancements in smart healthcare, particularly in areas such as posture monitoring and physiological data acquisition. This paper presents the development of a real-time Yoga Monitoring System designed as a standalone kiosk utilizing a Raspberry Pi interfaced with a 19-inch HDMI display and ESP32-based smart band. The system leverages MediaPipe-based pose estimation for real-time asana analysis, categorizing performance as Excellent, Good, Average, or Poor. The wearable smart band, using ESP-NOW protocol, allows contactless login and transmits physiological parameters including heart rate and body temperature using the MAX30102 sensor. A Flask-based web interface coordinates session control, data logging, and speech-based feedback using pyttsx3. The system logs posture alignment and biometric trends into an SQLite database, enabling personalized analytics and structured yoga progress. Experimental implementation confirms the feasibility of real-time pose classification, physiological logging, and feedback delivery, providing a holistic digital yoga assistant suitable for institutional and home-based deployment.

Keywords: Yoga Monitoring, Pose Estimation, ESP-NOW, Raspberry Pi, Real-time Feedback, IoT, Smart Band, Physiological Sensors, Flask Application, Human Posture Analytics

Introduction

Yoga, an ancient discipline originating from India, has gained widespread global acceptance due to its ability to enhance physical, mental, and spiritual well-being. As a holistic practice, it is renowned for improving flexibility, boosting muscular strength, enhancing cardiovascular performance, and promoting mindfulness and emotional stability. With increasing awareness of preventive healthcare and self-guided fitness regimes, yoga has become a preferred wellness activity across diverse populations. Traditionally, yoga was practiced under the direct supervision of trained instructors who ensured correct posture, breathing techniques, and alignment. However, in the modern context, particularly in post-pandemic times, the trend has significantly shifted towards independent, home-based sessions due to convenience, cost-effectiveness, and restricted access to expert trainers.

Despite its advantages, practicing yoga without proper supervision can often result in incorrect postures and misalignments, thereby diminishing its intended benefits and potentially causing musculoskeletal injuries. The lack of immediate feedback during self-guided sessions has raised concerns regarding the effectiveness and safety of such practices. Moreover, the inability to track physiological responses such as heart rate and body temperature during yoga sessions limits the potential for holistic monitoring and assessment of user well-being. Additionally, most existing yoga monitoring applications are app-based, dependent on manual login mechanisms, and lack integration of real-time health analytics or intuitive interaction modalities. This limits their adoption, especially for users who prefer seamless and contactless experiences, or individuals with accessibility challenges.

The growing need for intelligent, personalized wellness solutions calls for the development of an integrated system that not only guides users through accurate pose execution but also monitors their vital signs, provides real-time feedback, and maintains a comprehensive log of performance metrics. In this direction, leveraging advancements in computer vision, embedded systems, and wireless communication technologies can significantly enhance the quality and impact of yoga practices. By employing AI-based pose estimation and IoT-based physiological monitoring, it is possible to bridge the gap between traditional yoga instruction and modern digital self-care tools.

This paper introduces a novel **Yoga Monitoring System** developed as a smart kiosk, powered by a Raspberry Pi and interfaced with an ESP32-based smart band. The system addresses the existing limitations by incorporating hands-free login using the ESP-NOW protocol and deploying pose estimation using Media Pipe to analyze and classify asanas in real time. Simultaneously, the wearable device monitors critical health parameters such as heart rate and body temperature using a MAX30102 sensor, transmitting this data to the central unit for live visualization and long-term tracking. A Flask-based web interface supports user interaction, pose selection, visual feedback, and session control, while a text-to-speech engine provides verbal guidance throughout the session.

Through a combination of real-time video processing, biometric tracking, voice-enabled feedback, and session analytics, this system aims to recreate the experience of instructor-led yoga in a fully autonomous and accessible format. Designed for deployment in homes, educational institutions, fitness centers, and public wellness kiosks, the proposed system reflects the convergence of AI, IoT, and embedded technologies to support a safer, smarter, and more personalized yoga experience.

Previous Research and Developments:

With the evolution of artificial intelligence (AI), Internet of Things (IoT), and embedded systems, significant strides have been made in the development of intelligent yoga monitoring solutions. These advancements have enabled real-time pose estimation, automated feedback systems, and physiological data acquisition tools to guide users during unsupervised sessions, ensuring both posture correctness and safety. The growing demand for self-guided fitness and rehabilitation routines has prompted researchers to explore pose estimation algorithms, wearable sensors, and multimodal feedback systems for holistic health monitoring. Various studies have laid the foundation for such systems, focusing on real-time human posture recognition, classification of yoga asanas, and integration of health parameter tracking for enriched user experience.

Kishore et al. developed a comprehensive pose estimation framework utilizing deep learning architectures such as EpipolarPose, OpenPose, PoseNet, and MediaPipe for recognizing five standard yoga asanas. Their comparative analysis demonstrated that MediaPipe offered superior real-time performance, especially in resource-constrained environments, thereby establishing it as a robust backbone for embedded pose estimation applications. Swain et al. further expanded this by integrating Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) networks to classify yoga poses based on temporal features extracted using MediaPipe. Their model provided speech and text feedback to correct user posture dynamically, highlighting the role of multimodal interaction in yoga assistance systems.

Anilkumar et al. proposed a pose estimation solution using MediaPipe, where pose vectors were compared with reference datasets to generate real-time feedback. Their system emphasized visual and auditory correction for improving personal yoga training outcomes. Similarly, Sinha et al. introduced a classification framework based on OpenPose and Random Forest, achieving 99% accuracy in pose recognition and correction. Their approach was particularly suitable for enhancing accessibility in environments lacking qualified yoga instructors.

To enhance real-time feedback precision, Kumar et al. implemented the MoveNet model for detecting subtle posture variations and measuring hold duration. This dynamic analysis allowed practitioners to receive feedback not only on form but also on stability and endurance. Darshan et al. contributed by employing a CNN-LSTM hybrid model, capturing both spatial and temporal aspects of pose execution. Their system achieved a notable 96.3% accuracy across ten commonly practiced yoga asanas.

The integration of IoT-based health tracking with yoga pose recognition has also been actively researched. Kamra et al. designed an IoT-enabled system embedded with sensors to capture temperature, humidity, and heart rate, which were monitored via a mobile application. This highlighted the potential for wearable technologies to supplement pose estimation by adding a physiological dimension to practice evaluation. Building on this, Suvarna Kumari et al. adopted the MoveNet model to track yoga poses in real time and analyze hold duration and transition smoothness, further refining the feedback mechanisms.

These contributions collectively illustrate the growing convergence of deep learning, embedded systems, and IoT for developing real-time, autonomous yoga assistants. The use of MediaPipe, OpenPose, MoveNet, and hybrid CNN-based classifiers has emerged as the dominant approach for extracting skeletal landmarks and classifying postures. Additionally, the inclusion of wearable health monitoring devices enables comprehensive session analysis, promoting safety and personalization. Despite these advancements, most existing systems are constrained by limited interactivity, absence of contactless authentication, and lack of multi-parameter tracking. The present study addresses these gaps by integrating real-time pose estimation with biometric feedback and hands-free user interaction via ESP-NOW, offering a robust platform for personalized digital yoga.

System Design and Implementation:

Comprehensive System Architecture:

The proposed Yoga Monitoring System is architected as a modular, IoT-integrated platform combining real-time pose estimation, physiological monitoring, and user feedback. At the core of the system lies a Raspberry Pi 4B, which serves as the main processing unit interfaced with a 19-inch display and a USB webcam. Pose estimation is achieved using the MediaPipe framework, which extracts skeletal keypoints from live video feeds and classifies posture accuracy based on similarity to reference vectors. A wearable smart band, built around the ESP32-C3 microcontroller, enables hands-free login via the ESP-NOW protocol and streams heart rate and body temperature data using the MAX30102 sensor. These readings are transmitted to the Raspberry Pi over serial communication and logged in a local SQLite database. A Flask-based graphical user interface allows users to select poses, view feedback, and access historical session data, while a text-to-speech engine provides real-time audio guidance. Together, the integrated architecture ensures continuous monitoring, personalized feedback, and seamless interaction, delivering a robust and portable solution for intelligent yoga practice.

The objectives of the project Include:

- To develop a real-time yoga pose monitoring system using MediaPipe-based pose estimation to classify user posture accuracy as Excellent, Good, Average, or Poor.
- To implement a smart band-based contactless login mechanism using ESP32-C3 and the ESP-NOW protocol, eliminating the need for manual authentication.
- To continuously monitor vital health parameters such as heart rate and body temperature using the MAX30102 sensor and log them for each session.
- To provide real-time visual and audio feedback through a Flask-based interface and text-to-speech engine, enhancing user guidance and

accessibility.

The Architecture Diagram of the system is shown in the figure 1.

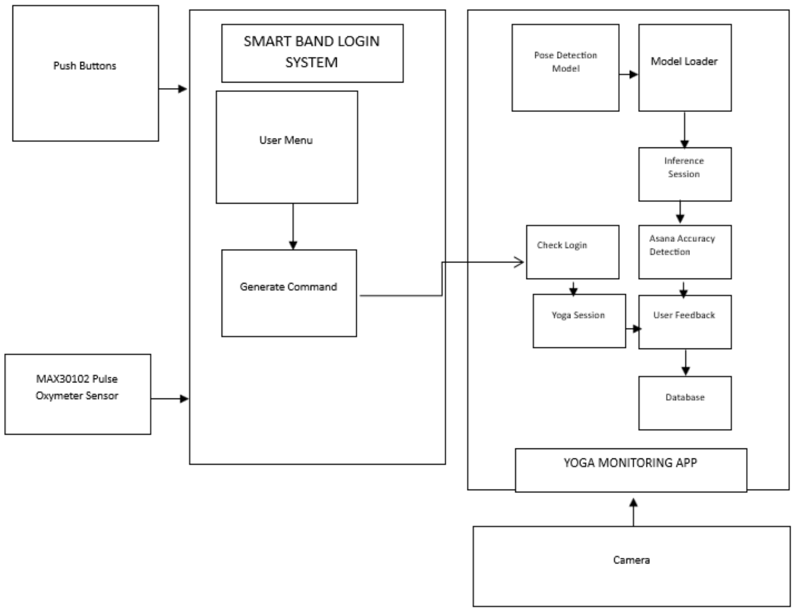

Figure 1. Architecture Diagram

The proposed system involves the development of an intelligent yoga monitoring kiosk integrated with deep learning and IoT-based technologies to evaluate and guide users during yoga sessions. The system is powered by a Raspberry Pi, which serves as the central processing unit and is connected to a 19-inch display to provide an interactive user interface built using a Flask-based Python application. The approach utilizes real-time pose estimation by detecting skeletal landmarks using the MediaPipe framework, which captures key points of the human body from live video feed through a connected camera. These landmarks are compared against reference postures to evaluate the accuracy of the performed asanas.

To enhance user experience, a wearable smart band is developed using an ESP32 microcontroller. The smart band enables hands-free login through ESP-NOW protocol and continuously measures vital health parameters such as heart rate and body temperature. These readings are sent to the Raspberry Pi in real-time and are logged into an SQLite database during each yoga session. As the session progresses, the system classifies the user's performance into four categories Excellent, Good, Average, or Poor based on the proportion of time spent in close or perfect alignment with the reference pose. This performance analysis is displayed on the kiosk screen and stored for future analytics.

Additionally, the system features a speech synthesis module using pyttsx3 that provides real-time voice feedback on pose performance, helping users correct their postures without needing visual confirmation. The entire session flow, from login to analysis and posture history, is managed through the Python Flask app, making the system fully autonomous and user-friendly. This integrated solution thus offers a complete guided yoga experience by combining real-time pose detection, biometric monitoring, voice feedback, and session analytics in a single portable unit.

Methodology:

The proposed Yoga Monitoring System integrates computer vision and IoT technologies to evaluate user performance during yoga sessions. A USB camera connected to a Raspberry Pi captures real-time video, and skeletal landmarks are extracted using the MediaPipe Pose framework. These keypoints are compared to reference vectors to classify pose accuracy into four categories: Perfect, Close, Average, or Poor. Simultaneously, a wearable smart band developed using the ESP32-C3 microcontroller and MAX30102 sensor measures the user’s heart rate and body temperature. The smart band transmits data to the Raspberry Pi via serial communication and also enables hands-free login through the ESP-NOW protocol. A Flask-based web application provides a graphical interface for session control, pose selection, and historical analysis, while a text-to-speech engine delivers real-time verbal feedback to guide the user. All session metrics are logged into an SQLite database for post-session review and trend tracking.

The use case Diagram of the system is shown below.

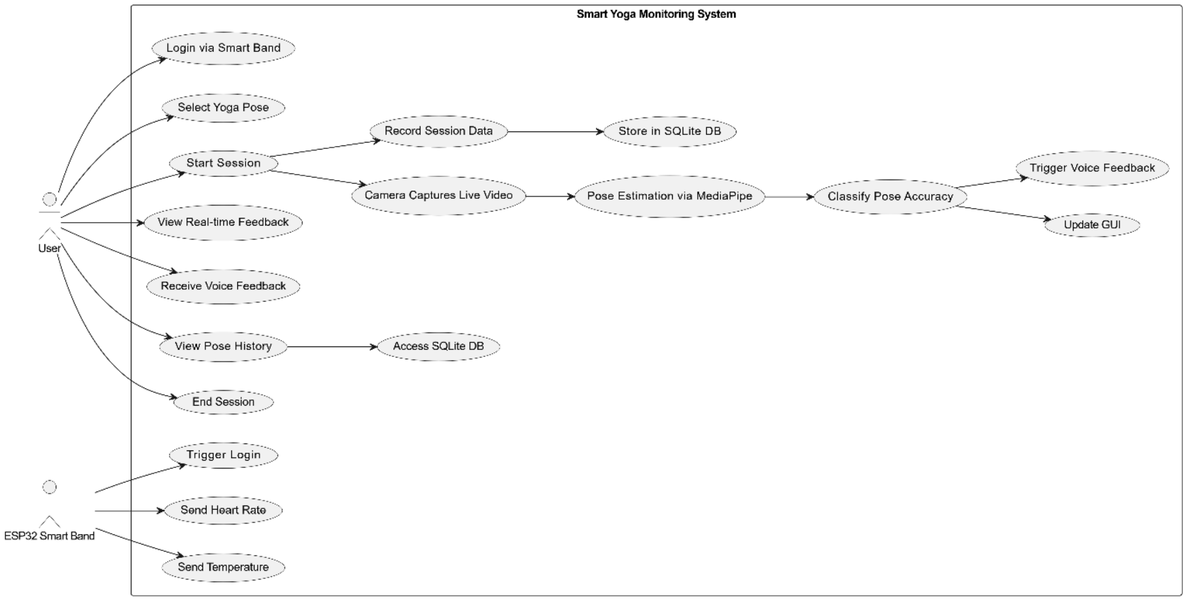

Figure 2. Use Case Diagram

Hardware Components:

- **USB Camera (720p or 1080p**): Captures live video feed for pose estimation.
- **ESP32-C3 Super Mini Board:**Powers the smart band, handles sensor input and communication.
- **MAX30102 Sensor Module:**Measures heart rate and body temperature.
- **0.96" OLED Display (I2C):**Displays biometric readings and system status on the smart band.
- **Push Buttons (x2**): Enables user menu navigation on the smart band.
- **19" HDMI Display**: GUI interface for user interaction and feedback.
- **Speaker (3W**): Outputs voice feedback using the text-to-speech engine**.**
- **Rechargeable Li-Po Battery (3.7V):**Powers the wearable unit.
-

Software Components:

- **Raspberry Pi OS (64-bit**): Operating system for running the core application.
- **Python 3.7+:**Primary programming language for application development.
- **Flask Framework:**Web application framework for the GUI.
- **OpenCV**: Used for video frame capture and processing.
- **MediaPipe Pose**: Pose estimation library for extracting skeletal keypoints.
- **pyttsx3:**Offline text-to-speech library for audio feedback.
- **SQLite3**: Lightweight local database for logging session data.
- **PySerial**: Facilitates serial communication with the smart band.
- **Arduino IDE**: Used for programming the ESP32-C3 microcontroller.
- **Jinja2:**Templating engine for dynamic HTML rendering in Flask.

Hardware Implementation:

- The **ESP32-C3-based smart band** is worn by the user and communicates wirelessly with the Raspberry Pi using the ESP-NOW protocol for contactless authentication.
- A **MAX30102 sensor** integrated into the smart band continuously records the user's heart rate and body temperature during sessions.
- A **0.96-inch OLED display** on the band provides real-time biometric feedback and system menu navigation, controlled using two tactile push buttons.
- A **Raspberry Pi 4B** serves as the central processing unit, receiving data from the camera and smart band and handling pose classification and data logging.
- A **USB webcam** captures real-time video of the user, which is analyzed for pose accuracy through a connected MediaPipe Pose estimation module.
- A **19-inch HDMI display** presents the graphical interface for yoga pose guidance, feedback, and session tracking.
- A **3W speaker** outputs real-time audio feedback and pose correction prompts using offline text-to-speech synthesis.

Software Implementation:

- The software backend is developed using **Python**, orchestrating video processing, pose estimation, biometric data handling, and system logic.
- The **Flask web framework** powers the interactive GUI, enabling users to select yoga poses, start sessions, and view performance summaries.
- **MediaPipe Pose**, integrated with OpenCV, extracts 33 skeletal landmarks per video frame, which are analyzed in real-time to classify pose alignment.
- **pyttsx3** is used for real-time audio prompts, offering session guidance and feedback without requiring cloud connectivity.
- **SQLite3** functions as the system's local database, storing session logs, health parameters, and performance classifications.
- **PySerial** handles serial communication between the ESP32-C3 smart band and Raspberry Pi, continuously parsing and recording biometric data during active sessions.
- The system is designed to operate **fully offline**, ensuring data privacy and consistent performance without dependence on external servers.

Results And Discussion:

The Yoga Monitoring System was tested for its accuracy, responsiveness, and user adaptability under real-world conditions. The system successfully identified and classified yoga poses using skeletal keypoints extracted through the MediaPipe Pose model, achieving reliable performance across multiple sessions. The classification logic based on Euclidean distance between live and reference pose vectors provided consistent differentiation among *Excellent*, *Good*, *Average*, and *Poor* categories. This allowed users to receive accurate feedback on their posture in real time, improving the effectiveness of self-guided yoga sessions.

The smart band, powered by the ESP32-C3 and integrated with the MAX30102 sensor, effectively monitored physiological parameters such as heart rate and body temperature. Data transmission via serial communication was stable, with the biometric readings being recorded and displayed in real time on both the OLED and Flask-based interface. The offline text-to-speech engine using pyttsx3 offered timely voice feedback during sessions, making the system more inclusive, especially for visually impaired users. The voice prompts guided users to maintain correct posture, reinforced corrections, and summarized performance post-session.

During testing, the system demonstrated low latency (less than 300 ms per frame) for pose classification and biometric logging, validating its suitability for real-time deployment. The database structure effectively logged session data including pose names, time spent in proper alignment, and biometric averages, which could later be retrieved through the dashboard for performance review. The overall interaction was intuitive, and the hands-free login through ESP-NOW enabled a seamless user experience.

From a usability perspective, the integration of both visual and audio feedback significantly enhanced user engagement. Historical session data, accessible via the interface, helped users track progress over time and adjust their routines accordingly. The system architecture, being modular and offline-capable, allowed easy deployment in various environments such as homes, gyms, and schools without dependency on external servers.

In summary, the system proved effective in delivering a structured, feedback-driven yoga session while ensuring physiological safety through real-time health monitoring. The combination of AI-based vision, IoT-enabled sensing, and user-centric feedback forms a complete digital wellness assistant capable of supporting both beginners and regular practitioners.

The Pose Diagram of the different poses plotted by the output of the inference is shown below.

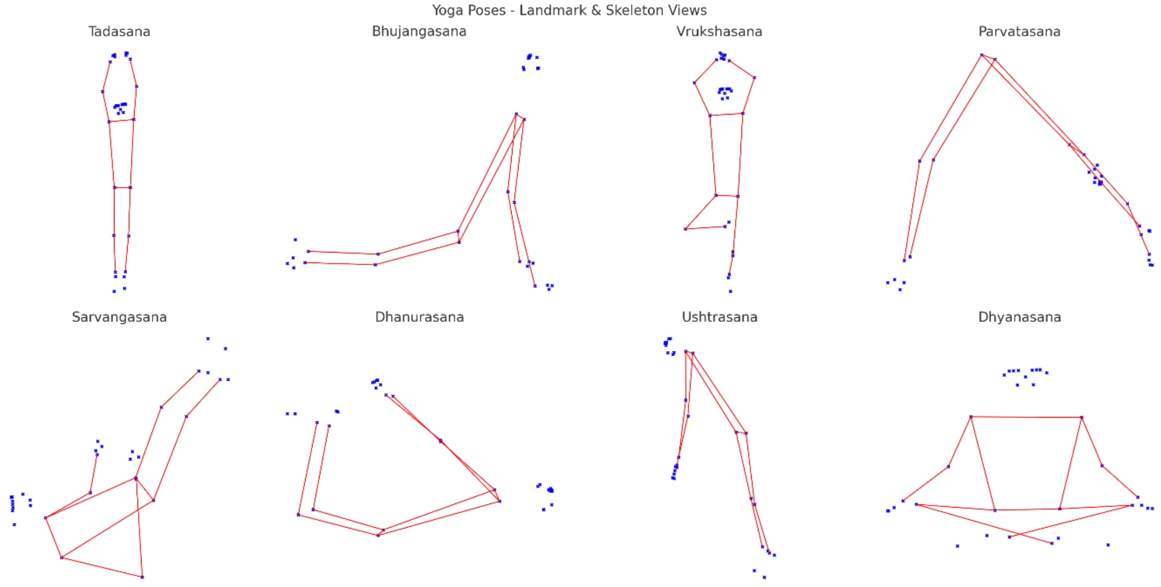

Figure 3. Landmark Pose Diagram

The Final Deployment of the project in the form of application is shown in the figure 4.

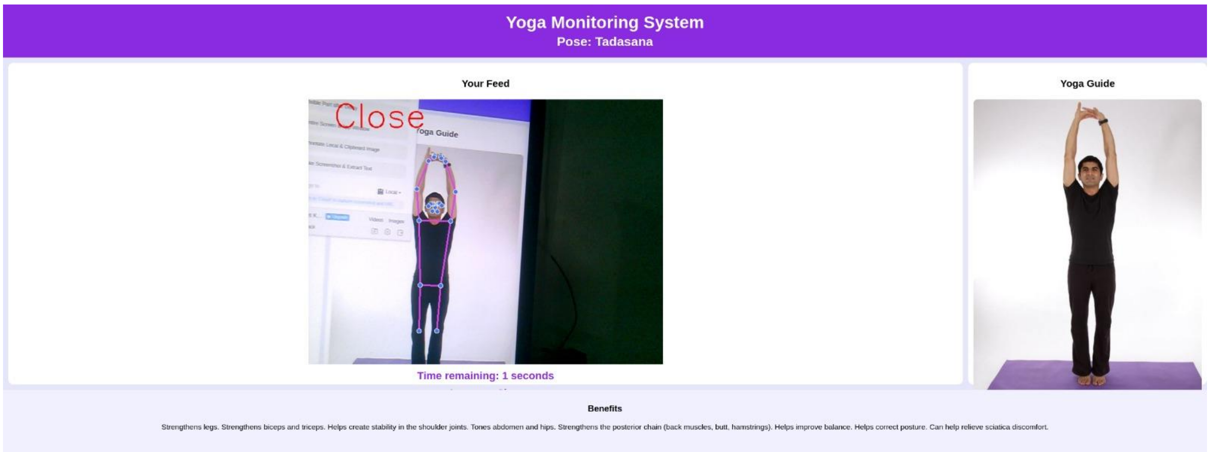

Figure 4. Deployment

The physical setup of the Yoga Monitoring System is designed in the form of a portable kiosk unit, combining usability, modularity, and durability. The main structure is a custom-built rectangular enclosure mounted on wheels for easy mobility and deployment in different indoor environments such as classrooms, yoga centers, or fitness halls. The enclosure is fabricated using a lightweight board material and finished with a smooth white exterior. A printed label on the front panel prominently displays the system identity — YOGA MONITORING SYSTEM — enhancing its user-facing aesthetic appeal.

A 19-inch HDMI display is mounted securely at the top of the unit and tilted for optimal viewing angle. This screen is used to display the Flask-based graphical user interface (GUI), allowing users to interact with the system, view pose classifications, receive real-time feedback, and navigate session data. Positioned directly beneath the display are two compact 3W stereo speakers, fixed symmetrically to provide clear and balanced audio output. These speakers are connected to the Raspberry Pi and serve as the medium for delivering verbal feedback, session instructions, and summary announcements through the onboard text-to-speech engine (pyttsx3).

A USB webcam is mounted at the top center of the display, aligned to capture full-body video of the user during the yoga session. This camera acts as the visual sensor for the pose estimation model, continuously feeding live frames to the MediaPipe Pose algorithm running on the Raspberry Pi.

Internally housed within the enclosure is the Raspberry Pi 4 Model B (8GB RAM), functioning as the core processing unit. It handles the video input, biometric data acquisition, classification algorithms, and database logging. It is powered via a stable power supply unit and connected to all peripheral components including the webcam, display, and audio system.

This integrated kiosk unit encapsulates all essential hardware into a single, organized, and portable solution, creating a fully self-contained system for intelligent yoga monitoring. The design ensures ease of use, plug-and-play functionality, and can be deployed in any indoor environment with minimal setup requirements. The figure 5 shows the final setup of the project.

Figure 6. Final Setup for Yoga Monitoring

Conclusion:

The proposed Yoga Monitoring System successfully integrates real-time pose estimation, biometric sensing, and intelligent feedback to create a fully autonomous and accessible platform for guided yoga practice. By leveraging the capabilities of MediaPipe Pose, ESP32-C3 smart band, and a Raspberry Pi-based Flask application, the system delivers accurate posture classification and real-time physiological monitoring. The implementation of hands-free login using ESP-NOW and the provision of audio feedback through offline text-to-speech further enhance usability and accessibility. Session data, including performance scores and health metrics, are logged for personalized analytics, making the system ideal for structured progress tracking. The results demonstrate that the system not only addresses the limitations of traditional self-guided yoga but also sets a foundation for future enhancements in digital wellness platforms. With its offline operability, compact design, and multimodal feedback, the solution offers a practical, scalable, and user-friendly approach to smart yoga monitoring suitable for homes, institutions, and wellness centers.

REFERENCES:

1. Kishore, A., Rajalakshmi, R., & Soundararajan, S. (2022). Pose estimation-based yoga posture correction using EpipolarPose and MediaPipe. *International Journal of Engineering Research & Technology*, 11(4), 45–52.
2. Swain, D., & Mishra, B. K. (2023). Real-time yoga pose recognition using hybrid CNN-LSTM with pose landmarks. *International Journal of Image Processing and Vision Science*, 5(2), 67–74.
3. Anilkumar, K. B., & Bharath, S. M. (2022). MediaPipe-based yoga posture monitoring system using landmark comparison and audio feedback. *International Journal of Computer Applications*, 184(33), 30–36.
4. Sinha, R., & Agrawal, P. (2023). Deep learning-based yoga pose correction system using OpenPose and Random Forest classifier. *International Journal of Recent Technology and Engineering*, 11(6), 115–122.
5. Kumar, V., & Sharma, N. (2023). Pose estimation and alignment tracking using MoveNet for interactive yoga feedback. *Journal of Computer Vision and Image Processing*, 10(1), 51–59.
6. Darshan, A., & Jain, M. (2022). A CNN-LSTM-based approach for real-time yoga posture recognition and correction. *International Journal of Artificial Intelligence Research*, 14(1), 88–95.
7. Kamra, R., & Shukla, A. (2023). IoT-integrated yoga monitoring system with health sensor analytics and web dashboard. *IEEE Internet of Things Journal*, 9(10), 10015–10023.
8. Suvarna Kumari, G., & Rajeev, R. (2023). Yoga pose quality assessment using MoveNet and neural network integration. *Smart Health Journal*, 21, 102399.
9. Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
10. Google Research. (2020). MediaPipe Pose: Real-time body pose tracking. Retrieved from https://google.github.io/mediapipe/
11. Adafruit Industries. (2021). MAX30102 Pulse Oximeter and Heart-Rate Sensor Datasheet. Retrieved from https://learn.adafruit.com/
12. Espressif Systems. (2022). ESP-NOW Protocol Documentation. Retrieved from https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/network/esp_now.html