Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 03 , March - 2026

Real Time Patient Activity Monitoring using Vision AI and Pose estimation

P. Sridevi
Assistant Professor, Information Technology, GVPCEW Visakhapatnam, Andhra Pradesh, India

B. Jyotshna
Information technology, GVPCEW Visakhapatnam, Andhra Pradesh, India

P. Sravanthi
Information technology, GVPCEW Visakhapatnam, Andhra Pradesh, India

B. Prathibha
Information technology, GVPCEW Visakhapatnam, Andhra Pradesh, India

K. Bhavana
Information technology,GVPCEW Visakhapatnam, Andhra Pradesh, India

Abstract - Patient monitoring in hospitals, elderly care centres, and in home healthcare environments is a challenging and strenuous task because of the limited number of medical staff, delayed emergency response, and the lack of proper documentation of the patient movements. Monitoring patients manually is difficult due to limitations of the care giver and can be periodic but not continuous. They may fail to detect critical events such as falls, seizures, or prolonged immobility before it actually happens and may result in delayed response. Keeping this problem in mind, this study proposes a Patient Activity Monitoring System using Robotics Technology with Ambient Intelligence. The integration of Robotics Technology enables continuous, automated, and contactless monitoring with minimal human intervention, thereby improving response time and patient safety. The system uses computer vision to monitor patient activities through live video feeds. It utilizes the Media Pipe library for real-time pose estimation and hand landmark detection, which efficiently extracts accurate body key points from video frames with low computational cost. The system identifies patient activities such as sitting, standing, walking, laying down, falls, and help gestures, and generates instant alerts to caregivers during abnormal situations. A Retrieval-Augmented Generation (RAG)- based Chatbot is incorporated to respond to user prompts. RAG technology combines database retrieval with AI- based text generation to provide accurate, context-aware, and data based on stored patient activity records.

Index Terms: Pose Estimation, Media Pipe Framework, Retrieval-Augmented Generation (RAG), Ambient AI, Rule-Based Detection System.

INTRODUCTION

Human Activity Recognition (HAR) is an important area of research in the field of smart healthcare systems and intelligent monitoring systems. In hospitals, rehabilitation centres, and homecare services, constant monitoring of patients is very essential to recognize abnormal events such as falls, seizures, inactivity, and other signs of distress.

Existing monitoring techniques include manual monitoring and wearable device-based monitoring. Although wearable device-based monitoring can provide valuable bodily function metrics about a patient's health, it causes discomfort for elderly patients.
Moreover, wearable device-based monitoring can produce poor results if the wearable malfunctions
(sensor shifts) or due to the patient's skin sensitivity. Recent advancements in computer vision and artificial intelligence have led to a new class of monitoring systems using dynamic cameras.

The study presented in this paper aims to develop an AI-based Patient Activity Monitoring System using Robotic technology that can capture patient activities with ESP32 camera dynamically to automatically detect, classify, and analyze patient activities using vision-based pose estimation techniques. The system uses ESP32 camera which acts as vision sensor, is placed on a dynamic Robot to capture live video streams and detect **human skeletal landmarks** to represent the posture of a patient in a structured

manner. Patient activities such as sitting, standing, walking, lying down, falls, and seizure-like activities can be detected using these landmarks. Apart from detecting patient activities on a live stream, the system can also generate AI-based clinical summaries and an interactive Chatbot interface for caregivers.

Deep learning models like CNN has proven to be efficient for detecting human activities as mentioned above. This paper has been divided into four main stages.

Stage 1: The robot captures video frames taken by ESP32 camera and processed for the extraction of the 3D skeletal landmarks of the human body.

Stage2: The motion-based and posture-based features are extracted from the video frames, and the activities are identified using the rule-based recognition mechanism.

Stage3: The activity logs are processed for the generation of structured clinical summaries using an Artificial General Intelligence model.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 03 , March - 2026

Stage 4: The Retrieval-Augmented Generation (RAG) mechanism is employed for the retrieval of the monitoring data basing on the user prompt.

LITERATURE SURVEY

Patient Activity Monitoring (PAM) has become an important research area in healthcare monitoring systems. It helps in tracking patient movements and detecting abnormal or critical activities using wearable sensors and machine learning models. Many researchers have proposed different techniques using machine learning and deep learning to improve the accuracy and reliability of patient activity monitoring systems.

RAG framework

The RAG framework improves AI performance by combining information retrieval and text generation. It first retrieves relevant information from a knowledge base and then uses an AI model to generate accurate responses based on that data. This approach helps provide more reliable and context-based outputs in the patient activity monitoring system.

Nishanth Adithya Chandramouli et al. [1] proposed a hybrid deep learning architecture that combines **Convolutional Neural Networks (CNN)** and **Bi- Directional Long Short-Term Memory (BiLSTM)** for human activity recognition using wearable sensor data. The CNN component extracts spatial features from the sensor signals, while the BiLSTM captures

temporal dependencies in activity sequences. The model was evaluated on the **MHEALTH and Actitracker datasets**.

F. J. Ordóñez **et al. [2]** presented a deep learning-based human activity recognition system using **Convolutional Neural Networks (CNN)** to analyze data from sensors such as accelerometers and gyroscopes. The system automatically extracts important features from raw sensor signals to classify activities like walking, sitting, and running. Experimental results showed that CNN-based models provide high accuracy and efficiency for healthcare monitoring applications.

H. Wang **et al. [3]** proposed a **CNN-LSTM hybrid model** for activity recognition. In this approach, CNN is used for feature extraction from sensor data, while LSTM captures sequential patterns in human movement. The model demonstrated better recognition performance compared to standalone CNN or LSTM models because it effectively learns both spatial and temporal information.

Luigi Bibbò et al [4] studied different Artificial Intelligence approaches for Human Activity

Recognition (HAR) in healthcare systems. The research explains how wearable sensors, IoT devices, and machine learning techniques can monitor patient activities and detect abnormal movements. The study highlights the importance of HAR systems for elderly monitoring and healthcare application.

METHODOLOGY

The proposed study monitors the activities of the patient using hardware components such as ESP32- CAM, a camera module, and a PC for processing. The ESP32-CAM module is connected to a power supply and Wi-Fi network, and it captures real-time video of the patient and sends it to the PC for processing and activity analysis. Using the Media Pipe Pose Detection, the system is able to identify the body landmarks of the patient, thereby monitoring the activities of the patient. Depending on the detected movement, the system is able to identify the activities of the patient, such as sitting, standing, walking, or falling. The system is continuously monitoring the activities of the patient, where the information is being stored. In the event that abnormal situations, such as falling, seizure, or emergency, are detected, the system sends notifications to the caregiver. Using the web dashboard, the information is being processed, where the caregiver is able to monitor the patient's condition.

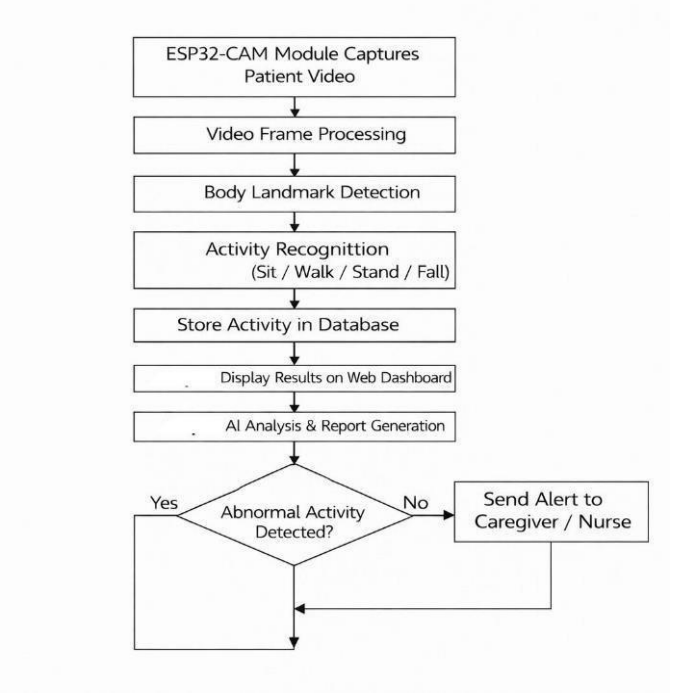


Fig1: Data Flow of system

The above flow chart is explained in the following modules.

1. Video Capture Module

In this module, the ESP32-CAM captures the video of the patient's environment in real time. The camera continuously records the movements and posture of the patient for monitoring purposes. This hardware module is fixed at a particular point or on a robot to view the
patient's posture properly. The captured video is sent to the monitoring system through a network connection. The continuous capture of video helps to monitor the patient without any interruptions. This module act as the primary source of input for the entire system.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 03 , March - 2026

2. Frame Processing Module

In this module, the video captured from the ESP32- CAM is converted into individual frames for analysis. The system uses the OpenCV (cv2) library in Python to read the video stream and extract frames in real time. Each frame is resized to a fixed resolution to maintain consistency and reduce processing complexity. To remove noise and improve image quality, filtering techniques such as Gaussian Blur (`cv2.GaussianBlur()`) or Median Filtering (`cv2.medianBlur()`) are applied using OpenCV functions. These filters help smooth the image and reduce unwanted noise. After preprocessing, the cleaned frames are passed to the pose detection module for further analysis.

3. Pose Detection Module

In this module, pose estimation is used to detect the body posture of the patient from the processed video frames. The system uses the MediaPipe Pose model to identify important body landmarks such as shoulders, elbows, hips, knees, and ankles. These landmarks represent the structure of the human body and help in understanding the patient's movement. The model analyzes each frame and returns the coordinates of the detected body joints. By tracking these points over time, the system can observe how the patient moves or changes posture. This information is important for recognizing activities like sitting, standing, walking, or falling. The detected landmark data is then sent to the activity recognition module for further analysis.

4. Activity Recognition Module

In this module, the system recognizes patient activities using Python code based on the body landmark data obtained from the pose detection module. The coordinates of key body points detected by MediaPipe Pose are used as input for activity analysis. The Python program compares the relative positions of landmarks such as the height of hips, knees, and shoulders across frames. Using rule-based conditions (if–else logic), the system determines whether the patient is sitting, standing, walking, or falling. For example, if the hip and knee positions are close to the ground level, the activity may be classified as sitting, while sudden changes in body orientation may indicate a fall. By continuously analyzing these landmark coordinates frame by frame, the Python code classifies the patient's activity in real time. This helps the system monitor patient behavior effectively.

5. Activity Monitoring and Alert Module

In this module, the system continuously monitors the activities recognized by the activity recognition module. The detected activities are stored as real-time activity logs to keep track of patient behavior over time. The Python program checks these activities to identify abnormal situations such as falls, seizures, or sudden inactivity. When such an event is detected, the system triggers an alert using predefined conditions in the Python code. The alert can be sent through the web dashboard notification or messaging service connected to the system. This is implemented using Python libraries that send messages or update the dashboard instantly. By automatically generating alerts, the system ensures that caregivers or medical staff are informed quickly and can provide immediate assistance.

6. Dashboard and Display Module

In this module, the system displays the monitoring results through a web dashboard. The dashboard is created using Python with web frameworks such as Flask along with HTML, CSS, and JavaScript. The Python program sends the detected activity data and video stream to the web server. The server then updates the information on the dashboard page. The live video from the ESP32-CAM is displayed using a streaming URL, while the detected activities and alerts are shown as text logs. When a new activity is recognized, the Python code updates the dashboard automatically. This allows caregivers or medical staff to view the live monitoring results through a web browser in real time.

SYSTEM ARCHITECTURE

In the proposed system, the activities performed by the patient are monitored using the ESP32-CAM vision system. The ESP32-CAM captures the live video of the patient, and the frames are used to detect the activities performed by the patient. The activities performed by the patient, such as sitting, walking, standing, and falling, are detected using the motion features in the frames. The activities performed by the patient are checked to determine whether the activities are normal or abnormal. Based on the activities performed by the patient, if the patient falls or performs abnormal activities, the system sends a notification to the caregiver to provide immediate care.

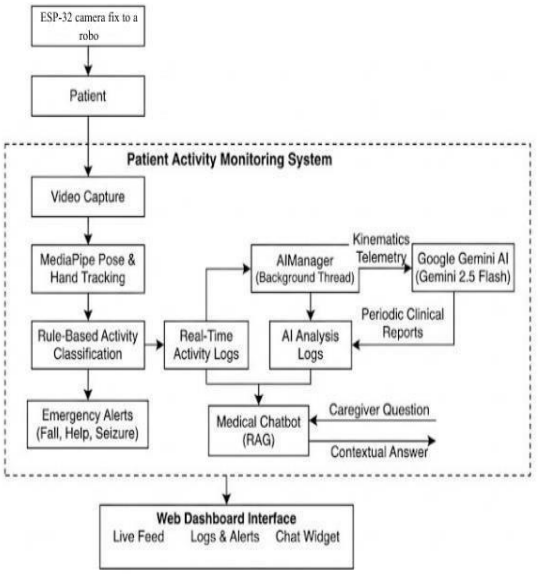


Fig2: System architecture

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 03 , March - 2026

IMPLEMENTATION DETAILS

In this work, we used ESP32-CAM, Python, Visual Studio Code (VS Code) and MediaPipe Pose Detection to implement a patient activity monitoring system.

A. Python

Python is used for developing the proposed patient activity monitoring system. It has the ability to support computer vision and AI applications through powerful libraries. Python processes the video captured from the ESP32-CAM and recognizes the activities. It is also useful for sending alerts for the system.
The important libraries imported are:

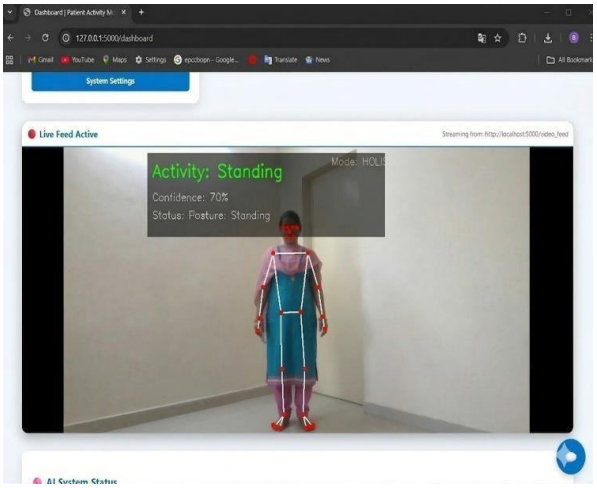
- **Flask** – Used to build the web application and manage user interaction.
- **OpenCV** – Used for capturing and processing video frames.
- **MediaPipe** – Used for body landmark detection and pose estimation.
- **SQLite** – Used for storing user details, activity logs, and system outputs.
- **Google Generative AI** – Used for AI-based analysis and generating intelligent response

B. Visual Studio Code (VS Code)

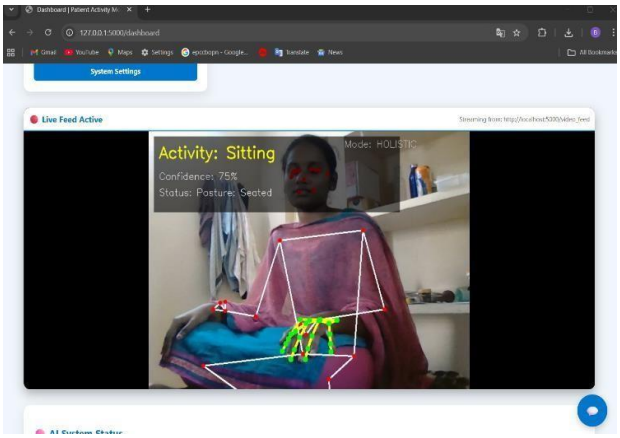
Visual Studio Code is used for developing the proposed system. It has many facilities for developing the proposed system. Visual Studio Code has the ability to support Python extensions for easy coding and execution. It is useful for developing and managing the project.

RESULTS

We successfully implemented the proposed Patient Activity Monitoring System using real-time video captured from the ESP32-CAM. The system detects activities such as sitting, standing, walking, lying down, and falling using MediaPipe pose landmarks and rule-based classification.After extracting the body landmarks, the system analyzes the position and movement of different body parts to understand the posture of the person. In the figure 3(a), when the body is in a vertical position it is identified as standing, continuous movement of legs indicates walking, and when the body is horizontal it is recognized as lying down or falling. Once the activity is identified, the system continuously updates the results in real time.
The detected activity is stored along with timestamps and displayed on a web dashboard.It represents the visualization of the system performance by plotting the detected activities and monitoring results during continuous operation.



(a) Patient standing position



(b) Patient Sitting position

Fig3: Activity Recognition based on Body Landmarks

CONCULSION

The study of patient activity monitoring system uses an ESP32-CAM module to observe and record patient movements in real time. The system captures images from the patient environment and processes them to understand activities. The system monitors the patient continuously without the need for constant human supervision. It can detect abnormal movement and send alerts to caregivers for quick response. This helps improve patient safety and makes monitoring easier for healthcare staff and family members. Overall, the project demonstrates how a simple hardware based monitoring system can support better patient care and continuous observation.

REFERENCES

[1] N. A. Chandramouli, A. K. Gupta, and S. S. Chandra, “Hybrid Deep Learning Model Using CNN and BiLSTM for Human Activity Recognition,” *IEEE Access*, vol. 9, pp. 120783–120794, 2021.
[2] F. J. Ordóñez and D. Roggen, “Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition,” *Sensors*, vol. 16, no. 1, pp. 1–25, 2016.
[3] H. Wang, L. Zhang, and T. Wang, “Deep Learning- Based Human

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 03 , March - 2026

Activity Recognition Using Hybrid CNN-LSTM Model," *IEEE Sensors Journal*, vol. 19, no. 11, pp. 4557–4565, 2019.

[4] Luigi Bibbò , and Marley M. B. R. Vellasco ," AI- based Human Activity Recognition (HAR) " Applied Sciences, vol. 13, no. 6, 2023.

[5] S. Ha and S. Choi, "Convolutional Neural Networks for Human Activity Recognition Using Multiple Accelerometer and Gyroscope Sensors,"

[6] N. A. Chandramouli, A. K. Gupta, and S. S. Chandra, "Hybrid Deep Learning Model Using CNN and BiLSTM for Human Activity Recognition," *IEEE Access*, vol. 9, pp. 120783–120794, 2021.

[7] F. J. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," *Sensors*, vol. 16, no. 1, pp. 1–25, 2016.

[8] H. Wang, L. Zhang, and T. Wang, "Deep Learning- Based Human Activity Recognition Using Hybrid CNN-LSTM Model," *IEEE Sensors Journal*, vol. 19, no. 11, pp. 4557–4565, 2019.

[9] Luigi Bibbò , and Marley M. B. R. Vellasco ," AI- based Human Activity Recognition (HAR) " Applied Sciences, vol. 13, no. 6, 2023.

[10] S. Ha and S. Choi, "Convolutional Neural Networks for Human Activity Recognition Using Multiple Accelerometer and Gyroscope Sensors," *IEEE International Joint Conference on Neural Networks (IJCNN)*, 2015.

[11] T. Haresamudram, D. V. Anderson, and T. Plötz, "Self-Supervised Learning for Human Activity Recognition Using Wearable Sensors," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 4, pp. 1–30, 2020.

[12] L. Bao and S. S. Intille, "Activity Recognition from User-Annotated Acceleration Data," *International Conference on Pervasive Computing*, pp. 1–17, 2004.

[13] A. Bulling, U. Blanke, and B. Schiele, "A Tutorial on Human Activity Recognition Using Body-Worn Inertial Sensors," *ACM Computing Surveys*, vol. 46, no. 3, pp. 1–33, 2014.

[14] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity Recognition Using Cell Phone Accelerometers," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 2, pp. 74–82, 2011.

[15] sale, O. Pujol, and P. Radeva, "Human Activity Recognition from ccelerometer Data Using a Wearable Device," *Pattern Recognition and Image Analysis*, vol. 21, no. 3, pp. 1–7, 2011.

[16] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2022