Human Pose Estimation using Machine Learning

A Project Report

submitted in partial fulfillment of the requirements

of

AICTE Internship on AI: Transformative Learning
with
TechSaksham – A joint CSR initiative of Microsoft & SAP

by

Atharva Karale ,

atharvakarale2604@gmail.com

Under the Guidance of

P.Raja, Master Trainer,Edunet Foundation

ACKNOWLEDGEMENT

We take this opportunity to express our deepest gratitude to all the individuals who have been instrumental in helping us throughout this thesis work.

Firstly, I would like to extend my sincere thanks to my supervisor, P.Raja sir. His mentorship has been invaluable and his guidance has been a true cornerstone of my academic journey. he has not only provided critical feedback and support for my project but also shown an unwavering belief in my capabilities. His encouragement and thoughtful critiques have been a source of innovative ideas and inspiration, which have significantly contributed to the successful completion of this work. The confidence he placed in me was truly motivating. Beyond the project itself, his support has been instrumental in my growth as a professional. The lessons and insights he has shared have extended far beyond technical skills; they have shaped me into a more thoughtful, responsible, and motivated individual.

Atharva Karale

ABSTRACT

Human pose estimation is a critical task in computer vision, with applications spanning healthcare, sports analysis, human-computer interaction, and security surveillance. The primary challenge lies in accurately detecting human poses in diverse environments, handling occlusions, variability in body movements, and maintaining real-time performance for live applications. This project addresses these challenges by developing a robust pose estimation system using deep learning techniques.

The system utilizes a pre-trained convolutional neural network (CNN) model integrated with OpenCV to detect and map key body points from images, videos, or live camera feeds. The project leverages state-of-the-art pose estimation frameworks, such as OpenPose, to ensure high accuracy and scalability for single or multi-person scenarios. The methodology includes preprocessing input data, extracting key body points, and visually representing it. The implementation uses Python, OpenCV, and Streamlit to deliver an interactive and user-friendly application. Results are validated through visual outputs and quantitative performance metrics, including detection accuracy and processing time.

In conclusion, the proposed pose estimation system successfully addresses the challenges of accuracy, scalability, and real-time performance. It provides a foundation for applications in diverse fields such as fitness monitoring, physiotherapy, and gesture-based controls. Future work could involve extending the system for activity recognition, improving multi-person detection, and enhancing computational efficiency for large-scale deployment.

TABLE OF CONTENT

Abstract I

Chapter 1. Introduction 1

1.1 Problem Statement 1

1.2 Motivation 1

1.3 Objectives.. 2

1.4. Scope of the Project.. 2

Chapter 2. Literature Survey 3

Chapter 3. Proposed Methodology ....

Chapter 4. Implementation and Results ....

Chapter 5. Discussion and Conclusion ....

References..

LIST OF FIGURES

Figure No.	Figure Caption	Page No.
Figure 1	System workflow for Pose Estimation System	**07**
Figure 2	Web page of Pose Estimation Application	**09**
Figure 3	Result Image	**09**
Figure 4	Result Video	**10**
Figure 5	Result of Live webcam	**11**
Figure 6		
Figure 7		
Figure 8		
Figure 9		

CHAPTER 1

Introduction

1.1 Problem Statement:

The problem being addressed in this project is the accurate and efficient estimation of human poses from images, videos, or live camera feeds. Human pose estimation involves identifying key body points (such as joints and limbs) and mapping their relative positions to understand body posture and movements. This task is critical in various applications, including healthcare, sports analytics, human-computer interaction, animation, and security surveillance.

Why is this significant ?

This problem is significant because solving it enables transformative applications in domains like:

- **Healthcare**: Monitoring patient posture and movement during rehabilitation.
- **Sports Analytics**: Improving athlete performance by analyzing their movements.
- **Surveillance and Safety**: Identifying suspicious activities or monitoring crowd behavior.
- **Human-Computer Interaction**: Developing gesture-based control systems for enhanced usability.

Addressing these challenges contributes to making pose estimation systems more robust, scalable, and suitable for real-world applications.

1.2 Motivation:

This project was chosen to address the growing need for accurate and real-time human pose estimation, which has become essential in various domains where understanding human movements is critical. The motivation stems from the rapid advancements in artificial intelligence and computer vision, making it possible to develop systems that can analyze and interpret human poses with high precision.

Potential Applications and Impact:

- Healthcare and Rehabilitation
- Sports and Fitness
- Security and Surveillance
- Animation and Entertainment
- Human-Computer Interaction

1.3 Objective:

The primary objective of this project is to design and implement a robust and efficient human pose estimation system that can accurately detect and map key body points from images, videos, or real-time camera feeds.

Specific objective

- **Develop a Pose Estimation Framework**: Utilize state-of-the-art deep learning models to detect keypoints of the human body (e.g., joints, limbs).Implement the system using tools like OpenCV and pre-trained models (e.g., OpenPose).

- **Ensure Real-Time Performance**: Optimize the system for real-time processing to support applications requiring live feedback, such as surveillance or fitness tracking.

- **Handle Diverse Scenarios**: Design the system to perform effectively across varying conditions, such as different lighting, backgrounds, and camera angles.Support single-person and multi-person pose estimation scenarios.

- **Create a User-Friendly Interface**: Develop an interactive and intuitive front end using tools like Streamlit to allow users to upload images, videos, or use a live camera for pose estimation.

Scope of the Project:

1. **Human Pose Detection**: The system focuses on detecting key body points (e.g., joints and limbs) and their connections to estimate human poses from static images, videos, or live camera feeds.
2. **Platform and Accessibility**: A user-friendly interface built using Streamlit for seamless interaction, allowing users to upload images/videos or utilize real-time camera feeds.
3. **Technologies Used**: OpenCV, TensorFlow (or similar frameworks), and pre-trained pose estimation models (e.g., OpenPose) to ensure accuracy and efficiency.
4. **Applications**: Healthcare and Rehabilitation, Sports and Fitness, Security and Surveillance, Animation and Entertainment, Human-Computer Interaction

Limitations:

1. **Complex Backgrounds**:Performance may degrade in highly cluttered or dynamic backgrounds where human poses are occluded or hard to distinguish.
2. **Real-Time Constraints**:While optimized for real-time processing, performance may vary depending on the hardware, especially for multi-person pose estimation or high-resolution inputs.
3. **Environmental Factors**:Varying lighting conditions, camera angles, and resolutions can impact the accuracy of keypoint detection.

CHAPTER 2

Literature Survey

2.1 Review relevant literature or previous work in this domain.

Human pose estimation has been a significant area of research in computer vision due to its wide range of applications. Numerous methods and algorithms have been proposed over the years, with advancements driven by improvements in deep learning and computational power. Below is a review of relevant literature and previous work in this domain:

Traditional Approaches

1. **Early Template-Based Methods**:
   - Initial pose estimation methods relied on matching templates or silhouettes of the human body against the input image. These methods lacked flexibility and failed in scenarios involving dynamic poses or occlusions.
   - Example: Techniques based on Histogram of Oriented Gradients (HOG) for body detection.
2. **Pictorial Structures**:
   - Models like **Pictorial Structures** represented the human body as a collection of parts connected via deformable springs. These methods used probabilistic models to infer poses but struggled with complex backgrounds or overlapping individuals.

2.2 Mention any existing models, techniques, or methodologies related to the problem.

Human pose estimation has seen significant advancements, with numerous models and techniques developed to address its challenges.

1. **OpenPose**: which employs a bottom-up approach by first detecting key body points and then associating them to individuals using Part Affinity Fields (PAFs). OpenPose is known for its versatility in detecting full-body, hand, and facial keypoints, making it highly effective for multi-person scenarios. However, it struggles with occlusions and overlapping individuals and is computationally intensive for real-time applications.
2. **DeepPose**, proposed by Toshev and Szegedy, which marked one of the earliest uses of deep learning in pose estimation. This top-down approach directly regresses body joint positions using convolutional neural networks (CNNs). While it performs well in single-person scenarios, it is inefficient for multi-person detection and lacks the speed required for real-time tasks.

3. **PoseNet** and **MediaPipe** focus on optimizing pose estimation for mobile and edge devices. PoseNet is designed for single-person pose detection and achieves quick inference, whereas MediaPipe provides real-time performance on resource-constrained devices, albeit with limited multi-person detection capabilities. Both are ideal for applications that prioritize efficiency over complex scenarios.

2.3 Limitations in Existing Solutions

1. **Challenges in Complex Backgrounds** *:* Many existing models, such as OpenPose and DeepPose, struggle to accurately detect poses in environments with cluttered or dynamic backgrounds.
2. **Real-Time Performance:** While OpenPose and other state-of-the-art models achieve high accuracy, they often require significant computational resources, making real-time pose estimation infeasible on standard hardware
3. **Scalability for Multi-Person Detection:**Multi-person pose estimation remains challenging, particularly in crowded environments. Top-down approaches like Mask R-CNN rely on individual detection first, which can be computationally expensive, while bottom-up approaches like OpenPose may struggle with grouping keypoints accurately.
4. **Usability and Accessibility** :Many pose estimation frameworks are designed for researchers or developers and lack user-friendly interfaces. This makes them less accessible to non-technical users.

2.4 How This Project Addresses These Gaps

1. Improved Detection in Complex Backgrounds: The system uses a combination of pre-trained deep learning models and advanced preprocessing techniques to handle cluttered or dynamic backgrounds. By leveraging Part Affinity Fields (PAFs) to associate keypoints, the system ensures accurate pose detection, even in busy environments.

2. **Real-Time Performance on Standard Hardware**: Lightweight frameworks, such as OpenCV's `dnn` module, are used to optimize the computational performance of pre-trained TensorFlow models. Efficient image resizing and input normalization minimize processing overhead.

3. Scalable Multi-Person Detection: The project adopts a bottom-up approach for pose estimation, which detects keypoints independently before associating them with individuals. This is both computationally efficient and scalable for multi-person scenarios.

4. User-Friendly Interface : The project employs Streamlit to create an intuitive web-based interface. Users can easily upload images, videos, or use live camera feeds to perform pose estimation with minimal effort.

CHAPTER 3

Proposed Methodology

The proposed methodology outlines the system design and implementation strategyfor the Human pose Estimation System. It ensures real-time operation, user-friendly interaction and secure data handling**.**

3.1 System Design

The system design integrates several interconnected modules to ensure smooth funcationality:

1.Input Module

- Accepts inputs from images, videos, or a live camera feed

2.Preprocessing Module

The input image or frame is preprocessed to ensure compatibility with the pose estimation model. This step includes:

- **Resizing**: Adjusting the image dimensions to match the model's input requirements (e.g., 368x368 pixels).
- **Normalization**: Standardizing pixel values to improve model performance.
- **Color Conversion**: Converting color formats (e.g., RGB to BGR) as required by OpenCV or TensorFlow.

3.Pose Estimation Module

- The core processing happens here using the pre-trained **graph_opt.pb** model.
    - **Keypoint Detection**: Identifies critical body points such as joints (e.g., nose, shoulders, knees).
    - **Part Affinity Fields (PAFs)**: Establishes connections between detected keypoints to form a skeleton of the human body.
    - The detected keypoints and their connections are visually represented on the input image or video frame.

4.Postprocessing Module

Smoothing: Temporal smoothing to ensure the pose is consistent across video frames.

Tracking: Following a specific person's pose across multiple frames or detecting multiple individuals simultaneously.

5.User Interface Module

Displays the results, such as detected skeletons and poses, overlaid on the in

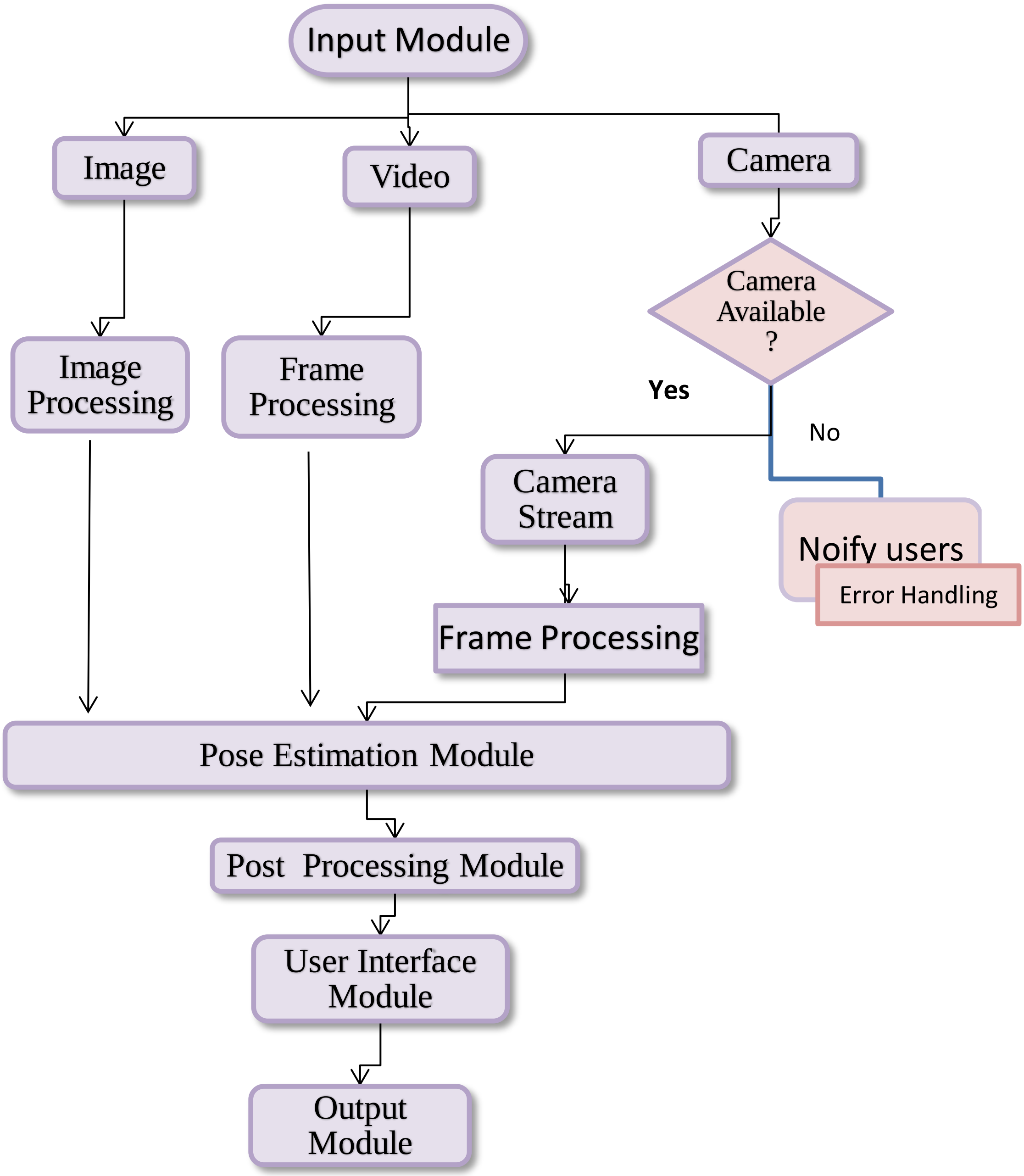

Figure 1: System workflow for Pose Estimation

3.2 Requirement Specification

3.2.1.Hardware Requirements:

1. Processing Unit: A multi-core processor with high clock speed for general computation tasks and lightweight pose estimation models.

2. Camera/Imaging Devices: High-resolution cameras to capture images or videos for pose estimation

3. RAM: Minimum 4GB to handle real-time processing efficiently.

3.2.2.Software Requirements:

1.Operating System: Windows/Linux/MacOS.

2.Programming Language: Python

3.Libraries/Frameworks:

- **OpenCV**: A popular computer vision library used for image and video processing.
- **TensorFlow:** A deep learning framework used to build, train, and deploy machine learning models.
- **Streamlit**: A Python-based framework for building interactive web applications
- **NumPy**: A library for numerical computations and array manipulation
- **PIL (Pillow)**: A Python library for image manipulation and processing.

CHAPTER 4

Implementation and Result:

4.1 Snap Shots of Result

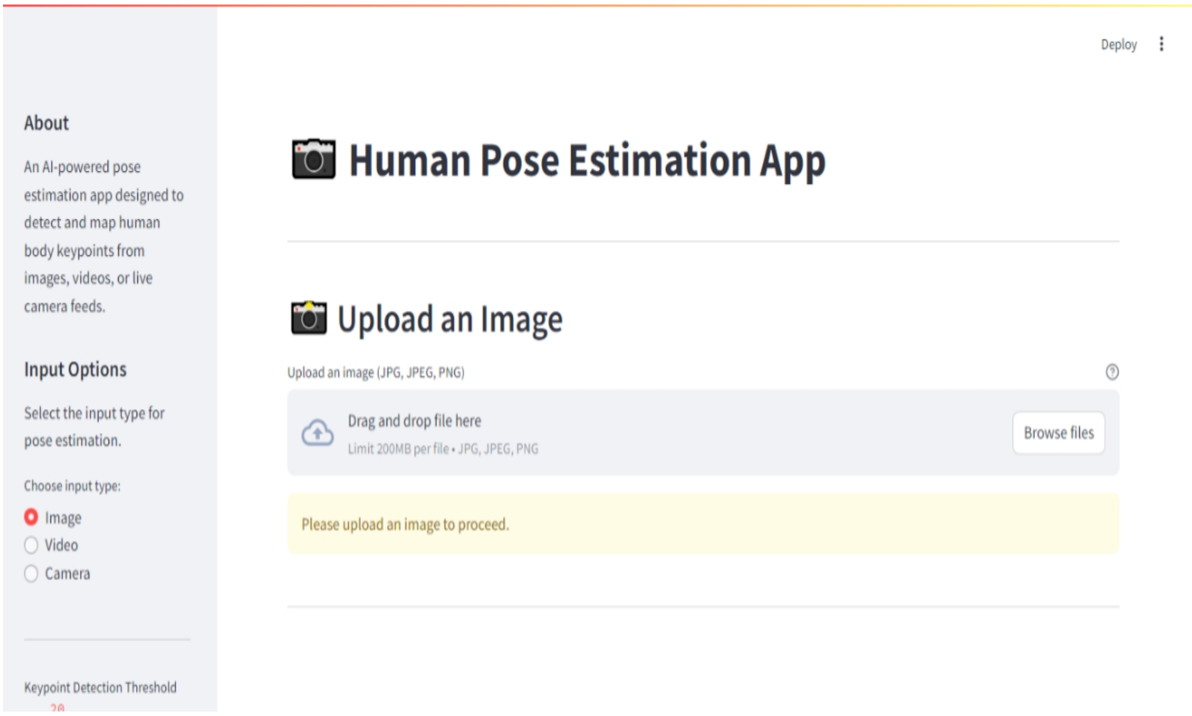

Figure 2: Web page of Pose Estimation Application

This is a First User Interface after running the program.In this Page you get the About section , Input Option.You have to select one of them. you get Result accordingly.

Figure 3: Result Image

This is a Result Image in which human Key points are detected clearly .

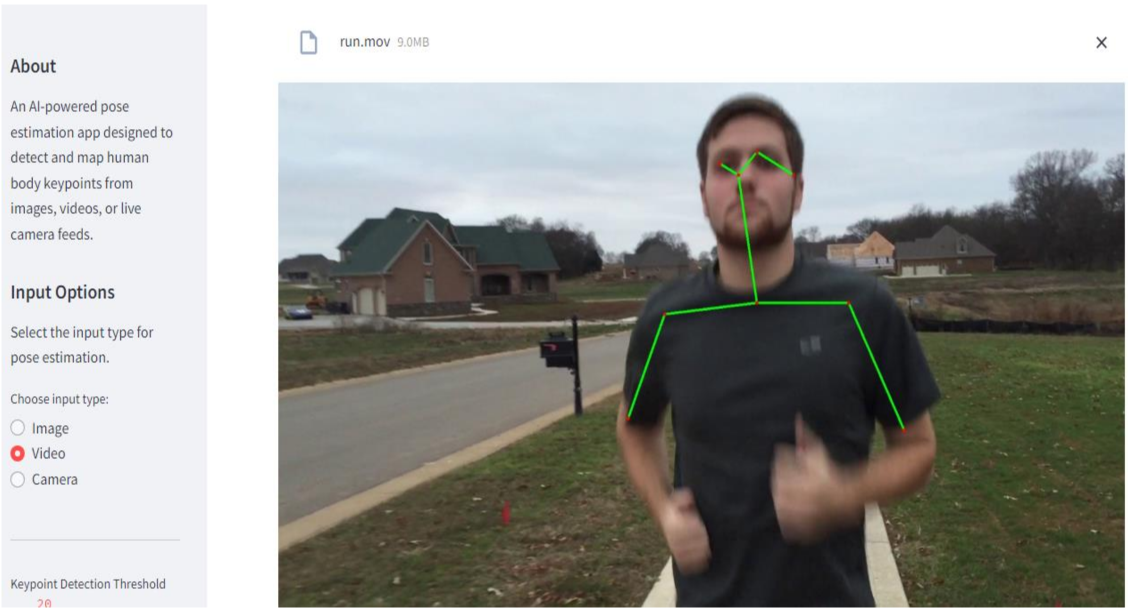

Figure 4: Result Video

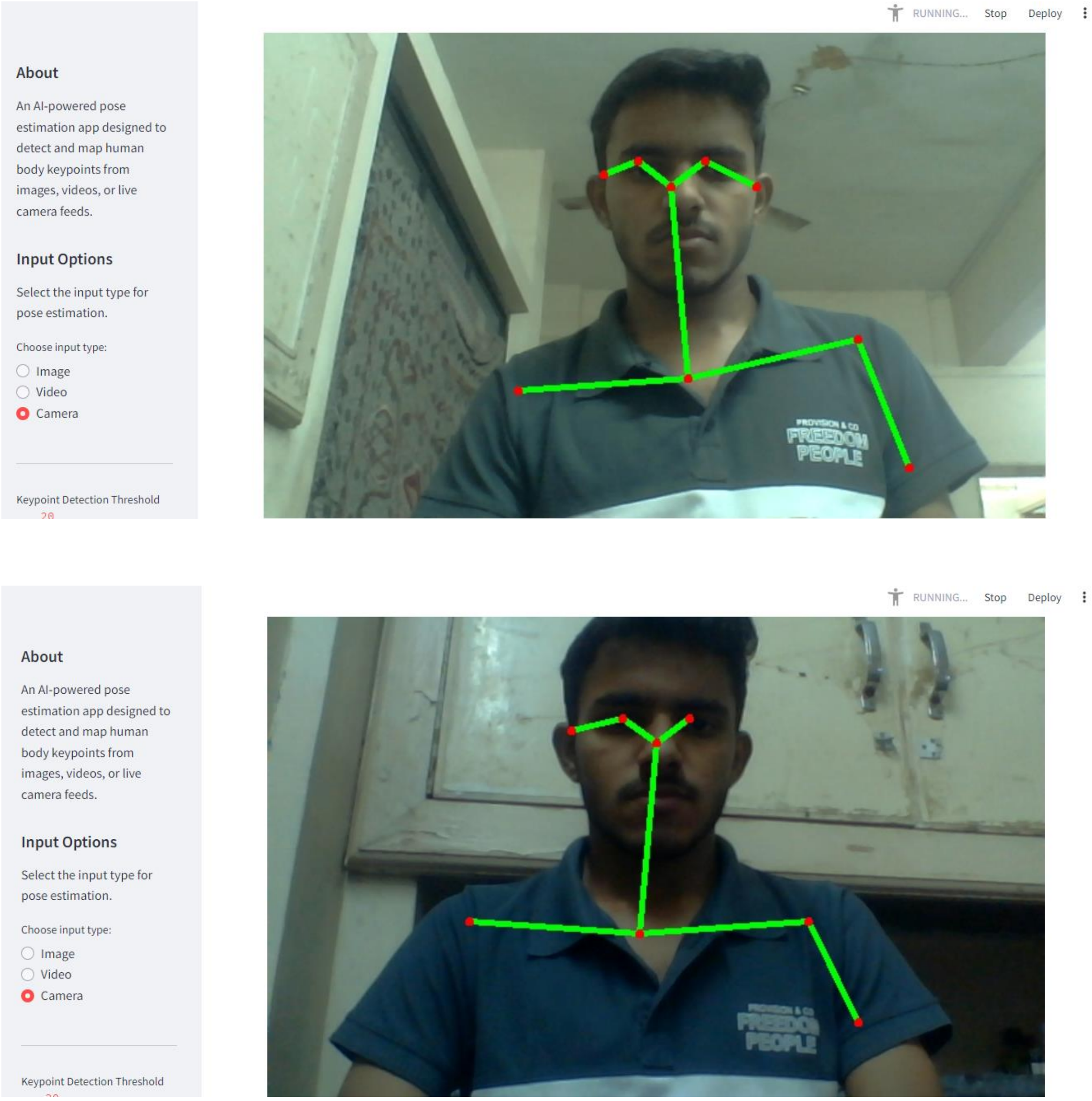

Figure 5: Result of Live webcam

4.1GitHub Link for Code:

https://github.com/Atharva0426/Pose_Estimation_Project.git

CHAPTER 5

Discussion and Conclusion

5.1 Future Work:

Future improvements to the pose estimation system include integrating activity recognition to classify movements like walking or yoga poses, enhancing occlusion handling using temporal data or advanced algorithms, and optimizing the model for edge devices through compression and quantization. Adding support for multi-person pose estimation in real-time and extending to 3D pose estimation will broaden its applications in virtual reality, augmented reality, and motion capture. Training on diverse datasets can improve generalization across varied environments, while an improved user interface with export options and analytics will enhance usability. Lastly, tailoring the system for domain-specific applications, such as healthcare and sports, along with advanced error detection mechanisms, will increase its accuracy, scalability, and practical relevance.

5.2 Conclusion:

This project successfully developed a robust and efficient pose estimation system capable of detecting and mapping human body keypoints from images, videos, and live camera feeds. By leveraging pre-trained deep learning models and integrating them with tools like OpenCV and Streamlit, the system provides accurate and scalable pose detection in real-world scenarios. Its user-friendly interface makes it accessible to non-technical users, and its real-time performance opens doors for applications in healthcare, sports analytics, human-computer interaction, and surveillance. The project addresses critical challenges such as occlusion handling, adaptability across environments, and multi-person detection. Overall, this system serves as a foundation for further advancements in pose estimation and its integration into diverse fields, significantly contributing to the accessibility and practical use of computer vision technologies.

REFERENCES

[1]. Ming-Hsuan Yang, David J. Kriegman, Narendra Ahuja, "Detecting Faces in Images: A Survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume. 24, No. 1, 2002.