**Human Pose Estimation System**

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

**Prachi Hivarkar,prachihivarkar800@gmail.com**

Under the Guidance of

**P.Raja Sir , Master Trainer , Edunet Foundation**

**ACKNOWLEDGEMENT**

I would like to take this opportunity to express our heartfelt gratitude to all individuals who supported us directly or indirectly during the development of this project.

Firstly, we extend our sincere thanks to our supervisor, Mr. P Raja Sir, for being an exceptional mentor and guide. His invaluable advice, constructive feedback, and unwavering encouragement have been a constant source of inspiration throughout the course of this project. His belief in our abilities and his insightful guidance not only facilitated the successful completion of this work but also instilled in us a deeper understanding of the subject.

It has been an honor and privilege to work under his guidance over the past month. His mentorship extended beyond the technical aspects of the project, shaping us into better professionals and individuals. The knowledge and skills gained during this time will undoubtedly be instrumental in our future endeavors.

Lastly, we would like to express our gratitude to our family, friends, and peers for their continuous support and encouragement throughout this journey.

Yours Sincerely,

Prachi Hivarkar

#### …..

#### **ABSTRACT**

Human Pose Estimation (HPE) is a technology in computer vision focused on detecting and analyzing human body positions and orientations. This project aims to develop a robust system using advanced machine learning techniques to accurately identify and track human poses in real-time. The system has applications in fitness monitoring, gesture recognition, sports analysis, healthcare, and human-computer interaction.

The proposed approach leverages deep learning architectures, such as Convolutional Neural Networks (CNNs) and Transformer models, trained on datasets like COCO and MPII. It detects key points on the human body, including joints and limbs, to construct skeletal representations of poses. Preprocessing techniques address variations in lighting, background, and occlusions, ensuring adaptability to diverse environments.

A major focus is on optimizing computational efficiency to enable real-time processing on resource-constrained devices like smartphones and embedded systems. The system also supports multi-person pose estimation, making it suitable for group activities such as sports or fitness classes.

The implementation includes a user-friendly interface for pose visualization and actionable insights, such as correcting posture or improving performance. Additionally, the system has potential applications in healthcare, such as monitoring rehabilitation exercises and aiding individuals with physical challenges.

By integrating computer vision, deep learning, and practical application design, this project aims to deliver an accurate, efficient, and adaptable HPE system. It promises to enhance human activity analysis across various domains, fostering innovative solutions for real-world challenges

**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.** |
|  | PSS Model | **5** |
|  | DeepPose Architecture Model | **5** |
|  | Architecture Diagram | **6** |
|  | User Interface Image | **11** |
|  | Uploaded Image | **11** |
|  | Extracted Image | **12** |
|  | Uploaded Image | **12** |
|  | Pose Estimated Image | **13** |
|  | Uploaded Video | **13** |
|  | Extracted Pose | **14** |

**CHAPTER 1**

**Introduction**

* 1. **Problem Statement:**

Human Pose Estimation (HPE) systems using AI and ML address the need for accurate, real-time analysis of human body poses. Traditional methods are costly, labor-intensive, and struggle with scalability, while challenges like occlusions and device constraints hinder existing solutions.

This problem is significant due to applications in healthcare, fitness, sports, and human-computer interaction. By enabling real-time posture analysis and activity tracking, the proposed system offers an accessible, efficient, and scalable solution to meet the growing demand for advanced human activity analysis.

* 1. **Motivation:**

This project was chosen to address the limitations of traditional methods for human activity monitoring, which are often costly, labor-intensive, and lack real-time capabilities. By leveraging AI, ML, and Streamlit, the project aims to create an efficient, scalable, and user-friendly system for accurate pose analysis. It aligns with the increasing demand for advanced yet accessible solutions to understand and improve human activity across various fields.

The system has broad applications in healthcare, fitness, sports, and human-computer interaction. It can assist in rehabilitation, enhance athletic performance, enable gesture-based controls, and improve security monitoring. With its focus on real-time analysis and scalability, this project promises to make human pose estimation technology widely accessible, driving innovation and impactful solutions across industries.

* 1. **Objective:**

The primary objectives of this project are:

* **Accurate Pose Detection:** Develop a system capable of identifying key points on the human body (joints and limbs) with high precision across various environments and scenarios.

 **Real-Time Analysis:** Ensure the system operates in real-time, making it suitable for dynamic applications such as fitness monitoring, sports analysis, and interactive interfaces.

** Adaptability and Scalability:** Design the system to handle diverse lighting conditions, backgrounds, occlusions, and multi-person scenarios without compromising performance.

** User-Friendly Interface:** Implement an interactive and accessible interface using Streamlit to visualize pose estimation results and provide actionable insights.

** Resource Efficiency:** Optimize computational efficiency to enable the system to run on resource-constrained devices like smartphones and embedded systems.

** Broad Applicability:** Address use cases in healthcare, fitness, sports, human-computer interaction, security, and content creation by offering practical and impactful solutions.

** Cost-Effectiveness:** Develop a solution that is affordable and accessible to a wide range of users, including individuals and professionals.

* 1. **Scope of the Project:**

The scope of the Human Pose Estimation System involves designing and implementing a real-time solution to detect and analyze human body posture and movements using computer vision and machine learning techniques. Utilizing tools such as OpenCV, Streamlit, and the Mediapipe library, this system can identify key body landmarks and provide actionable insights for various applications, including fitness tracking, ergonomic assessments, gesture recognition, sports analytics, and healthcare monitoring. The project aims to deliver an efficient, user-friendly platform capable of processing live video streams or pre-recorded footage with high accuracy and minimal computational requirements, making it versatile and accessible for diverse use cases.

Limitations:

The Human Pose Estimation System has a few limitations:

1. Lighting Issues: The system works best with good lighting, and poor or uneven lighting can affect its accuracy.
2. Obstructions: If parts of the body are blocked or overlapping, the system may struggle to detect poses correctly.
3. Diversity of Users: The system might not perform as well for people with different body types, clothing, or other variations not covered in the training data.
4. Hardware Demands: The system can require powerful computers, and may not work well on older or less powerful devices.
5. Speed and Action Complexity: It may have trouble detecting very fast movements or unusual actions that it hasn't been trained on.
6. Environmental Sensitivity: Changes in camera angle, distance, or background can affect performance.
7. Lack of Context: While it can identify poses, the system can't understand what the person is actually doing or the context behind the actions.

**CHAPTER 2**

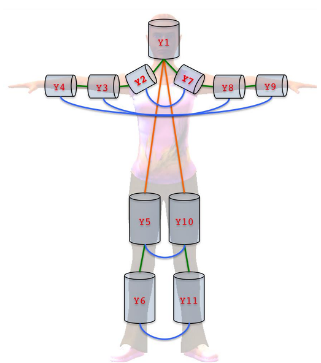
**Literature Survey**

The concept of human pose estimation has evolved significantly over the years, beginning with early methods like Pictorial Structures (PSs) introduced by Fischler and Elschlager. These models represented the human body as a graph of parts, made practical by Felzenszwalb and Huttenlocher using the distance transform trick. Despite their efficiency, traditional PS-based methods were limited by simple binary potentials and tree-based models, which could not incorporate complex image-dependent data. Subsequent advancements focused on enhancing these models by using richer part detectors and hierarchical approaches. Notable contributions include Yang and Ramanan’s mixture model of parts and Johnson and Everingham’s exploration of mixtures at the model scale. These developments marked a shift towards capturing complex joint relationships and higher-order spatial dependencies in pose estimation.

Modern approaches leverage deep learning techniques to achieve more robust and accurate pose estimation. Convolutional Neural Networks (CNNs) and regression-based methods, such as those by Ionescu et al., have enabled significant strides in 2D and 3D pose estimation. Image-dependent PS models, such as those using global classifiers (e.g., Gkioxari et al.), have shown promise but are often limited in their generalizability. Furthermore, cascades of deep neural network regressors, though commonly applied to facial point detection, are underexplored for full-body pose estimation. With advancements like MediaPipe and other lightweight frameworks, real-time, accurate, and user-friendly solutions are emerging, addressing earlier challenges such as robustness to occlusion, computational overhead, and usability across diverse applications.

Some Existing Models, Techniques or Methodologies :

1.Pictorial Structures (PSs):  
Introduced by Fischler and Elschlager, PS models represent the human body as a graph of parts connected by joints. Felzenszwalb and Huttenlocher improved the model's computational efficiency using the distance transform trick. While simple and effective for early pose estimation, these models lacked the ability to handle complex relationships and image-dependent variations.

Top of Form

Bottom of Form

Fig PSs Model

2. DeepPose Model:  
DeepPose was one of the first methods to apply deep learning to pose estimation. It treated pose estimation as a regression problem using Convolutional Neural Networks (CNNs) to predict keypoint coordinates. This approach marked a significant improvement over traditional hand-crafted feature methods by leveraging data-driven learning.

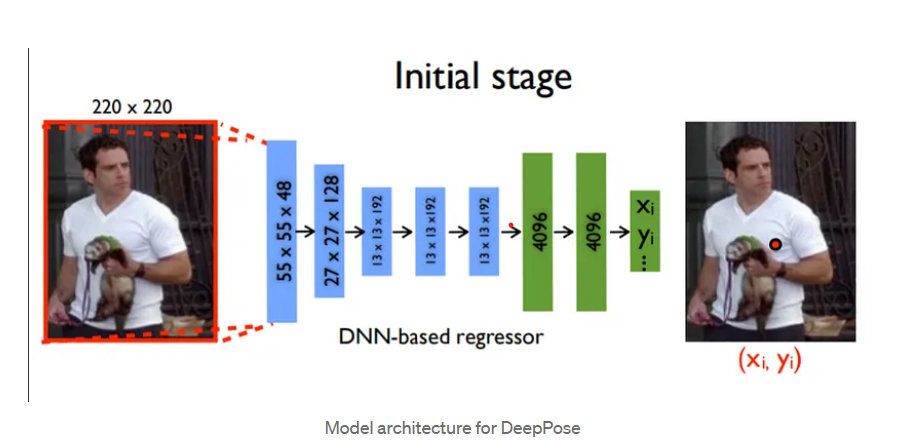
 

Fig 2: DeepPose Model

\

**CHAPTER 3**

**Proposed Methodology**

**System Design**

Provide the diagram of your Proposed Solution and explain the diagram in detail

.

User Interface

Preprocessing

Input Video

Human Detection |

Pose Estimation Model

Post-Processing

Pose Visualization

Output (Keypoints)

Fig 3: Architecture Diagram

Detailed Description of the Human Pose Estimation System Using OpenCV, Streamlit, and MediaPipe

1. Input Image/Video:
   * The system begins by receiving an input in the form of an image or a video. This input can be provided by the user either through a file upload mechanism in Streamlit or through a live webcam feed using OpenCV.
   * In Streamlit, the user can upload an image or video using the file\_uploader() method, while in OpenCV, the webcam feed is captured using cv2.VideoCapture(). These inputs are then passed to the preprocessing step for further handling.
2. Preprocessing:
   * Once the input image or video is received, it is processed to ensure it is in the correct format for pose estimation. This involves resizing the image or video frames to a standard resolution, typically to maintain consistency and optimize performance. The resizing is done using OpenCV's cv2.resize() function.
   * Additionally, the image is converted from the BGR color format (default in OpenCV) to the RGB color format. This conversion is necessary because the MediaPipe Pose model expects the input in RGB format for accurate processing. This is done using the cv2.cvtColor() function in OpenCV.
3. Human Detection:
   * The next step involves detecting humans in the image or video and localizing key body landmarks. This is achieved using MediaPipe’s Pose model, which is capable of identifying human figures and locating 33 specific body parts such as the head, shoulders, elbows, wrists, hips, knees, and ankles.
   * MediaPipe processes the input frame and uses machine learning models to detect these keypoints. The system returns landmarks that are spatially arranged, and each landmark has a set of coordinates (x, y) along with a confidence score indicating the accuracy of the detection.
4. Pose Estimation Model:
   * MediaPipe’s Pose model is responsible for the core task of pose estimation. It estimates the positions of the 33 keypoints on the human body. These keypoints include the main joints and parts like the nose, eyes, ears, shoulders, elbows, hands, waist, knees, and feet.
   * For each detected human figure, the model generates a set of keypoint coordinates and assigns each one a confidence score. The system uses this data to establish the body’s pose, and it outputs the keypoints in terms of normalized (x, y) positions relative to the input image’s dimensions.
   * The keypoints are crucial for identifying the body’s posture and movements.
5. Post-Processing:
   * After the pose estimation, post-processing is performed to enhance the accuracy and stability of the keypoint data. This involves filtering out any detected keypoints that have low confidence scores. The system typically sets a threshold (e.g., 0.5), and any landmarks below this threshold are ignored.
   * For video processing, temporal smoothing techniques are applied to ensure that the keypoints are tracked consistently across multiple frames. This helps reduce jitter and ensures smooth transitions as the person moves. Additionally, any errors or inconsistencies in landmark detection due to rapid movement or occlusion are minimized.
6. Pose Visualization:
   * Once the keypoints have been detected and filtered, they are visualized by drawing a skeleton over the input image or video. The keypoints are connected by lines to form a skeleton that represents the posture of the person.
   * OpenCV is used to draw the keypoints as circles and connect them using lines, while MediaPipe provides a predefined set of connections between the landmarks (known as POSE\_CONNECTIONS) to form the skeletal structure.
   * This visualization step makes it easy for users to see the detected human pose, as it overlays the keypoints and lines on the original image, showing the positions of various body parts in relation to one another.
7. Output:
   * The final output is displayed to the user, where the processed image or video is shown with the detected human pose overlaid on it. This output can either be in the form of a static image or a real-time video stream.
   * In Streamlit, the final annotated frame is displayed using st.image() for images or st.video() for videos, allowing the user to interactively view the results. The system also allows for the output to be saved or used for further analysis, depending on the application’s needs.
   * Additionally, the system may provide raw keypoint data (the (x, y) coordinates of each landmark), which can be utilized for further analysis, such as tracking specific body movements, calculating angles between joints, or even performing activity recognition.

**Requirement Specification**

* + 1. **Hardware Requirements:**
* Processor: Minimum 2 GHz dual-core CPU or Intel i5 or high performance.
* RAM: At least 4 GB (8 GB recommended for better performance)
* Storage: Minimum 500 MB of free space
* Internet Connection: Required for Streamlit application deployment and testing.

* + 1. **Software Requirements:**
* IDE: VS Code or PyCharm for writing and testing Python scripts.
* Browser: Modern browser (Google Chrome or Firefox) for accessing the app.
* Operating System: Windows 10/11, macOS, or Linux (Ubuntu recommended).
* Programming Languages: Python 3.x for application development.
* Libraries: Streamlit, MediaPipe, OpenCV, NumPy, and optionally Matplotlib.
* Streamlit Web Deployment: Streamlit sharing, Heroku, or AWS for hosting.

**CHAPTER 4**

**Implementation and Result**

* 1. **Snap Shots of Result:**

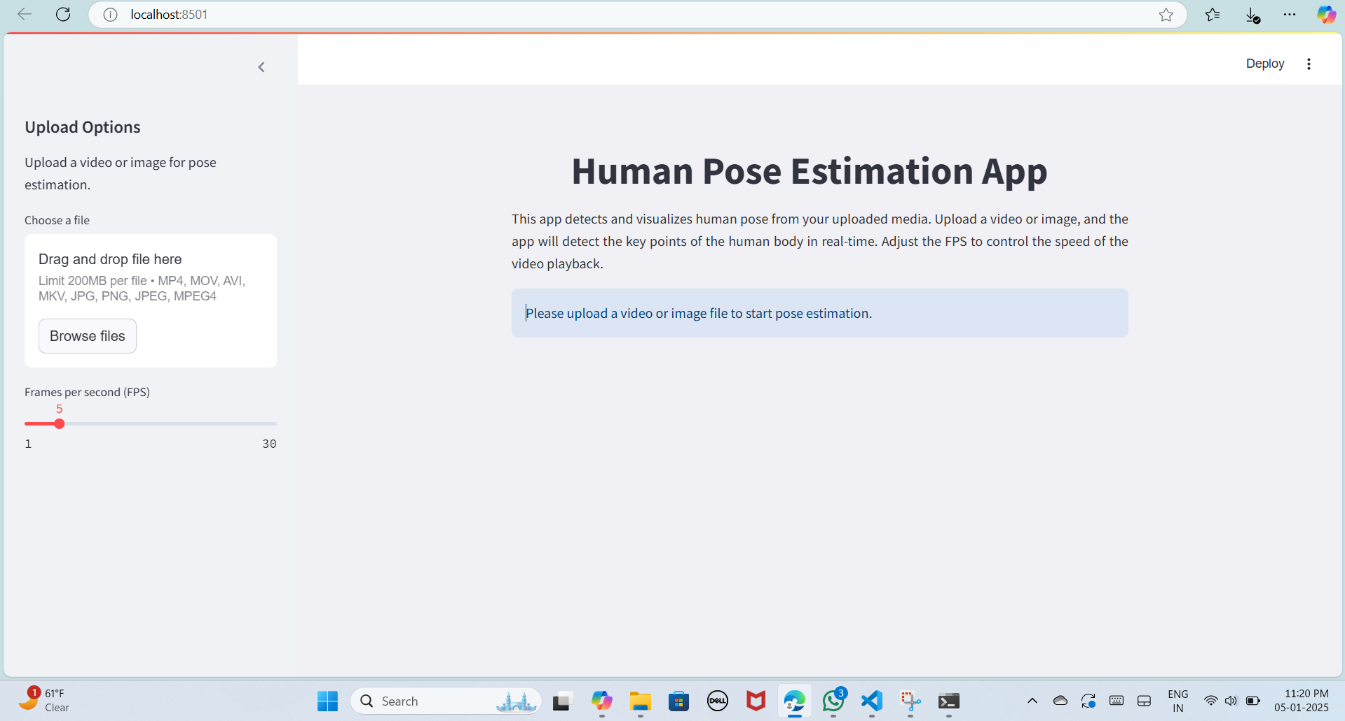
. 

Fig 4: User Interface

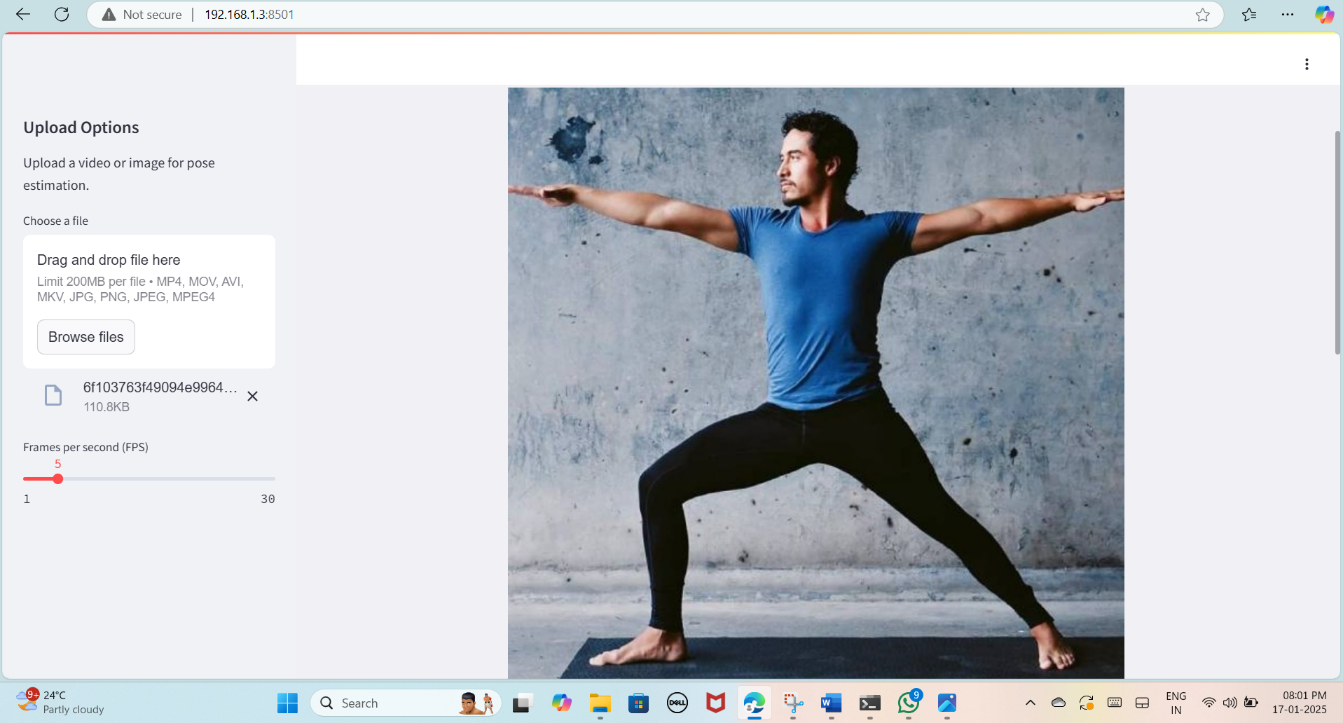


Fig 5: Uploaded Image



Fig 6: Extracted Image

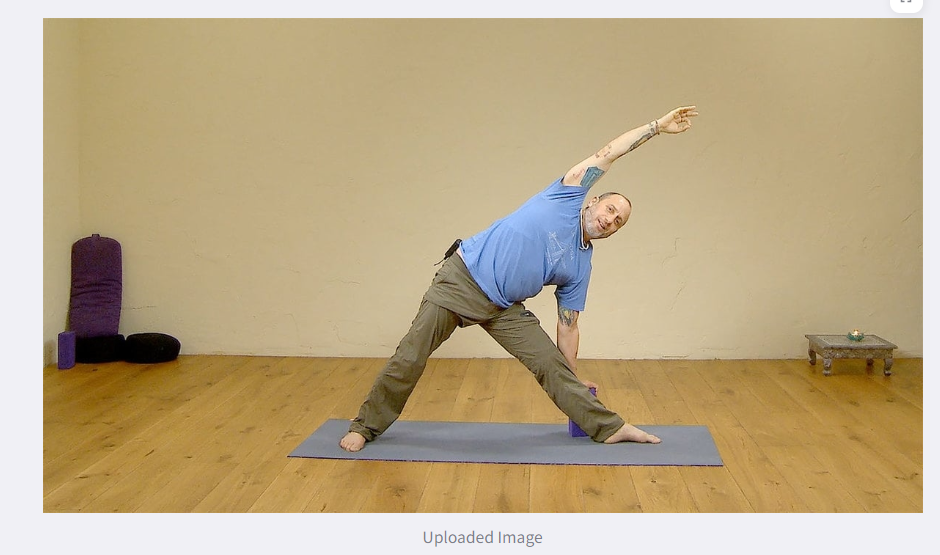


Fig 7: Uploaded Image

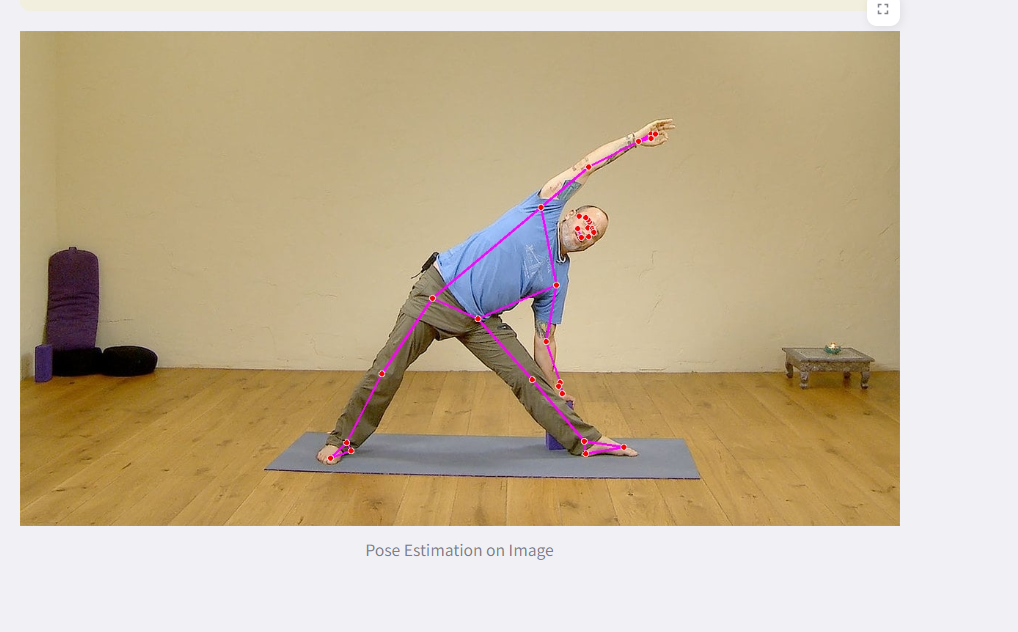


Fig 8: Pose Estimated Image



Fig 9: Uploaded Video

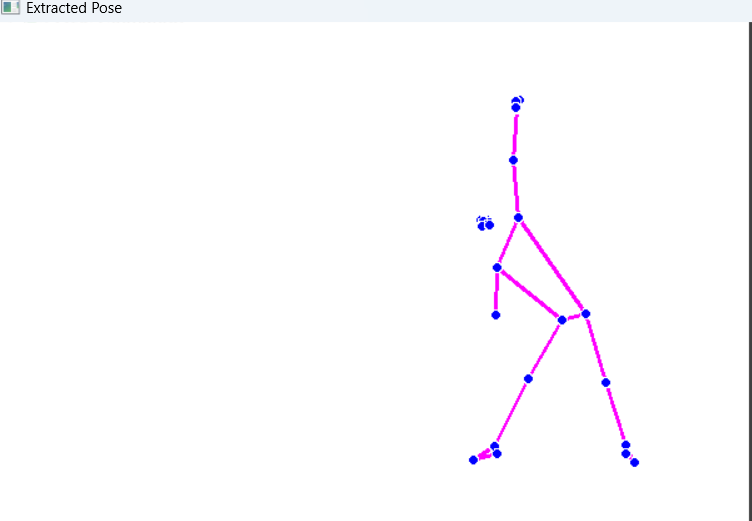


Fig 10 : Extracted Pose

* 1. **GitHub Link for Code:**

<https://github.com/PrachiHivarkar/Human-Pose-Estimation-App.git>

**CHAPTER 5**

**Discussion and Conclusion**

* 1. **Future Work:**

Model Improvements

1. Architecture Enhancements
   * Explore transformer-based architectures that have shown promise in computer vision tasks.
   * Investigate hybrid approaches combining convolutional neural networks (CNNs) with graph neural networks (GNNs) to model relationships between joints.
2. Loss Function Optimization
   * Design custom loss functions that better handle occlusions, multi-person interactions, and depth estimation in 3D pose tasks.
   * Incorporate adversarial training to improve robustness against challenging scenarios.
3. Dataset Augmentation
   * Generate synthetic data using techniques like GANs (Generative Adversarial Networks) to augment datasets with rare or extreme poses.
   * Use domain randomization to simulate diverse environments, reducing dependency on real-world data.
4. Handling Occlusions and Noise
   * Develop occlusion-aware models that predict missing keypoints using context from visible joints.
   * Implement denoising techniques to handle sensor inaccuracies or artifacts in images.
5. Multi-View and Temporal Analysis
   * Integrate multi-view learning to provide complementary perspectives, improving accuracy in occluded or ambiguous scenarios.
   * Use temporal models like LSTMs or Transformers to incorporate sequential context for smoother pose predictions.

Performance Optimization

1. Lightweight Models
   * Create models tailored for edge devices with lower memory and power requirements using techniques like model pruning, quantization, or distillation.
2. Real-Time Enhancements
   * Optimize inference pipelines to reduce latency for real-time applications, particularly in sports analytics, surveillance, and AR/VR.

Unresolved Issues

1. Bias and Fairness
   * Address biases in datasets to improve performance across diverse populations, body types, and clothing styles.
   * Use federated learning for privacy-preserving data collection from diverse demographics.
2. Rare Pose Handling
   * Focus on improving recognition of uncommon or complex poses through targeted data augmentation and fine-tuning strategies.
3. Generalization Across Domains
   * Apply domain adaptation techniques to ensure the model performs well across varied scenarios, such as different camera setups, lighting conditions, and cultural contexts.
4. 3D Pose Estimation Challenges
   * Refine depth estimation for 3D pose prediction to ensure accuracy in monocular setups.
   * Investigate combining 2D and 3D pose models to enhance reliability in diverse applications.

* 1. **Conclusion:**

The development of a human pose estimation system represents a transformative advancement in artificial intelligence and computer vision, offering precise identification and analysis of human movement. By capturing the positions and orientations of key body joints, it enables a wide spectrum of applications across industries, including healthcare, sports, entertainment, and robotics. The system excels in handling complex scenarios, such as multi-person tracking in crowded environments or managing partial occlusions caused by overlapping subjects or environmental factors. Its robustness is attributed to advanced machine learning architectures, such as convolutional neural networks (CNNs) and transformers, which extract spatial and temporal features with exceptional accuracy.

Furthermore, the incorporation of optimization techniques, including model compression, pruning, and quantization, ensures the system operates efficiently even on resource-constrained devices like smartphones or edge devices. This makes it suitable for real-time applications, such as virtual reality (VR) interfaces, live sports performance analysis, and surveillance systems. Additionally, its versatility is enhanced by the ability to generalize across diverse environments, from outdoor sports fields to indoor healthcare facilities, through techniques like domain adaptation and data augmentation. By bridging the gap between understanding human behavior and implementing intelligent systems, human pose estimation drives innovation, creating opportunities for more natural human-computer interactions and paving the way for groundbreaking advancements in AI-powered 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.
2. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y., 2019. Openpose: Realtime multi person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence 32, 172–186.
3. Kim, J.W., Choi, J.Y., Ha, E.J., Choi, J.H., 2023. Human pose estimation using mediapipe pose and optimization method based on a humanoid model. Applied Sciences 13, 2700.
4. H-W, Kim J (2022) Real-time workout posture correction using OpenCV and MediaPipe. J Korean Inst Inf Technol 20(2):305–312
5. Cho H, Lee S (2023) A real-time posture detection and correction system for office workers using MediaPipe and OpenCV. In: Proceedings of the international conference on software engineering and information technology, pp 231–23
6. A. Rahmadani, B. S. Bayu Dewantara and D. M. Sari, "Human Pose Estimation for Fitness Exercise Movement Correction," 2022 International Electronics Symposium (IES), Surabaya, Indonesia, 2022, pp. 484-490
7. Hruthika, Jupalle, et al. "Deep Learning Based Human Pose Estimation Using Opencv." *International Journal of Innovations in Engineering Research and Technology*, vol. 7, no. 12, 2020, pp. 246-253.
8. Streamlit Tutorial by Intellipaat:

<https://www.youtube.com/live/YzvMpvXyUfs?si=ypQNMW64WLwQwppq>

1. MediaPipe Tutorial on youtube:

<https://youtu.be/bZKO8jwOfjg?si=Dn7JGWBcy5jdfaF0>