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

Manav Shah, manavshah1704@gmail.com

Under the Guidance of

P.Raja, Master TrainerEdunet Foundation

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to everyone who supported me throughout this project.

Firstly, I extend my sincere thanks to my guide, **P. Raja**, Master Trainer at Edunet Foundation, for his valuable guidance, encouragement, and insightful advice during the development of this project. His mentorship has been instrumental in completing this work successfully.

I would also like to thank **Pawan Sir** for his continuous support, motivation, and assistance, which made this journey smoother and more enriching.

Lastly, I am grateful to everyone who contributed directly or indirectly to this project and helped me grow as a learner.

Manav Shah

ABSTRACT

The project, Human Pose Estimation using Machine Learning, focuses on developing a system capable of detecting and analyzing human body poses from images or video sequences. Understanding human movements and postures is challenging in fields like sports, healthcare, and surveillance. Without pose estimation, tasks such as motion analysis and injury prevention become manual, slow, and error prone.

The objective of this project is to design and implement a machine learning model to identify key body landmarks and predict human postures accurately. A structured approach was followed, including data collection, preprocessing, and identifying key attributes to select the most suitable algorithm. The model was trained, evaluated, and deployed to ensure accuracy, reliability, and applicability in real-world scenarios.

The results demonstrated that the proposed model effectively detects human poses, showcasing its potential in various applications. In conclusion, this project highlights the role of machine learning in addressing complex pose estimation challenges and provides a foundation for further research in activity recognition and rehabilitation monitoring.

TABLE OF CONTENT

LIST OF FIGURES

CHAPTER 1

1. Introduction

1.1 Problem Statement:.

Understanding human movements and postures is challenging in areas like sports, healthcare, and surveillance. Without pose estimation, tasks such as motion analysis and injury prevention become manual, slow, and error-prone

1.2 Motivation:

I choose this project because it focuses on human pose estimation and incorporates the use of OpenCV, as well as photos and videos, to apply machine learning models. I have always been excited about working on AI projects that involve processing visual data, such as images and videos. This project aligns perfectly with my interests and gives me an opportunity to explore and implement machine learning techniques in an area that genuinely excites me.

1.3 Objective:

The objective of this project is to design a reliable machine learning system for human pose estimation, leveraging computer vision techniques to analyze images and videos. It aims to enable real-world applications in healthcare, sports, and surveillance.

1.4 Scope of the Project:

□ **Applications**:

1. **Healthcare**: Monitor rehabilitation progress by analyzing movement.
2. **Sports**: Improve athletic performance by analyzing posture and technique.
3. **Surveillance**: Enhance security by detecting unusual movements.
4. **Activity Recognition**: Recognize activities like walking or running for fitness tracking.

□ **Limitations of the Project:**

1. **Environmental Factors**: Performance may degrade in poor lighting or when parts of the body are occluded.
2. **Real-Time Processing**: High computational power may be required for real-time video analysis, limiting its use on devices with lower specs.

3. **Accuracy in Extreme Poses**: The model may struggle with complex or extreme body postures due to algorithm limitations.
4. **Data Dependency**: The model's performance relies on the quality and diversity of the training data. Poor or narrow data may lead to suboptimal results.

CHAPTER 2
2. Literature Survey

2.1 literature referred:

- OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields (Zhou et al., 2018). OpenPose Paper
- Convolutional Pose Machines (Wei et al., 2016). CPM Paper
- AlphaPose: Real-Time and Accurate Human Pose Estimation (Fang et al., 2017). AlphaPose Paper
- DeepCut: Joint Subset Partition and Labeling for Multi Person Pose Estimation (Insafutdinov et al., 2016). DeepCut Paper
- PoseNet: A Convolutional Network for Real-Time Single-Object Detection (Google, 2017). PoseNet Paper

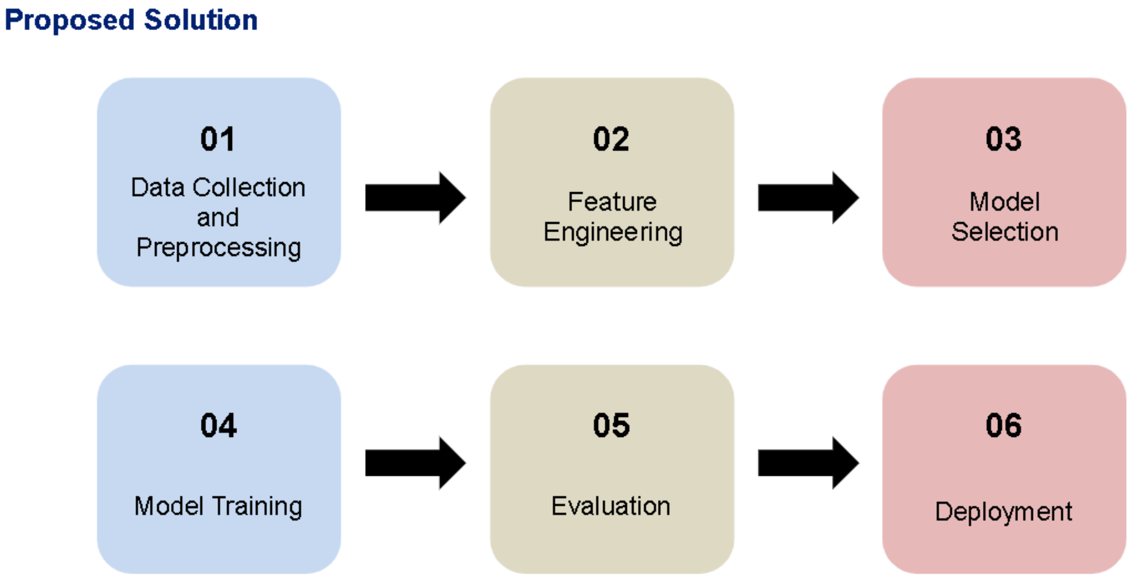
2.2 Other Existing Models and Techniques Related to the Problem:

1. **OpenPose**: One of the most popular models for real-time multi-person pose estimation. It uses part affinity fields to predict keypoints, making it suitable for a variety of applications like fitness, healthcare, and surveillance.
2. **PoseNet**: A more lightweight model developed by Google that can estimate human poses in real-time on mobile devices. PoseNet is less resource-intensive, making it suitable for low-power devices while maintaining reasonable accuracy.
3. **HRNet**: High-Resolution Network, known for its ability to maintain high-resolution representations through the entire network, providing superior performance in pose estimation tasks.
4. **AlphaPose**: As mentioned, AlphaPose is optimized for high-performance multi-person pose estimation and operates with high accuracy in dense crowds and complex scenes.
5. **DeepLabCut**: A deep learning toolbox that allows researchers to train pose estimation models tailored to their specific use case (e.g., animal motion tracking, human pose analysis in specialized fields). It’s known for its flexibility and ease of customization.

CHAPTER 3

3. Proposed Methodology

3.1 System Design



Proposed Solution

01 Data Collection and Preprocessing

02 Feature Engineering

03 Model Selection

04 Model Training

05 Evaluation

06 Deployment

Figure 3.1

3.2 Requirement Specification

3.2.1 Hardware Requirements:

1. **CPU**: Intel Core i5 (min), Intel Core i7 or Ryzen 7 (recommended).
2. **GPU**: NVIDIA GTX 1050 Ti (min), GTX 1660 Ti or RTX 2060 (recommended).
3. **RAM**: 8 GB (min), 16 GB (recommended).
4. **Storage**: 100 GB (min), 500 GB SSD (recommended).
5. **Camera**: 1080p or 4K camera (30 fps+), optional depth camera (e.g., Intel RealSense).
6. **Other Devices**: Optional microphone, VR headset for integration.

3.2.2 Software Requirements:

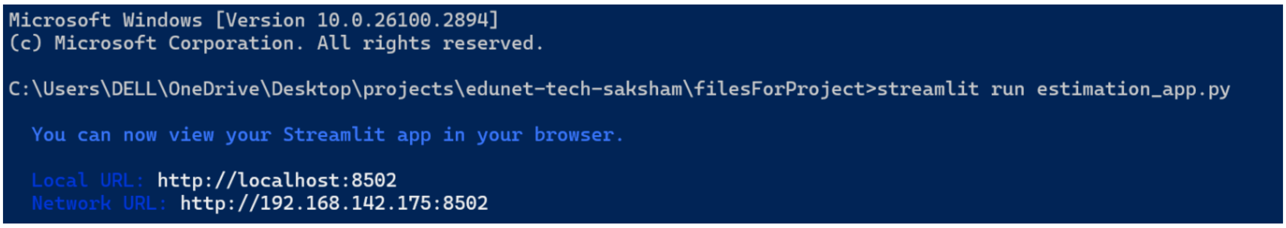
1. **Operating System**: Windows 10+ or Linux (Ubuntu 18.04+).
2. **Programming Languages**: Python 3.x, C++ (for optimization).
3. **Libraries**:
    - **Machine Learning**: TensorFlow, PyTorch, Keras.
    - **Computer Vision**: OpenCV, Dlib, MediaPipe.
4. **Data Management**: NumPy, Pandas, Matplotlib/Seaborn for visualization.
5. **Development Environment**: Jupyter Notebook, PyCharm/VS Code.
6. **Version Control**: Git, GitHub/GitLab.
7. **Cloud Services**: Google Colab (for GPU), AWS/Azure (for large-scale training).
8. **Containerization (Optional)**: Docker for environment isolation.

CHAPTER 4

4. Implementation and Result

Snap Shots of Result:

Snapshot 1: Running the Pose Estimation Model using Streamlit



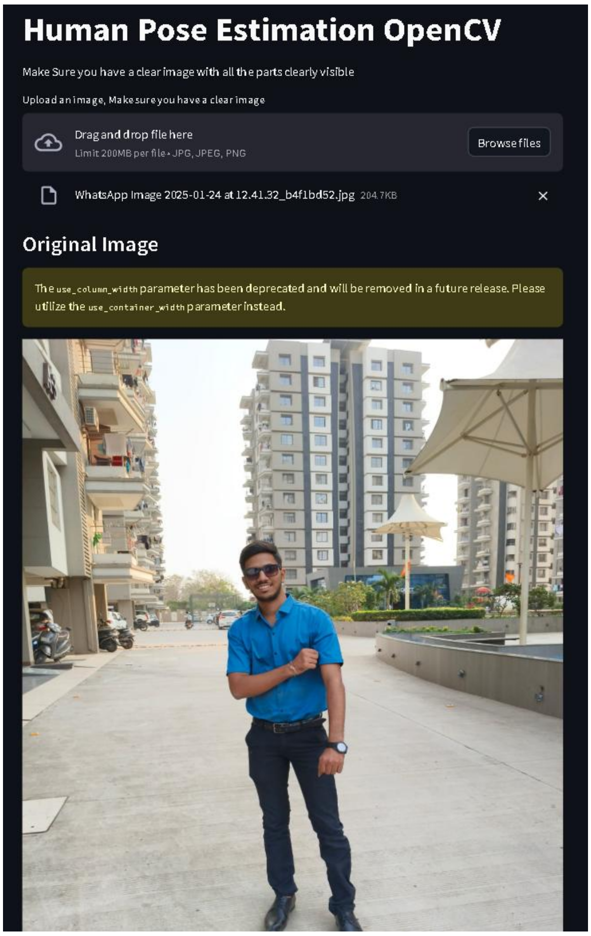
```
Microsoft Windows [Version 10.0.26100.2894]
(c) Microsoft Corporation. All rights reserved.

C:\Users\DELL\OneDrive\Desktop\projects\edunet-tech-saksham\filesForProject>streamlit run estimation_app.py

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8502
  Network URL: http://192.168.142.175:8502
```

Figure 4.1

Description: The first screenshot displays the execution of the human pose estimation project via Streamlit. The command is executed in the command prompt (CMD) using streamlit run <filename>.py, which launches the application in a web browser. The UI shows a simple interface where the user can upload an image or video, and the system will process it to detect and predict human poses.

Explanation: This snapshot illustrates the setup and deployment of the pose estimation model using Streamlit, which provides an interactive web interface for the end user. The user can interact with the model directly by uploading images and viewing real-time predictions. This showcases the ability to run the model locally and interact with it via an easy-to-use GUI.

Snapshot 2: Pose Estimation in Action:



Human Pose Estimation OpenCV

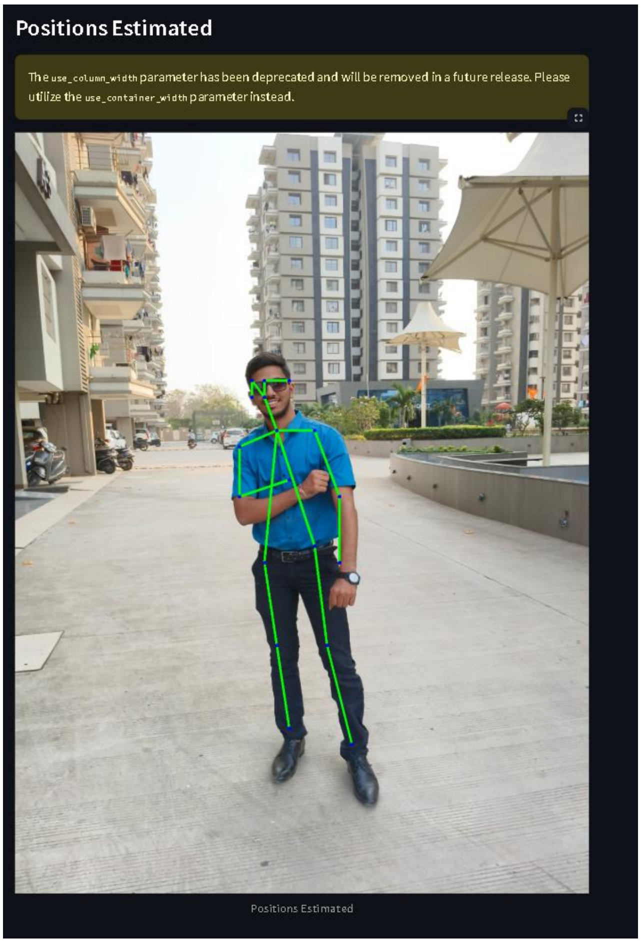
Make Sure you have a clear image with all the parts clearly visible

Upload an image, Make sure you have a clear image

WhatsApp Image 2025-01-24 at 12.41.32_b4f1bd52.jpg 204.7KB

Original Image

The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.



Positions Estimated

The use_column_width parameter has been deprecated and will be removed in a future release. Please utilize the use_container_width parameter instead.

Figure 4.2

Description: The second screenshot demonstrates the actual working of the pose estimation system. It shows an image (uploaded by the user) with the model's predictions overlaid. The human keypoints and body landmarks are marked on the image to represent the detected pose, with the skeleton of the body shown in real-time.

Explanation: This snapshot highlights the core functionality of the project—real-time human pose detection. The model processes the image to identify key body landmarks (e.g., head, elbows, knees) and outputs a skeleton representation on the person. The red dots represent the detected keypoints, and the lines connecting them indicate the estimated pose structure. This visualization is crucial for applications in healthcare, fitness, and surveillance, where accurate body posture analysis is needed.

GitHub Link for Code:

GitHub link

CHAPTER 5

5. Discussion and Conclusion

Future Work:

❖ Suggestions for Model Improvement and Future Work:

1. **Accuracy Enhancement**:
   - Use diverse datasets (COCO, MPII) and apply data augmentation for better generalization..
2. **3D Pose Estimation**:
   - Extend to 3D pose estimation for applications in VR, gaming, and motion capture.
3. **Complex Background Handling**:
   - Use background removal or segmentation (e.g., Mask R-CNN) to enhance pose detection in cluttered environments.
4. **Multimodal Integration**:
   - Combine pose estimation with depth sensors or wearable sensors for more accurate movement analysis in healthcare and fitness.
5. **Personalization & Feedback**:
   - Personalize pose estimation based on individual users' characteristics, especially for fitness or rehabilitation.
6. **Scalability & Deployment**:
   - Scale the model via cloud platforms and deploy on mobile devices for real-time pose detection applications.

Conclusion:

Overall Impact and Contribution of the Project:

The Human Pose Estimation using Machine Learning project contributes to advancing AI applications in understanding human movement. By accurately detecting body poses from images and videos, it has practical implications in healthcare, sports, fitness, and surveillance. It enables real-time motion analysis, injury prevention, and performance tracking.

Key contributions include:

- **Real-Time Pose Detection**: The model allows real-time human pose detection, making it useful in dynamic environments like sports coaching or security monitoring.

- **Machine Learning and Computer Vision**: The project uses advanced machine learning techniques to enhance the accuracy of pose prediction, improving body movement analysis.

- **Practical Impact**: It holds value in applications such as injury prevention in healthcare, performance improvement in sports, and security monitoring in surveillance.

This project lays the groundwork for further developments in AI-driven human activity recognition and has the potential to influence various fields by enabling better movement analysis and personalized health solutions.

REFERENCES

[1] **Video References:**

1. **Human Pose Estimation - A Complete Tutorial** (YouTube)
   - A detailed guide explaining how human pose estimation works, including the underlying algorithms and how to implement them.
   - Link
2. **Pose Estimation with TensorFlow** (YouTube)
   - A tutorial on how to use TensorFlow for pose estimation, ideal for implementation in real-time applications.
   - Link
3. **Real-Time Pose Estimation with OpenPose** (YouTube)
   - Learn about OpenPose, one of the most popular libraries for human pose estimation in real-time.
   - Link

[2] **Website References:**

1. **OpenPose** (by CMU Perceptual Computing Lab)
   - OpenPose is one of the most widely used real-time multi-person pose detection frameworks.
   - Link
2. **MediaPipe by Google**
   - A framework developed by Google for cross-platform, real-time ML solutions for human pose estimation and more.
   - Link
3. **Deep Learning for Pose Estimation (Research Paper)**:
   - This paper provides an in-depth analysis of deep learning techniques used for human pose estimation.
   - Link