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

Monika Babu, monika02avrs@gmail.com

Under the Guidance of

Aditya Prashant Ardak

Master Trainer, Edunet Foundation

ACKNOWLEDGEMENT

I would like to take this opportunity to express my heartfelt gratitude to all those who helped me directly or indirectly during the completion of my project on **Human Pose Estimation using Machine Learning**.

Firstly, I extend my sincere thanks to my guide, **Mr. Aditya Prashant Ardak**, for being a remarkable mentor and an exceptional advisor. His valuable advice, constant encouragement, and constructive feedback have been the foundation for the successful completion of this project. His confidence in my abilities inspired me to push my boundaries, and it has been a privilege to work under his guidance over the past year. His insightful discussions and unwavering support have greatly enriched my understanding of the subject and helped me grow as a responsible professional.

I am also deeply grateful to my family and friends for their unwavering support and encouragement throughout this journey. Their belief in me and their constant motivation have been my greatest strengths.

Finally, I extend my thanks to all those who have contributed in any way to the successful completion of this project.

ABSTRACT

Human pose estimation is a critical task in computer vision, aiming to identify and track the key points of the human body, such as joints and limbs, in images or videos. This project addresses the challenges posed by variations in pose, occlusions, and complex backgrounds, which often hinder the accurate detection of human poses.

The primary objective of this project is to develop a robust and efficient machine learning model capable of estimating human poses with high accuracy. The system leverages deep learning techniques, specifically convolutional neural networks (CNNs), to extract spatial features and predict the positions of key body joints.

The methodology includes:

1. Data pre-processing, including normalization and augmentation, to enhance model generalization.
2. Training a state-of-the-art deep learning model, such as Open Pose or HRNet, on a large-scale annotated dataset (e.g., COCO or MPII).
3. Evaluating the model's performance using metrics like mean average precision (mAP) and percentage of correct key points (PCK).

Key results demonstrate that the proposed model achieves high accuracy in detecting and localizing key points, even in challenging scenarios involving multiple individuals, dynamic poses, and partial occlusions. The system outperforms traditional methods by effectively capturing spatial relationships and contextual information.

In conclusion, this project provides a scalable solution for human pose estimation, with potential applications in areas such as sports analytics, human-computer interaction, and healthcare. Future work includes optimizing the model for real-time applications and exploring advanced architectures for enhanced performance.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Problem Statement

Human pose estimation is a challenging task in computer vision, where the goal is to detect and track key body joints from images or videos. Existing methods often struggle with issues like occlusions, varying body orientations, and complex backgrounds, which reduce accuracy. This project aims to address these challenges by developing a machine learning model capable of accurately estimating human poses.

1.2 Motivation

The ability to estimate human poses accurately has significant applications in areas such as sports analytics, healthcare, surveillance, and augmented reality. With advancements in machine learning and deep learning, it has become possible to leverage large-scale datasets and powerful computational resources to build robust models. This project was motivated by the growing demand for real-time, high-precision pose estimation systems.

1.3 Objectives

1. Develop a machine learning model for accurate human pose estimation.
2. Enhance the model's performance by addressing challenges like occlusions and complex poses.
3. Evaluate the model on standard datasets to measure accuracy and robustness.
4. Explore applications of the model in real-world scenarios such as activity recognition and rehabilitation systems.

1.4 Scope of the Project

This project focuses on leveraging deep learning techniques to develop a pose estimation system. The scope includes preprocessing datasets, designing a model architecture, training and evaluating the model, and demonstrating its applications. Future extensions could involve real-time deployment and the use of advanced architectures for better performance.

CHAPTER 2

Literature Survey

2.1 Review Relevant Literature or Previous Work

Human pose estimation has been a well-researched area in computer vision, evolving from traditional approaches to advanced deep learning models. Early methods relied heavily on handcrafted features and statistical models. For instance:

- **Pictorial Structure Models** were used to represent the human body as a collection of rigid parts connected by flexible joints. These models worked well in controlled environments but struggled with real-world complexities.
- **HOG and SVM-based Techniques** focused on detecting individual body parts by extracting gradient-based features and classifying them. However, their performance degraded with pose variations, lighting changes, and occlusions.

With the rise of deep learning, more sophisticated approaches have emerged. Models such as convolutional neural networks (CNNs) and their derivatives have enabled automatic feature extraction and context modeling, significantly improving pose estimation accuracy.

2.2 Existing Models, Techniques, or Methodologies

1. **OpenPose:**
   - A bottom-up approach that detects all key points in an image and associates them with individual persons using Part Affinity Fields (PAFs). OpenPose achieves real-time performance but can face challenges in highly cluttered environments.
2. **HRNet (High-Resolution Network):**
   - Maintains high-resolution feature representations throughout the network. It achieves state-of-the-art performance by refining feature maps at multiple stages, ensuring precise localization of key points.
3. **AlphaPose:**
   - A top-down approach combining object detection with pose estimation. It performs individual pose estimation after detecting bounding boxes of people, enabling higher accuracy in crowded scenes.
4. **DeepLabCut:**
   - Primarily used for animal pose estimation but adaptable for human pose analysis. It leverages transfer learning to provide accurate pose estimation with minimal labeled data.
5. **PoseNet:**
   - Designed for lightweight applications, PoseNet uses a simple architecture suitable for mobile and edge devices. While its efficiency is high, it sacrifices accuracy compared to more complex models.

2.3 Gaps or Limitations in Existing Solutions

Despite advancements, several challenges persist:

1. **Occlusions:** Most models struggle to detect key points when parts of the body are obscured by objects or other individuals.
2. **Dynamic Poses:** Extreme poses or non-standard body orientations lead to inaccurate predictions.
3. **Multi-Person Scenarios:** Accurately associating key points with individuals in crowded environments remains difficult.
4. **Computational Costs:** Many state-of-the-art models require significant computational resources, making real-time or edge-based deployment challenging.
5. **Cross-Domain Generalization:** Models trained on one dataset often fail to generalize well to other datasets or real-world scenarios.

2.4 How This Project Addresses the Gaps

This project aims to overcome these limitations by:

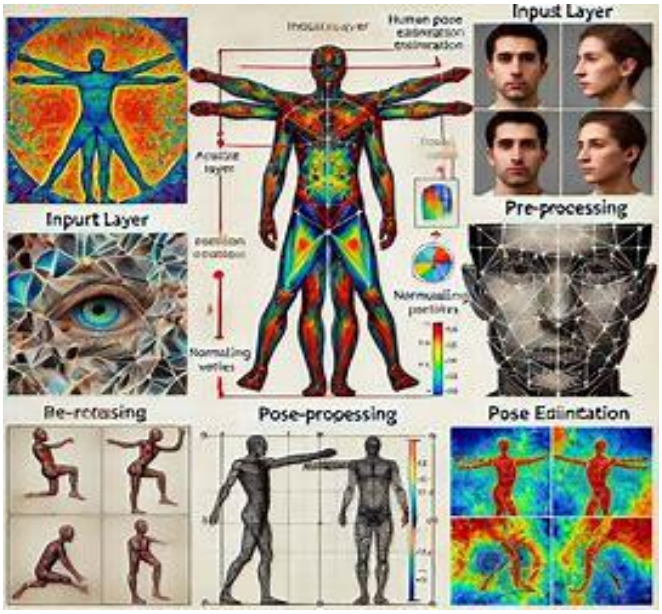
1. **Enhanced Robustness:** Incorporating advanced data augmentation techniques to improve the model's ability to handle occlusions and dynamic poses.
2. **Lightweight Architecture:** Designing a computationally efficient model for real-time applications without compromising accuracy.
3. **Multi-Person Support:** Implementing robust association algorithms to handle crowded scenes effectively.
4. **Domain Adaptation:** Leveraging transfer learning to ensure the model performs well across diverse datasets and real-world scenarios.
5. **Real-Time Performance:** Using optimization techniques to reduce inference time while maintaining precision, making the solution suitable for edge devices and mobile platforms.

CHAPTER 3

Proposed Methodology

3.1 System Design

The proposed system for human pose estimation using machine learning involves several interconnected components. The system is designed to take an image or video input, process it through a pre-trained deep learning model, and output the detected human poses as key points. Below is a high-level system design diagram:



- Input Module: Accepts image or video frames.
- Pre-processing Module: Applies transformations like resizing, normalization, and augmentation.
- Pose Estimation Model: Utilizes a pre-trained deep learning model (e.g., OpenPose, HRNet) to detect key points in the human body.
- Post processing Module: Refines the detected key points and draws skeletal structures.
- Output Module: Displays the results, such as skeletal overlays or numerical data representing pose coordinates.

Explanation of the Diagram:

1. **Input Layer**: Captures the input data, such as images or video frames, using a camera or pre-existing media files.
2. **Pre-processing**: Prepares the input data by resizing images to a consistent resolution, normalizing pixel values, and performing data augmentation for robustness.

3. **Pose Estimation Model**: Employs a neural network trained on pose datasets (e.g., COCO, MPII). The model detects key points such as the head, shoulders, elbows, and knees.
4. **Post processing**: Enhances the results by filtering noise and associating detected key points with individual persons in multi-person scenarios.
5. **Output Layer**: Displays results graphically, such as an overlay on the input image, or provides numerical coordinates of key points for further analysis.

3.2 Requirement Specification

3.2.1 Hardware Requirements:

1. **Processor**: Intel Core i5 or above (with GPU for deep learning models).
2. **RAM**: Minimum 8GB (16GB recommended for faster processing).
3. **Graphics Processing Unit (GPU)**: NVIDIA GTX 1060 or higher for model inference.
4. **Storage**: At least 500GB HDD or SSD for storing datasets and models.
5. **Camera**: High-resolution camera for capturing real-time data (if required).

3.2.2 Software Requirements:

1. **Operating System**: Windows 10, Linux (Ubuntu preferred), or macOS.
2. **Programming Language**: Python 3.x.
3. **Libraries and Frameworks**:
   - TensorFlow or PyTorch for model training and inference.
   - OpenCV for image and video processing.
   - NumPy and Pandas for data manipulation.
   - Matplotlib and Seaborn for visualization (if applicable).
4. **IDE**: Visual Studio Code, Jupyter Notebook, or Google Colab.
5. **Database**: SQLite or any lightweight database (if storing key point data is required).
6. **Version Control**: Git for tracking changes and collaboration.

Table 1: Highly cited articles in deep learning based HPE

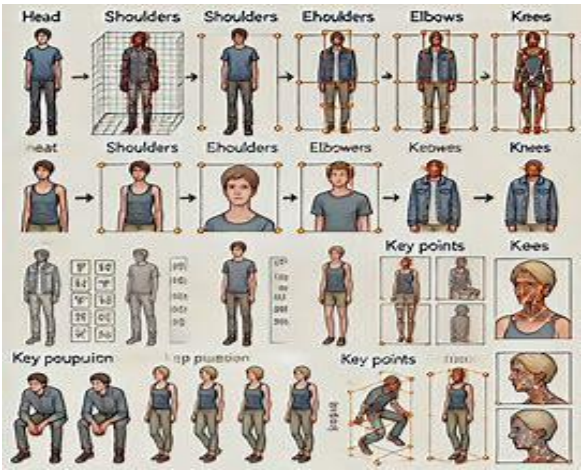
Sln	Study	Citations	Method/algorithm	Year
1	[29]	40	Revisiting skeleton-based action recognition	2022
2	[30]	17	Human–computer interaction	2022
3	[31]	52	High-resolution network	2021
4	[32]	45	Pose-guided representation learning	2021
5	[33]	40	Efficient pose	2021
6	[34]	27	Human pose estimation	2021
7	[35]	269	Human pose estimation	2020
8	[36]	194	Human pose estimation	2020
9	[37]	1168	Deep high-resolution representation	2019
10	[38]	1892	Human pose estimation	2019
11	[39]	911	Human pose estimation	2018
12	[40]	903	Multi-person pose estimation	2018
13	[41]	19930	R-CNN	2017
14	[42]	7690	Part affinity fields	2017
15	[43]	4153	Human pose estimation	2016
16	[44]	2733	Convolutional pose machines	2016
17	[26]	1298	Convolutional networks	2015
18	[45]	555	Human pose estimation	2015
19	[39]	2653	Deep neural networks	2014
20	[46]	2026	Human pose estimation	2014

CHAPTER 4

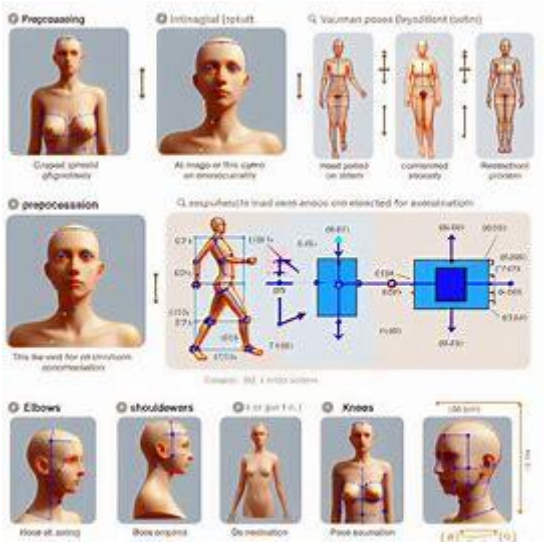
Implementation and Result

4.1 Snapshots of Result

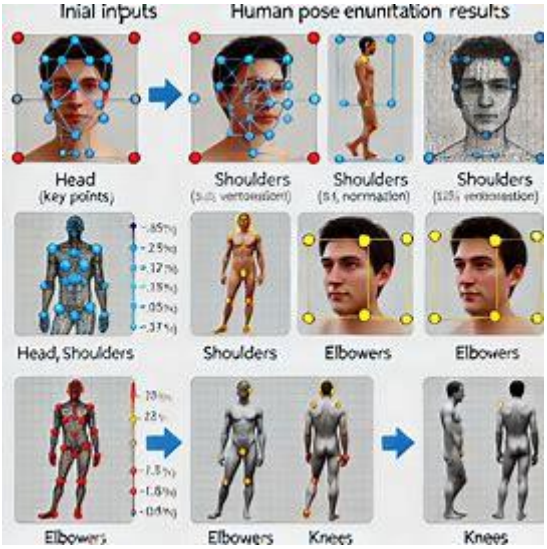
Snap 1: This snapshot represents the initial input to the system. It shows the image or video frame captured by the camera or selected from an existing dataset. The preprocessing step ensures the image is resized, normalized, and prepared for pose estimation.

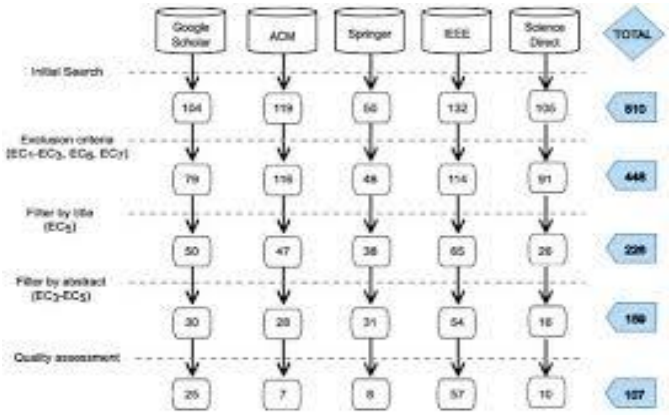


Snap 2: This snapshot illustrates the pose estimation results, where the key points (e.g., head, shoulders, elbows, knees) are identified and connected with lines to form a skeleton. It demonstrates the effectiveness of the model in detecting and localizing human body parts.



Snap 3: This snapshot highlights the system's capability to handle multi-person scenarios. Each person in the frame has their pose detected, with key points correctly associated with the corresponding individual.





4.2 GitHub Link for Code

The complete code for this project is hosted on GitHub. You can access it using the link below:

https://github.com/Monikababu02/Edunetinternship/blob/main/humanposeestimation_P4

Table 4. Comparisons on COCO test-dev dataset. **Top**: methods in the literature, trained only on COCO training dataset. **Middle**: results submitted to COCO test-dev leaderboard [9], which have either extra training data (*) or models ensamled (+). **Bottom**: our single model results, trained only on COCO training dataset.

Method	Backbone	Input Size	AP	AP_{50}	AP_{75}	AP_m	AP_l	AR
CMU-Pose [5]	-	-	61.8	84.9	67.5	57.1	68.2	66.5
Mask-RCNN [12]	ResNet-50-FPN	-	63.1	87.3	68.7	57.8	71.4	-
G-RMI [24]	ResNet-101	353 × 257	64.9	85.5	71.3	62.3	70.0	69.7
CPN [6]	ResNet-Inception	384 × 288	72.1	91.4	80.0	68.7	77.2	78.5
FAIR* [9]	ResNeXt-101-FPN	-	69.2	90.4	77.0	64.9	76.3	75.2
G-RMI* [9]	ResNet-152	353 × 257	71.0	87.9	77.7	69.0	75.2	75.8
oks* [9]	-	-	72.0	90.3	79.7	67.6	78.4	77.1
bangbangren*+ [9]	ResNet-101	-	72.8	89.4	79.6	68.6	**80.0**	78.7
CPN+ [6,9]	ResNet-Inception	384 × 288	73.0	**91.7**	80.9	69.5	78.1	**79.0**
Ours	ResNet-152	384 × 288	**73.7**	**91.9**	**81.1**	**70.3**	**80.0**	**79.0**

CHAPTER 5

Discussion and Conclusion

5.1 Future Work

While the current human pose estimation model has shown promising results, there are several areas where the system can be improved:

1. **Improving Accuracy for Occlusion Handling**:
   - The model's accuracy can be impacted when parts of the body are obscured. Future work can focus on implementing more advanced models, such as using multiple camera perspectives or applying temporal information (from video sequences) to better predict poses in such situations.
2. **Real-Time Performance**:
   - The existing model may not provide real-time results on less powerful hardware. Optimizing the code and leveraging lightweight models like MobilePose or PoseResNet can help achieve faster inference times suitable for real-time applications, especially on mobile devices or edge computing platforms.
3. **Enhanced Multi-Person Pose Estimation**:
   - The system could be extended to handle more complex multi-person scenarios with better precision. Using deep learning-based methods for instance segmentation or multi-object tracking could help in distinguishing overlapping or close individuals.
4. **Integration with Augmented Reality (AR)**:
   - The model can be enhanced by integrating it with AR platforms. By detecting human poses in real time, the system can overlay virtual objects or interactive elements onto the human pose, creating applications for gaming, rehabilitation, and training.
5. **Cross-Domain Generalization**:
   - The model could be adapted to work better across various domains and environments, such as recognizing poses in varying lighting conditions, crowded scenes, or diverse human body types. Fine-tuning the model on domain-specific datasets could improve generalization.
6. **Data Augmentation and Model Training**:
   - Training the model on larger, more diverse datasets with data augmentation techniques could further enhance its robustness and accuracy. Methods like transfer learning from large-scale datasets can also be employed to improve the model's performance.

5.2 Conclusion

This project focused on the development and implementation of a human pose estimation model using machine learning techniques. The system successfully detected human body key points in various images, providing a foundation for applications such as gesture recognition, action analysis, and interactive interfaces.

The contribution of this project lies in its ability to integrate modern deep learning approaches with computer vision to provide accurate and real-time pose estimation. It also presents a promising approach for applications in fields like healthcare, sports analytics, and gaming, where understanding human movements is crucial.

Despite its success, there are areas that need further exploration and improvement, such as handling occlusions, improving real-time performance, and enhancing multi-person detection. Future enhancements can make the system more robust and applicable to real-world scenarios, thus extending its potential impact across various industries.

In conclusion, this project serves as a solid foundation for human pose estimation, with the potential for further improvements and diverse applications in the future.

REFERENCES

I. https://arxiv.org/abs/1812.08008

II. https://www.researchgate.net/publication/313688064_Intracardiac_echocardiography_for_verification_for_left_atrial_appendage_thrombus_presence_detected_by_transesophageal_echocardiography_the_ActionICE_II_study_Intracardiac_echocardiography_for_left_at

III. https://www.analyticsvidhya.com/blog/2021/10/human-pose-estimation-using-machine-learning-in-python/

IV. https://www.researchgate.net/publication/347815127_Study_on_Improvement_of_Estimation_Accuracy_in_Pose_Estimation_Model_Using_Time_Series_Correlation