A Minor Project Synopsis

on

Human pose detection like Snapchat using Deep Learning

Submitted to Manipal University Jaipur

towards the partial fulfillment for the award of the degree of

Bachelor of Technology

In Information Technology

By

Arunim Sureka

219303090

IT 6F

Sarthak Garg

219302116

IT 6B

Under the Guidance of

Ms. Smaranika Mohapatra

MANIPAL UNIVERSITY
JAIPUR

Department of Information Technology

School of Information Technology

Manipal University Jaipur

2023-2024

Synopsis

1. Introduction

The advent of deep learning technologies has significantly transformed the landscape of computer vision, enabling remarkable advancements in image recognition, and understanding. One captivating application that has gained substantial attention is human pose detection, akin to the interactive filters popularized by platforms like Snapchat. This minor project embarks on the exploration of human pose detection using deep learning, a cutting-edge field that holds immense potential in various domains, from augmented reality to healthcare.

The essence of this project lies in harnessing the power of deep neural networks to accurately detect and analyze human poses from images or videos. The inspiration drawn from Snapchat's interactive filters underscores the project's goal to create an engaging and immersive user experience. By leveraging deep learning techniques, we aim to develop a robust system that can accurately identify and track human body poses in real-time, contributing to the evolution of interactive applications and systems.

This project delves into the intricacies of deep learning architectures, exploring key concepts such as convolutional neural networks (CNNs) and pose estimation algorithms. As we navigate through the project's development, we anticipate not only gaining valuable insights into the technical intricacies of human pose detection but also contributing to the broader narrative of how deep learning can enhance interactive experiences in the digital realm.

2. Motivation

The motivation behind Pose detection like snapchat using deep learning is the wide variety of applications in the industry offering solutions to diverse needs ranging from entertainment and gaming to healthcare, education, and security. The ability to accurately detect and track poses in real-time opens possibilities for creating more interactive and responsive systems in various industries.

The project was captivating for as it gave us an opportunity to explore AI and Deep Learning in a meaningful way and build a project that would help us to apply our knowledge of the same, due to the ever-increasing demand of AI and related technologies we felt it is a crucial skill to acquire and a great opportunity to explore the field of AI.

The project contributes to the field of computer vision and Augmented reality by providing an efficient and accurate pose detection system. The developed project can be applied to various domains including fitness tracking and entertainment.

3. Project Objectives

3.1 Pose Detection:

Implement a robust pose detection model capable of accurately estimating the key points of a person's body including joints and limbs.

3.2 Real-Time Tracking:

Develop algorithms for real-time pose tracking to ensure smooth and continuous detection as the subject moves within the camera frame.

3.3 Data Collection and Processing:

Curate a dataset of diverse poses and perform preprocessing to enhance model generalization.

3.4 Deep Learning model:

Design and train a deep learning model, potentially based on state-of-the-art architectures such as OpenPose and PoseNet to predict pose from the input images.

4. Methodology/ Planning of work:

4.1 Data Collection:

Gather a diverse dataset of annotated images depicting various poses to train and evaluate the model.

4.2 Data Preprocessing:

Clean and preprocess the dataset, performing data augmentation to enhance model robustness.

4.3 Model Architecture:

Choose an appropriate deep learning architecture for pose detection considering factors such as accuracy, speed, and suitability for real-time applications.

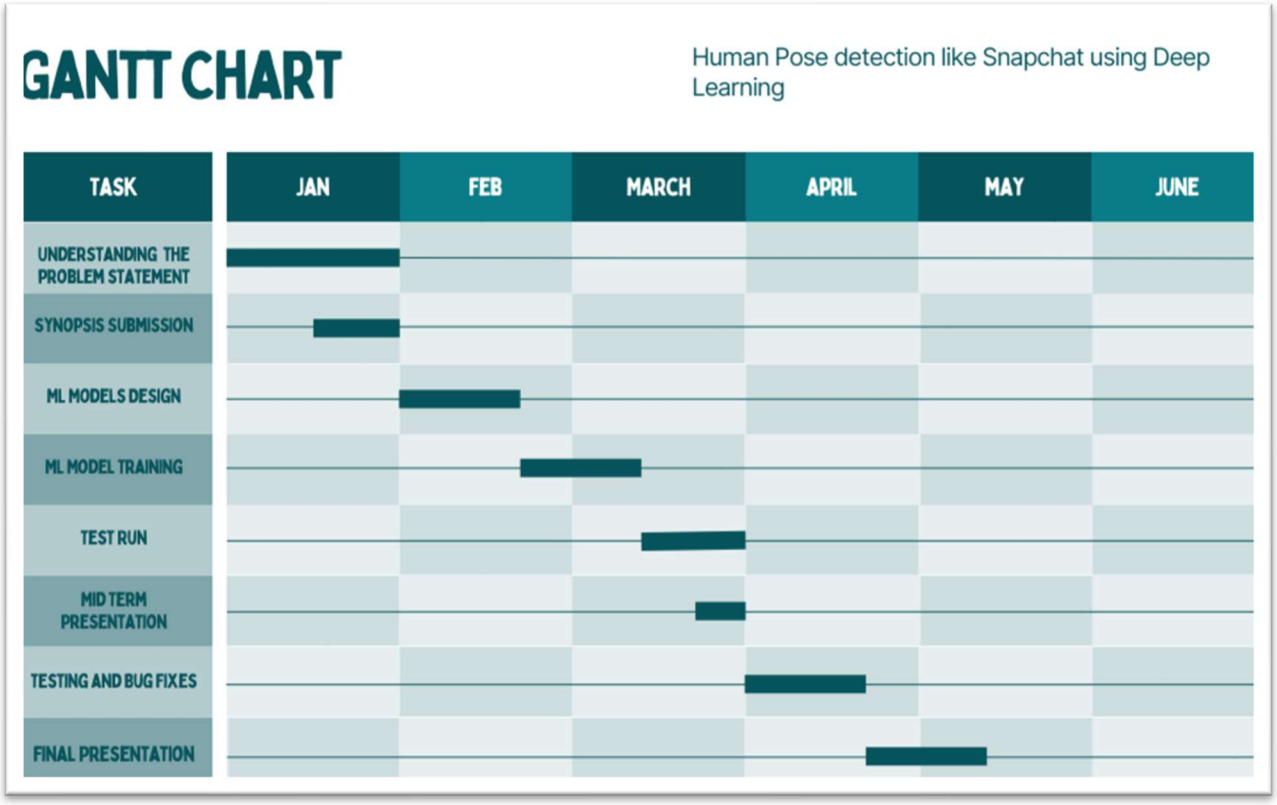
4.4 Training:

Train the model on the prepared dataset, fine – tuning parameters for optimal performance.

4.5 Real-time Implementation:

Implementing the trained model in real-time, ensuring efficient pose detection and tracking.

4.6 Tentative Project Timeline:



5. Facilities required for proposed work:

5.1 Software Requirements Platform:

5.1.1. Operating System: Windows OS
5.1.2. Programming Language: Python

5.2 Hardware Requirements:

5.2.1 Processor: Intel Core i5 or Higher
5.2.2 Disk Space: 80 GB (Minimum)
5.2.3 RAM: 8GB or Higher

5.3 Performance Requirements:

5.3.1 The system should be capable of producing pose detection results efficiently.

5.3.2 The application should run smoothly on systems with 8 GB or higher RAM.

5.3.3 The processing time for detecting and tracking poses should be optimized for real-time performance, especially during peak loads.

5.3.4 The system should strive to be available 100% of the time. In case of fatal errors, clear, and understandable feedback should be provided to the user.

5.4 Safety and Security Requirements:

5.4.1 The system should be designed with modular architecture to facilitate easy error, detection, and fixes.

5.4.2 Ensure that the system adheres to privacy and data security standards when handling user data.

5.5 Software Quality Attributes

5.5.1. Reliability:
- The system should reliably detect and track poses in various scenarios.
- Successful pose detection should be indicated to the user, ensuring confidence in the system's accuracy.

5.5.2 Maintainability:
- The system will be developed following standard software development conventions.
- Code should be well-documented and organized, facilitating easy review and redesign if necessary.

5.5.3. Portability:
- The software should be portable to various systems that meet the specified hardware requirements.
- Compatibility should be maintained across different Python environments and versions.

5.6Additional Requirements

5.6.1. Deep Learning Framework:
- TensorFlow, PyTorch, and Keras

5.6.2. Pose Detection Model:
OpenPose and PoseNet

5.6.3 Python Libraries:
NumPy, OpenCV, and Matplotlib

6. Bibliography/References:

1. Deep Learning for Computer Vision: A Brief Review
Author: Rajalingappaa Shanmugamani
Year: 2018
Publisher: Packt Publishing
ISBN-13: 978-1789341076

2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems
Author: Aurélien Géron
Year: 2019
Publisher: O'Reilly Media
ISBN-13: 978-1492032649

3. OpenPose: Real-time multi-person keypoint detection library
Authors: Cao, Z., Simon, T., Wei, S. E., & Sheikh, Y.
Year: 2019
Published in: IEEE Transactions on Pattern Analysis and Machine Intelligence
DOI: [10.1109/TPAMI.2018.2844175](https://doi.org/10.1109/TPAMI.2018.2844175)

4. Multi-Person 2D Pose Estimation using Part Affinity Fields
Authors: Alejandro Newell, Kaiyu Yang, and Jia Deng
Year: 2016
Published in: arXiv preprint arXiv:1611.08050

5. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization
Authors: A. Kendall, M. Grimes, and R. Cipolla
Year: 2015
Published in: Proceedings of the IEEE International Conference on Computer Vision (ICCV)

6. Python Machine Learning
Author: Sebastian Raschka
Year: 2015
Publisher: Packt Publishing
ISBN-13: 978-1783555130