AI Human Body Pose Estimation through Python- [OpenCV MediaPipe]

Introduction

This presentation provides a comprehensive overview of **human body pose estimation** using *Python* and *OpenCV*. We will explore the latest advancements in the field and discuss the key components of the project.

Understanding Human Pose Estimation

Human is the process of detecting and tracking key points on the human body to understand its position and movement. This slide will delve into the significance and applications of this technology.

What is Pose Estimation?

Pose estimation is a fundamental task in computer vision and artificial intelligence (AI) that involves detecting and tracking the position and orientation of human body parts in images or videos. Human pose estimation and tracking is a computer vision task that includes detecting, associating, and tracking semantic key points. Examples of semantic key points are “right shoulders,” “left knees,”.

Today, the most powerful image processing models are based on convolutional neural networks (CNNs). Hence, state-of-the-art methods are typically based on designing the CNN architecture tailored particularly for object or human pose

2d Pose estimation

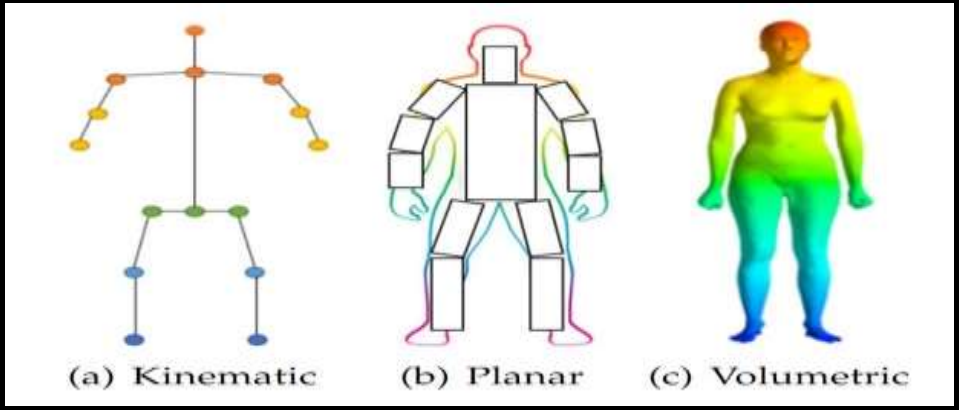
2D human pose estimation is used to estimate the 2D position or spatial location of human body keypoints from visuals such as images and videos. Traditional 2D human pose estimation methods use different hand-crafted feature extraction techniques for the individual body parts. Early computer vision works described the human body as a stick figure to obtain global pose structures. However, modern deep learning based approaches have achieved major breakthroughs by improving the performance significantly for both single-person and multi-person pose estimation. Some popular 2D human pose estimation methods include OpenPose, CPN, AlphaPose, and HRNet (we will cover them and others later in this article).

3d Pose estimation

3D Human Pose Estimation is used to predict the locations of body joints in 3D space. Besides the 3D pose, some methods also recover 3D human mesh from images or videos. This field has attracted much interest in recent years since it is used to provide extensive 3D structure information related to the human body. It can be applied to various applications, such as 3D animation industries, virtual or augmented reality, and 3D action prediction. 3D human pose estimation can be performed on monocular images or videos (normal camera feeds). Using multiple viewpoints or additional sensors (IMU or LiDAR), 3D pose estimation can be applied with information fusion techniques, which is a very challenging task. While 2D human datasets can be easily obtained, collecting accurate 3D pose image annotation is time-consuming, and manual labeling is not practical and expensive. Therefore, although 3D pose tracking has made significant advancements in recent years, especially due to the progress made in 2D human pose estimation, there are still several challenges to overcome: Model generalization, robustness to occlusion, and computation efficiency

Types of 3d model

- Kinematic Model
- Planar Model
- Volumetric model



Python OpenCV

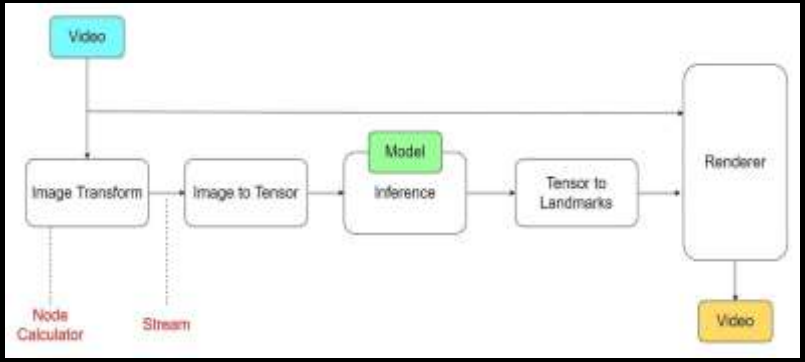
OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real-time operation which is very important in today's systems. By using it, one can process images and videos to identify objects, faces, or even handwriting of a human. When it integrated with various libraries, such as NumPy, python is capable of processing the OpenCV array structure for analysis. To Identify image pattern and its various features we use vector space and perform mathematical operations on these features.

The first OpenCV version was 1.0. OpenCV is released under a BSD license and hence it's free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. When OpenCV was designed the main focus was real-time applications for computational efficiency. All things are written in optimized C/C++ to take advantage of multi-core processing.

- face recognition
- Automated inspection and surveillance
- number of people – count (foot traffic in a mall, etc)
- Vehicle counting on highways along with their speeds
- Street view image stitching
- Video/image search and retrieval
- Robot and driver-less car navigation and control
- object recognition
- Medical image analysis

MediaPipe

MediaPipe is a Framework for building machine learning pipelines for processing time-series data like video, audio, etc. This cross-platform Framework works on Desktop/Server, Android, iOS, and embedded devices like Raspberry Pi and Jetson Nano.




MP Hands

MediaPipe Solutions

Solutions are open-source pre-built examples based on a specific pre-trained TensorFlow or TFLite model. You can check Solution specific models here. MediaPipe Solutions are built on top of the Framework. Currently, it provides sixteen solutions, as listed below.

- Face Detection
- Face Mesh
- Iris
- Hands
- Pose
- Holistic

Understanding Facial Recognition

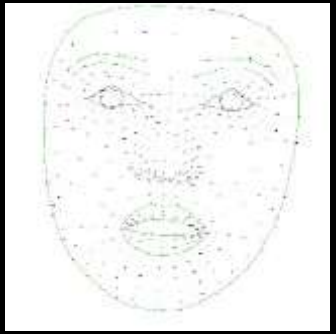
Facial recognition is a technology that identifies or verifies a person from a digital image or a video frame. It involves detecting and recognizing facial features such as eyes, nose, and mouth.

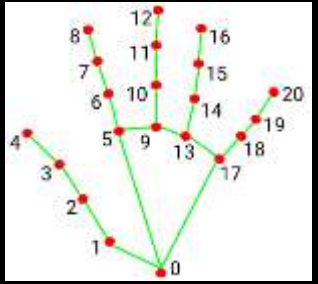
Mediapipe Holistic Model

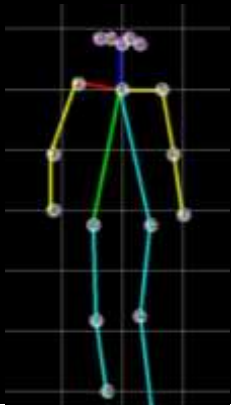
Mediapipe holistice model is used to detect the body posture by using default camera

This will based on three type:

WITH THE HELP OF DEFAULT CAMERA, DETECT FACEMESH LANDMARKS
WITH THE HELP OF DEFAULT CAMERA, DETECT HAND LANDMARKS
WITH THE HELP OF DEFAULT CAMERA, DETECT POSE LANDMARKS





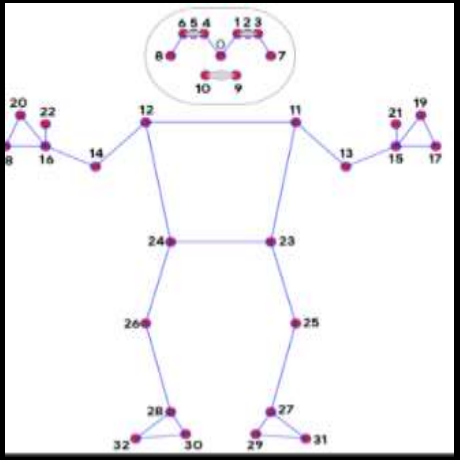


GYM TRACKER

Curl counter

Calculate angle

Focus on 13 && 14



Real-World Applications

Human pose estimation has diverse applications, including sports analytics, healthcare, virtual reality, and human-computer interaction. This slide will showcase real-world examples of these applications.

Conclusion

In conclusion, this presentation has provided a detailed overview of advancing human pose estimation through Python-OpenCV and Mediapipe. The potential impact of this technology and the future opportunities for innovation have been highlighted.

Thanks!

Developer details:

- Shirish Dwivedi
- Prince Raghav