POSE Estimation

PRESENTATION

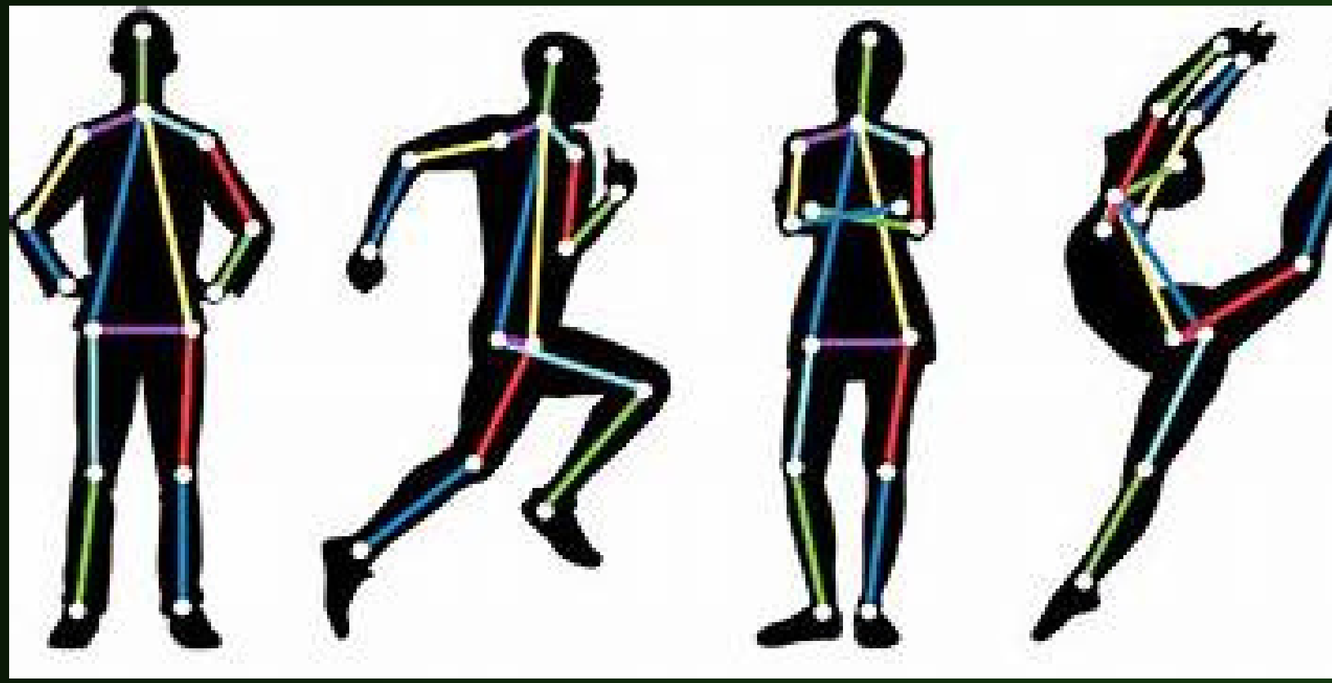
Guided By:SAMBIT SUBHASISH SAHU

Presented By:
LIPA PRUSTY
S S SIMRAN BEHERA

INTRODUCTION

The field of computer vision has witnessed significant dvancements in recent years, with pose estimation emerging s a pivotal area of research and application. Pose estimation volves the detection and localization of human body joints or ndmarks in images or video sequences. This technology has wide array of applications, ranging from human-computer interaction and augmented reality to sports analysis, healthcare, and surveillance.

This introduction focuses on pose estimation using readily vailable and powerful tools: OpenCV, MediaPipe, and Python.



Aim: The primary aim is to empower users to develop real-time, robust, and accurate human pose estimation applications. This involves:

- Understanding the theoretical underpinnings: Gaining a conceptual grasp of how pose estimation models work, including the detection of keypoints and the skeletal connections between them.
- Mastering practical implementation: Learning to integrate and utilize the functionalities offered by OpenCV, MediaPipe, and Python to process visual input, infer pose, and visualize the results.

: OBJECTIVE :

The objective of this project is to develop a real-time human pose estimation system using MediaPipe, Python, and OpenCV that accurately detects and tracks body landmarks. The system aims to support applications such as fitness monitoring, gesture-based control, and health posture analysis by providing an efficient and lightweight solution for human pose detection.

TOOLS & TECHNOLOGY:

Tools and Technologies Used:

1. Programming Language:
   - Python
2. Libraries & Frameworks:
   - MediaPipe – for real-time pose estimation
   - OpenCV – for image and video processing
3. Hardware:
   - Standard PC or Laptop with webcam
4. Software Environment:
   - Jupyter Notebook

: WORKFLOW :

CAPTURE INPUT:
- Read video from webcam using OpenCV

INITIALIZE POSE MODEL:
- Set up MediaPipe Pose with confidence thresholds.

PROCESS EACH FRAME:
- Convert BGR to RGB
- Detect body landmarks using MediaPipe

DRAW LANDMARKS:
- Overlay keypoints and skeleton on the frame

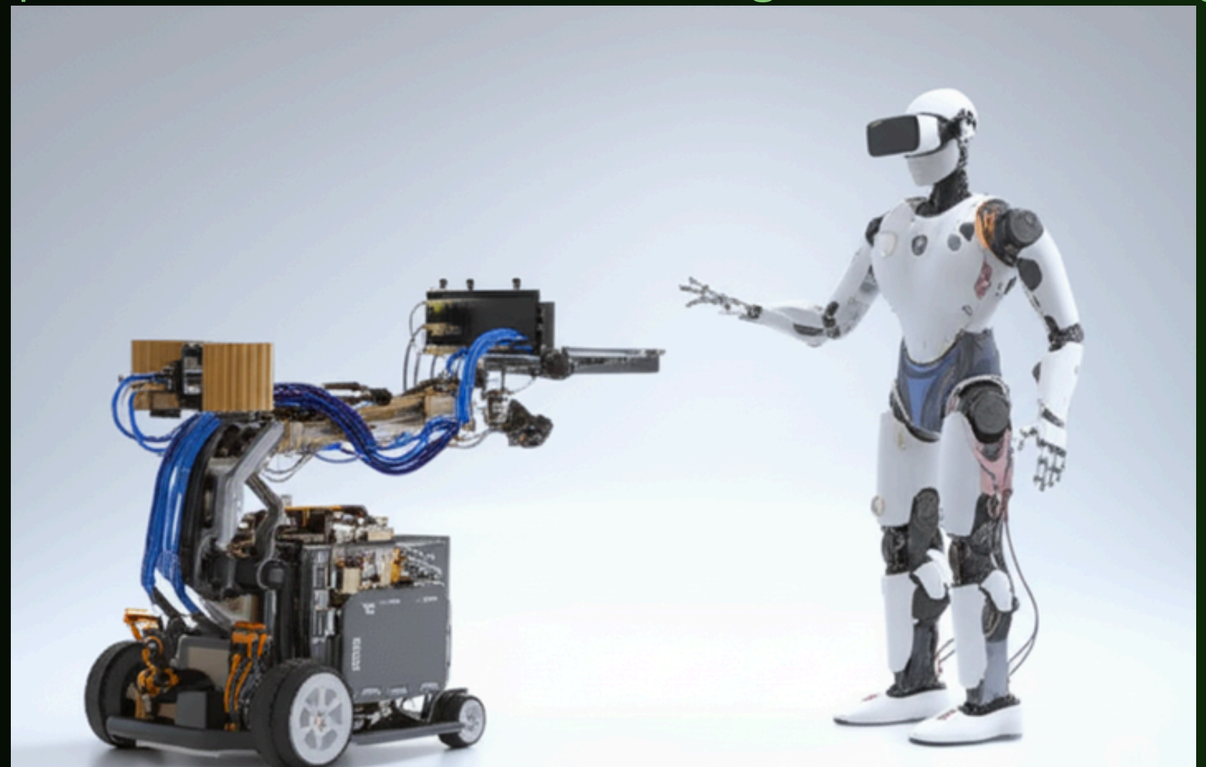
DISPLAY OUTPUT:
- Show the result in real - time using OpenCV window.

RELEASE RESOURCES:
- Close video stream and OpenCV.

: FUTURE SCOPE :

- Smarter Fitness & Healthcare: Real-time feedback rehabilitation, and posture correction

- Enhanced Human-Computer Interaction: More natural gesture control for gaming, and smart devices.



- Pose estimation is helping sports become smarter and safer like: Technique Improvement, Injury Prevention,Performance Analysis

- Improved Surveillance & Security: Advanced anomaly detection and behavior analysis

: CONCLUSION :

This project successfully demonstrates real-time human pose estimation using Python, OpenCV, and MediaPipe. By accurately detecting 33 body landmarks, the system enables applications such as fitness tracking, gesture recognition, and health monitoring. The use of lightweight and efficient tools makes it suitable for real-world deployment on basic hardware, proving its potential for various AI-powered human interaction systems

THANK YOU