REAL TIME SQUAT ANALYSIS USING MEDIAPIPE

ASP 484 2.0 Image Processing and Embedded Computer Vision

Name - W.L.S. Pasan

INTRODUCTION TO SQUAT ANALYSIS

What is Squat Analysis?

- Squat analysis is the process of examining the posture and movement during squats to improve form and reduce the risk of injury. Correct squat form ensures that the knees do not extend beyond the toes, the back remains straight, and the hips drop sufficiently to work the correct muscles.

- Analyzing squats allows fitness professionals or individuals to track their progress and make necessary adjustments to their form.

INTRODUCTION TO SQUAT ANALYSIS

Why Use MediaPipe?

- MediaPipe is a framework developed by Google that provides pre-built solutions for computer vision tasks, such as pose estimation. It can track body movements in real-time, making it ideal for applications like fitness analysis.

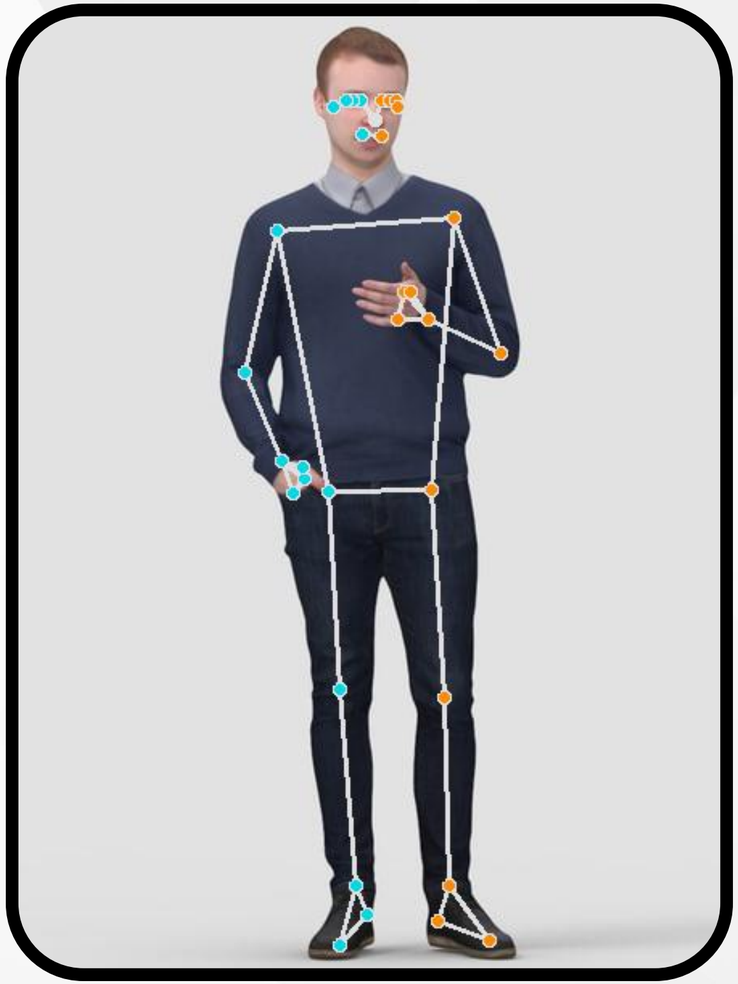
- It offers a light-weight and high-performance solution for real-time body landmark tracking and angle calculations, making it perfect for applications like squat analysis.

OVERVIEW OF MEDIAPIPE

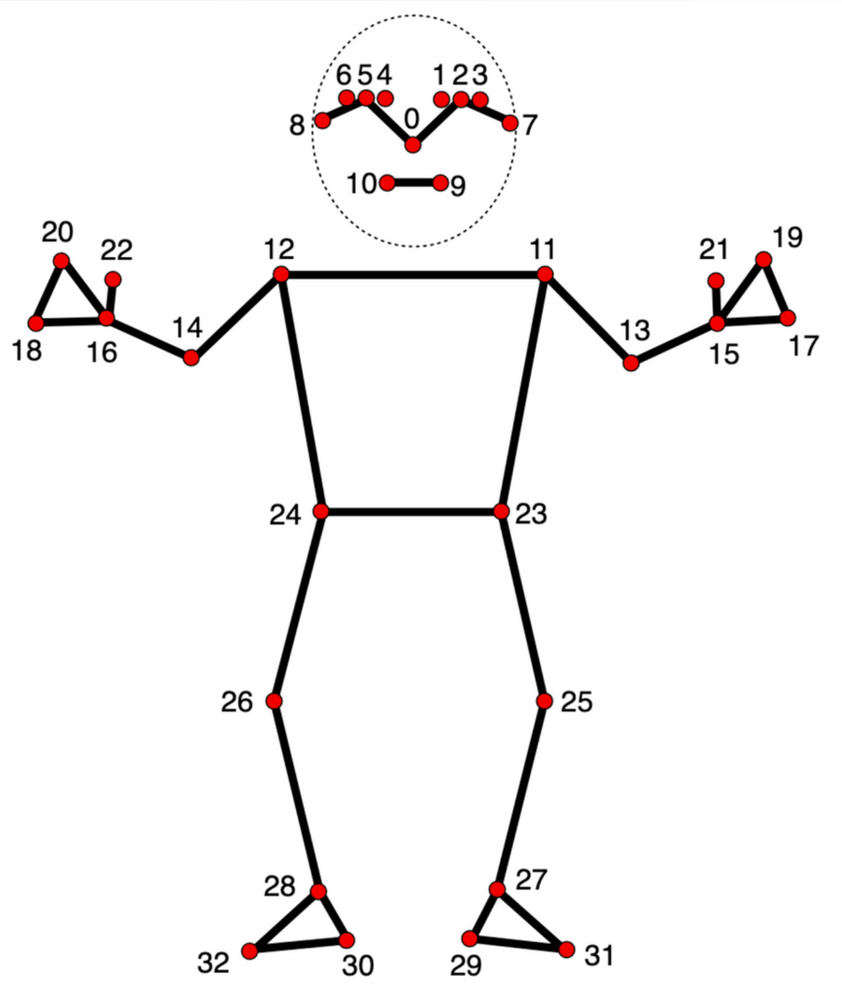
What is MediaPipe?

- MediaPipe is an open-source framework for building cross-platform multimedia processing pipelines. It has pre-built solutions for tasks like face detection, hand tracking, pose estimation, and object detection.

- In this case, we are using Pose Estimation, which uses a camera feed to detect body landmarks (like the hips, knees, and ankles), essential for understanding movement and posture.



OVERVIEW OF MEDIAPIPE

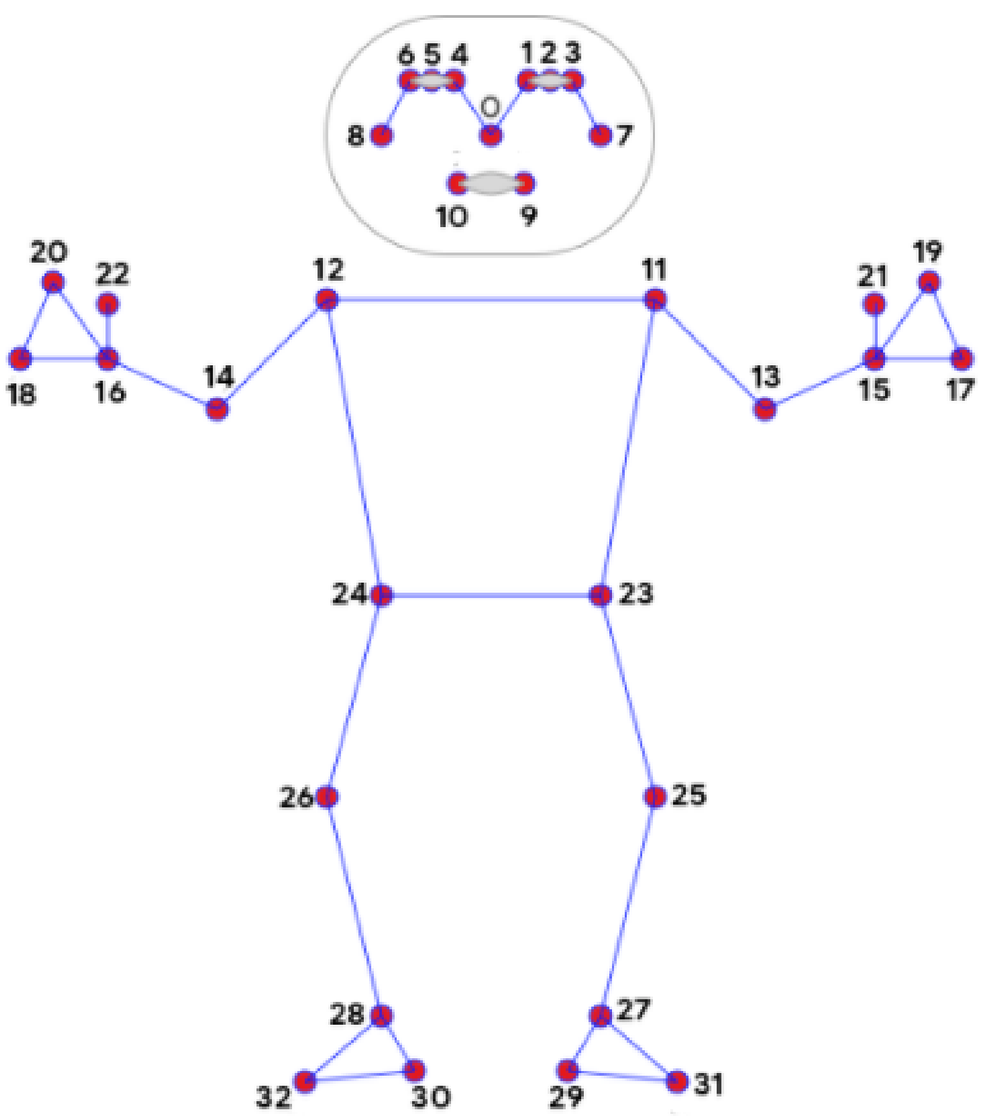


Why MediaPipe for Squats?

- MediaPipe Pose Estimation provides real-time tracking of body landmarks (such as shoulders, elbows, knees, and ankles) without requiring complex hardware setups or external sensors.

- The pose model identifies and tracks up to 33 key body points, allowing the program to analyze the angles of specific joints like the knees and hips, which is crucial for proper squat assessment.

OVERVIEW OF MEDIAPIPE

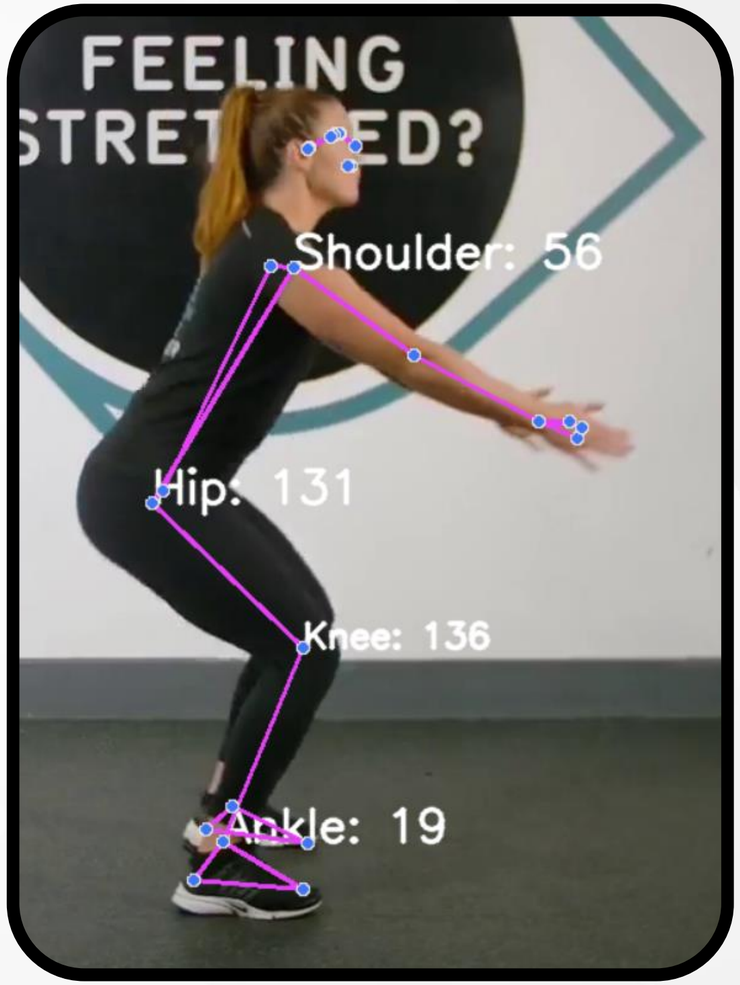


0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

KEY CONCEPTS OF SQUAT ANALYSIS

Body Landmarks Used:

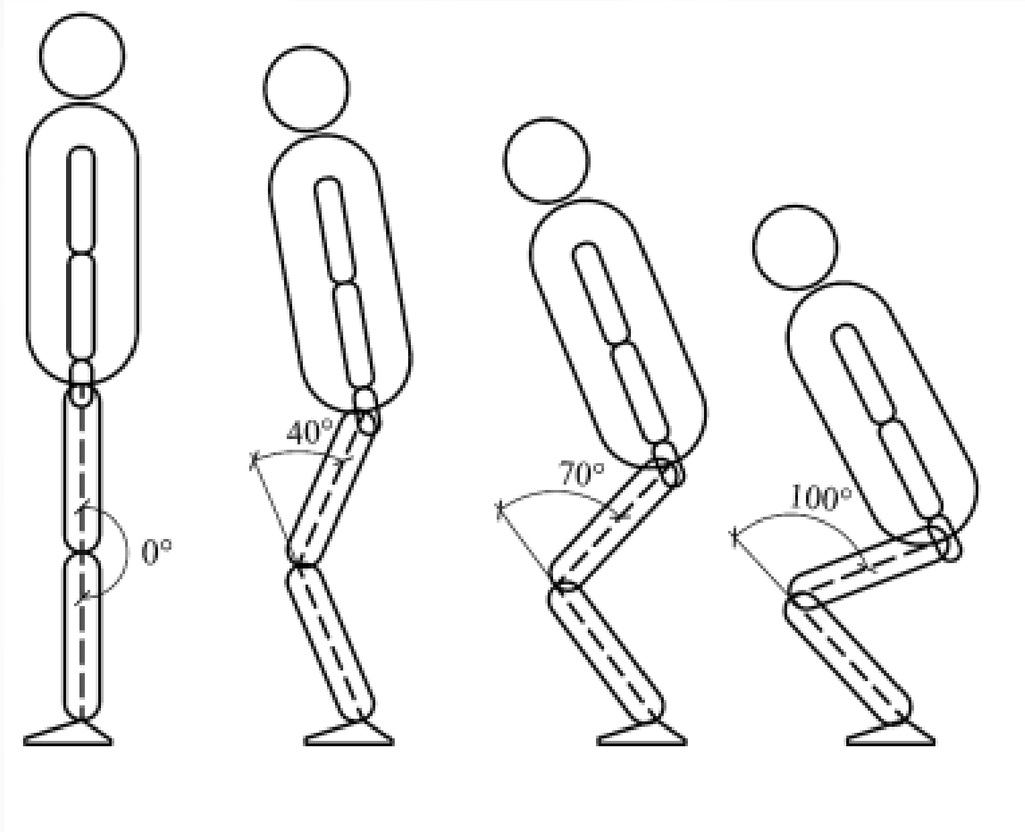
- **Shoulder:** This landmark tracks the position of the left hip.
- **Hip:** This landmark tracks the position of the left hip.
- **Knee:** This is a critical landmark for assessing squat depth. It helps calculate the angle between the hip, knee, and ankle.
- **Ankle:** Along with the knee and hip, this landmark helps calculate the angle that determines whether the squat depth is adequate.



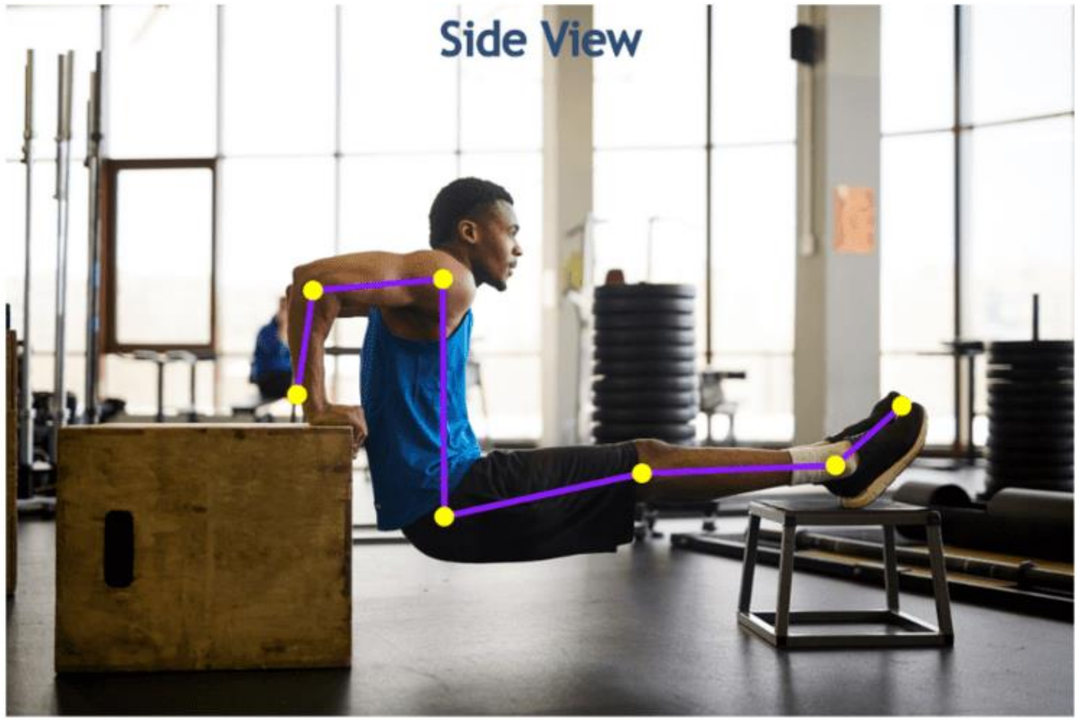
KEY CONCEPTS OF SQUAT ANALYSIS

Angles Measured:

- **Knee Angle:** This is the angle formed by the left hip, left knee, and left ankle during the squat. The knee angle provides insights into the squat depth. A larger angle (greater than 160 degrees) suggests a squat that’s not deep enough, while an angle smaller than 90 degrees means the squat is too deep, which can lead to injury if not performed correctly.



INTUITION OF FRONTAL AND SIDE VIEW FOR POSTURE ANALYSIS



When designing an application to analyze fitness exercises (Squat), it is essential to consider the camera's view of the person.

IMPLEMENTATION STEPS

- **Step 1: Import Required Libraries.**

Libraries like cv2 (OpenCV), mediapipe, and numpy are necessary for capturing webcam input, detecting body landmarks, and performing angle calculations.

- **Step 2: Initialize MediaPipe Pose Solution.**

The mediapipe.solutions.pose class is initialized to enable pose tracking in real-time.

- **Step 3: Capture Webcam Input (or Video Source).**

Use OpenCV's cv2.VideoCapture(0) function to capture video from your webcam.

- **Step 4: Detect Landmarks and Calculate Angles.**

Once the frame is captured, the landmarks are detected using the pose.process() method. Then, the key landmarks (hip, knee, ankle) are used to calculate the angle.

- **Step 5: Provide Feedback Based on Squat Depth.**

Based on the calculated knee angle, the program provides feedback: "Too High," "Too Low," or "Good." This feedback helps the user correct their squat form in real-time.

CODE EXPLANATION

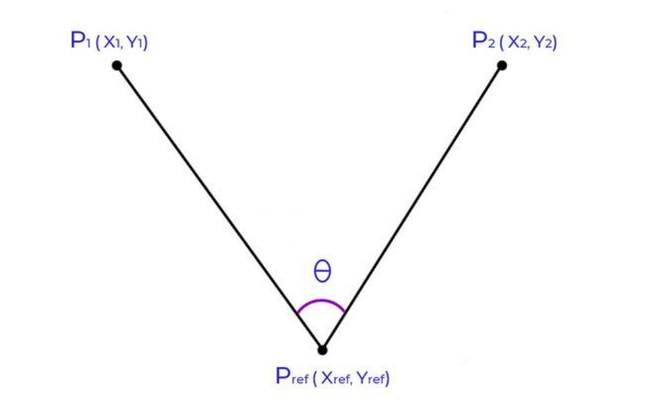
- **Imports:**

1. *cv2* for video capture and image processing.
2. *mediapipe* for pose estimation.
3. *numpy* for handling mathematical operations.
4. *csv, time* for handling csv file and time

- **Function:**

1. *calculate angle*

This function calculates the angle between three points (hip, knee, and ankle). It uses the arctangent of the difference in y and x coordinates between the points to calculate the angle in radians, which is then converted to degrees.



$$\theta = \arccos \frac{\overrightarrow{P_{1ref}} \cdot \overrightarrow{P_{2ref}}}{|\overrightarrow{P_{1ref}}| \cdot |\overrightarrow{P_{2ref}}|}$$

CODE EXPLANATION

- **Pose Estimation Process:**

1. The *pose.process()* method from MediaPipe detects the body landmarks.
2. The key landmarks (left hip, knee, ankle) are identified from the pose landmarks and used to compute the knee angle.

- **Real-Time Feedback:**

1. The angle is displayed on the screen, and text feedback is provided to guide the user on whether their squat depth is correct.

CODE EXPLANATION

```python
import cv2
import mediapipe as mp
import numpy as np
import csv
import time

# Initialize MediaPipe Pose
mp_pose = mp.solutions.pose
pose = mp_pose.Pose()
mp_drawing = mp.solutions.drawing_utils

# Function to calculate the angle between three points
def calculate_angle(a, b, c):
    a = np.array(a)  # First point
    b = np.array(b)  # Second point (vertex)
    c = np.array(c)  # Third point

    radians = np.arctan2(c[1] - b[1], c[0] - b[0]) - np.arctan2(a[1] - b[1], a[0] - b[0])
    angle = np.abs(radians * 180.0 / np.pi)

    if angle > 180.0:
        angle = 360 - angle

    return angle

## Thresholds Value
# Thresholds
```

DEBUG CONSOLE TERMINAL SPELL CHECKER 18

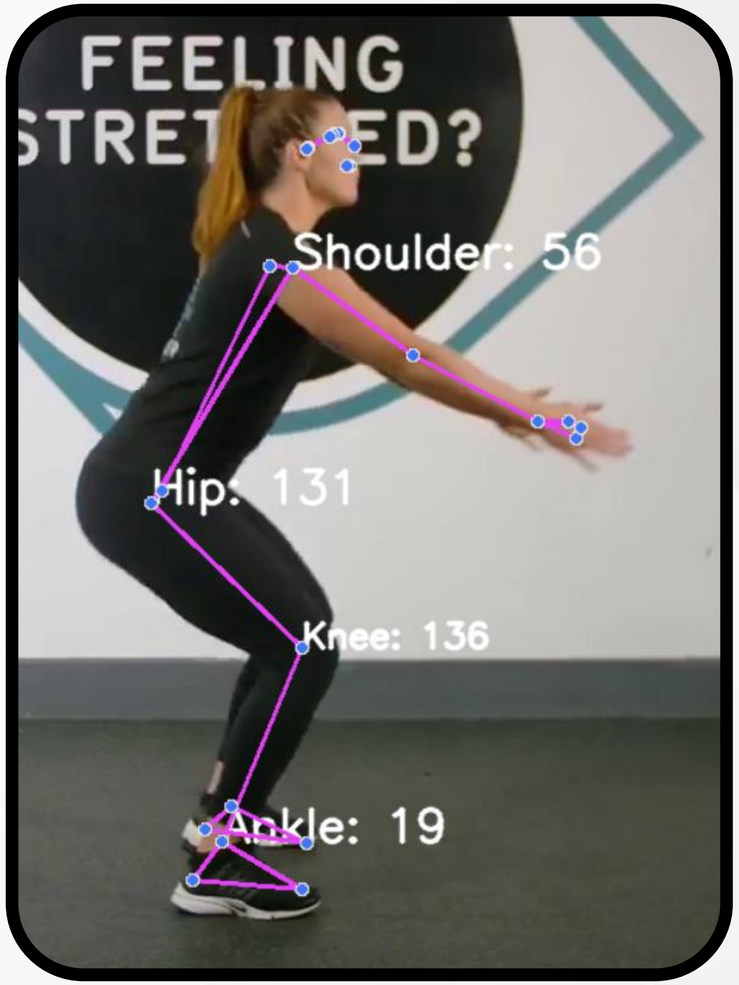
OUTPUT DEMONSTRATION

Real-time Feedback:

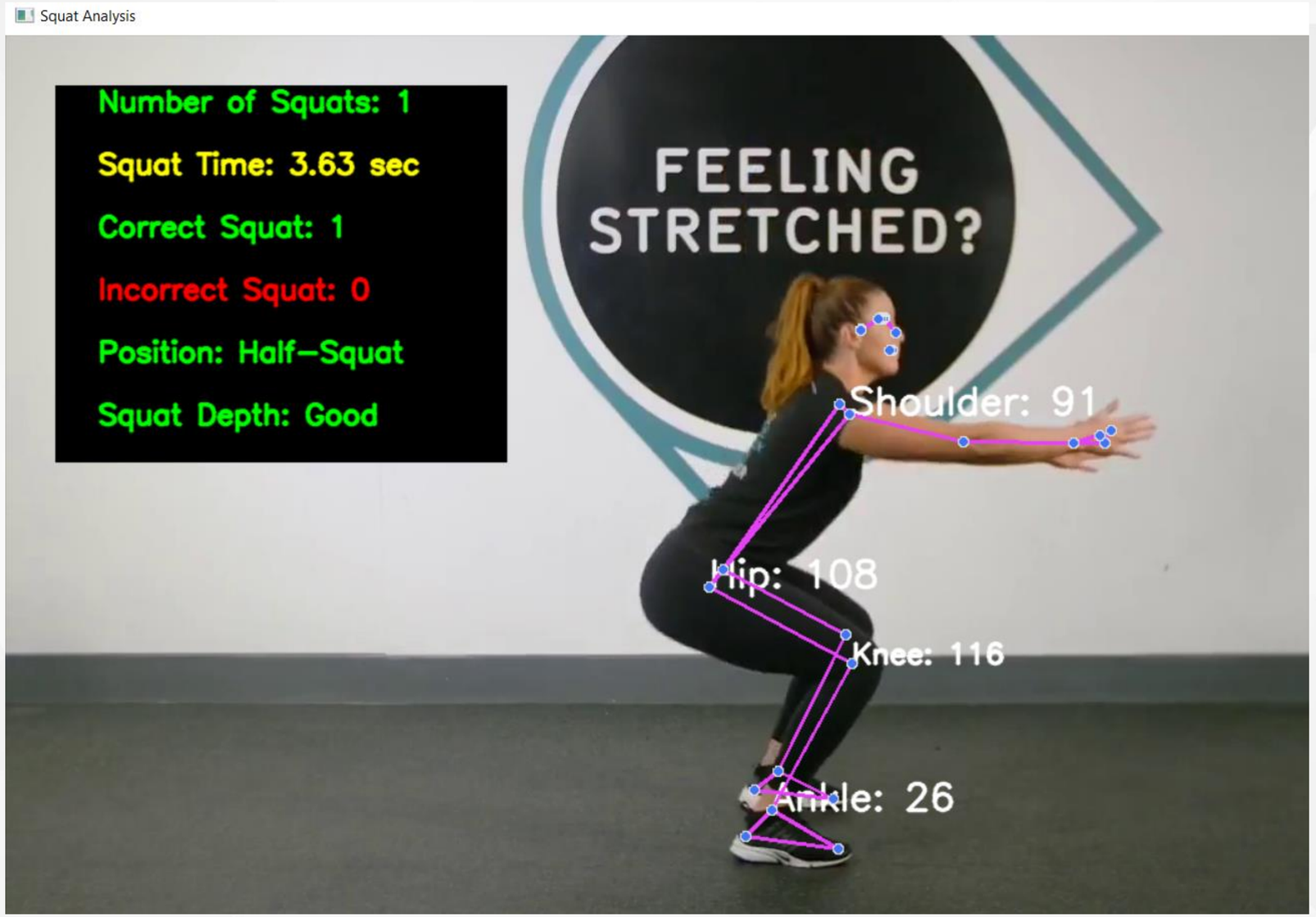
- In the webcam feed, the knee angle is displayed, and the user receives feedback about their squat form. If the angle is too large or too small, they are advised to adjust their depth.

Visual Aid:

- MediaPipe’s pose landmarks are drawn on the webcam feed, allowing the user to visualize their body posture in real-time. This feedback can help them correct their form as they perform the squat.



OUTPUT DEMONSTRATION



OUTPUT DEMONSTRATION

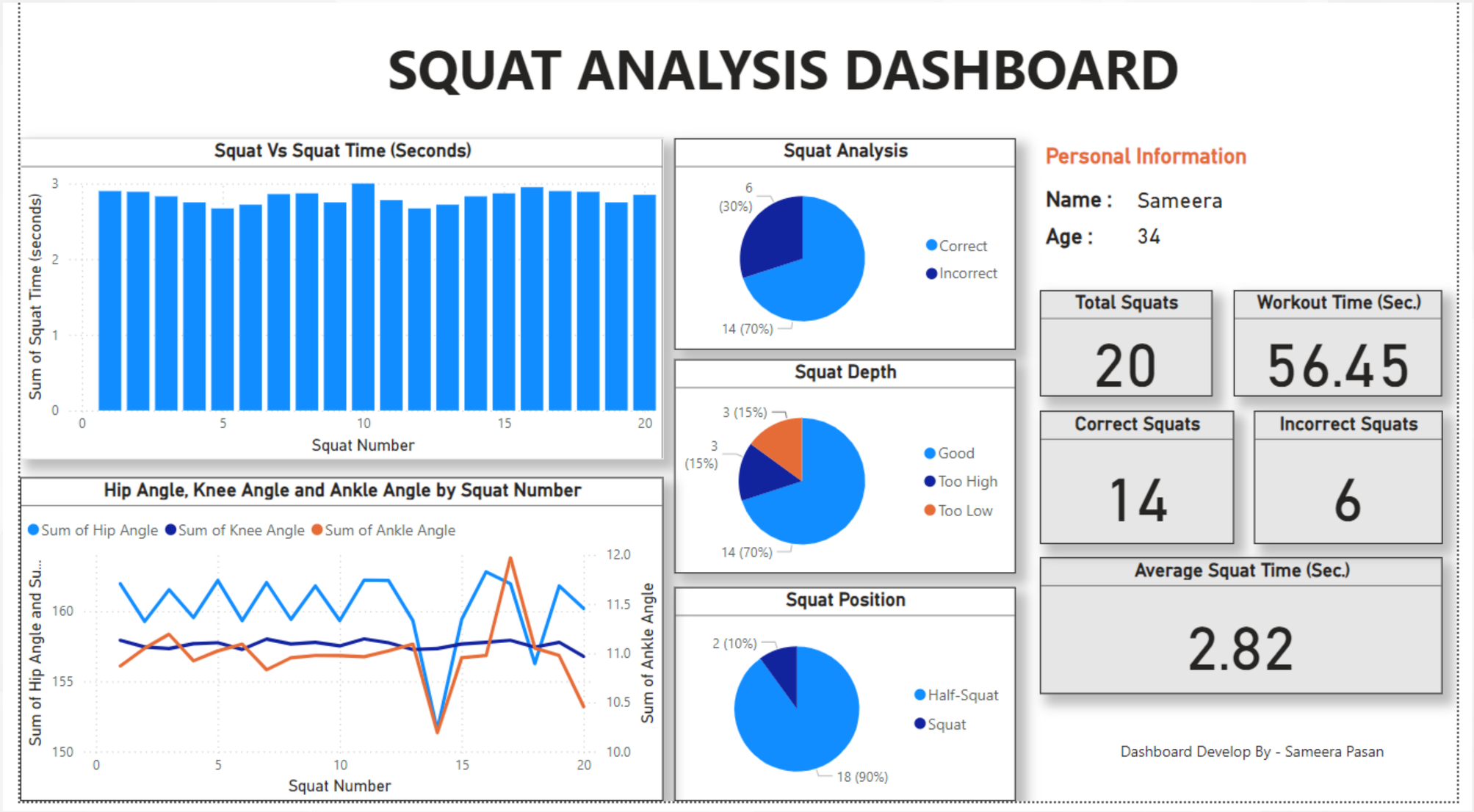
```
Enter Sportsman's name: INFO: Created TensorFlow Lite XNNPACK delegate for CPU.
Sameera
Enter Sportsman's age: 17
⊗ 0 ⚠ 0 ⓘ 52    0
```

final.py | squat_data.csv

squat_data.csv > data

```
1   Sportman Name,Sportman Age,Squat Number,Knee Angle,Hip Angle,Ankle Angle,Shoulder Angle,Squat Time (seconds),feedback,status
2   Sameera,22,1,157.91429234152534,161.94721221556685,10.869955987713391,21.60871416895004,2.77,Good,Half-Squat
3   Sameera,22,2,157.44274498061685,159.2581841738587,11.050444753764628,20.345294795193517,3.2,Good,Half-Squat
4   Sameera,22,3,157.32906267907742,161.5220690521706,11.192270419161646,22.098531244306322,4.83,Good,Half-Squat
5   Sameera,22,4,157.67600339534962,159.53513444804835,10.923291950499816,20.14317698905169,3.38,Good,Half-Squat
6   Sameera,22,5,157.741489656603,162.17481297973814,11.02434809084979,21.64674825296835,2.92,Good,Half-Squat
7   Sameera,22,6,157.2679203369362,159.31931187886693,11.092804695440472,20.05283004865753,2.8,Good,Half-Squat
8   Sameera,22,7,158.00549740728994,162.03169744550047,10.83232287357397,21.775312421095865,2.93,Good,Half-Squat
9   Sameera,22,8,157.6587832066509,159.40583048750025,10.953948789570685,20.433908792939615,3.24,Good,Half-Squat
10  Sameera,22,9,157.77241397952395,161.7903539127202,10.976207507071667,22.45276023690364,3.37,Good,Half-Squat
11  Sameera,22,10,157.5209511065977,159.32915590871414,10.975368081710558,20.57484791680099,4.53,Good,Half-Squat
12  Sameera,22,11,158.01554035633453,162.20923337666989,10.96396672452084,21.69050306053545,3.15,Good,Half-Squat
13
```

OUTPUT DEMONSTRATION



KEY TAKEAWAYS

- Squat analysis using MediaPipe allows fitness professionals and individuals to easily assess squat form.
- This approach focuses on detecting key body landmarks and angles, making it straightforward to track squat depth.
- The system offers real-time feedback, making it easy for users to improve their squat technique and reduce the risk of injury.
- It is a beginner-friendly approach to using computer vision for fitness analysis.

THANK YOU

Any Question?