Smartan AI Internship Task - Technical Report

Name: [Vikas Saini] **Task:** Form Correctness Detection Using Pose Estimation **Date:** December 12, 2025

1. Project Overview

This project implements a computer vision pipeline to analyze exercise form in real-time. Using MediaPipe Pose for keypoint detection, the system extracts skeletal landmarks and applies geometric rule-based logic to provide immediate feedback to the user. The goal is to prevent injury by detecting common form mistakes such as "flaring elbows" or "lifting too high."

2. Posture Rules & Logic

Per the assignment requirements, I have implemented logic for three distinct posture rules.

Rule 1: Lateral Raise (Wrist-Shoulder Alignment)

- **The Concept:** In a lateral raise, the weights should be lifted to shoulder height to maximize deltoid activation. Lifting the wrists significantly higher than the shoulders (above the "T-pose" line) shifts the load to the upper trapezius and increases the risk of shoulder impingement.
- **Technical Logic:**
    - I track the Y-coordinates of the LEFT_WRIST and LEFT_SHOULDER.
    - Since the coordinate system starts at the top-left (y=0), a smaller Y value means "higher" on the screen.
    - **Condition:** If Wrist.y < (Shoulder.y - Threshold), the system triggers the feedback: "ARMS TOO HIGH".

Rule 2: Bicep Curl (Elbow Stability)

- **The Concept:** A proper bicep curl requires the elbow to remain stationary at the side of the torso. If the elbow swings forward during the lift (shoulder flexion), the anterior deltoid takes over the work, reducing the effectiveness for the biceps.
- **Technical Logic:**
    - I calculate the geometric angle formed by three points: Shoulder → Elbow → Hip.
    - A perfectly vertical arm creates an angle close to 180°.
    - **Condition:** If the angle drops below 160° (indicating the elbow has moved significantly forward), the system triggers: "KEEP ELBOWS TUCKED".

Rule 3: Back Symmetry (Torso Alignment)

- **The Concept:** During standing exercises, the user should maintain a neutral spine and not lean to one side (asymmetry), which can lead to muscle imbalances or spinal strain.
- **Technical Logic:**
    - I compare the vertical positions of the LEFT_SHOULDER and RIGHT_SHOULDER.

- **Condition:** I calculate the absolute difference: abs(Left_Shoulder.y - Right_Shoulder.y). If this difference exceeds a normalized threshold (0.05), it indicates a side lean, triggering: "UNBALANCED POSTURE".

3. Challenges & Solutions

Challenge A: Self-Occlusion

- **Issue:** During exercises like the bicep curl, the forearm often crosses in front of the chest, potentially confusing the model about where the wrist is located.
- **Solution:** I utilized MediaPipe's min_tracking_confidence=0.5 parameter. This ensures that if the model loses visual track of a landmark, it relies on temporal consistency (tracking from previous frames) rather than guessing wildly.

Challenge B: Camera Noise & Jitter

- **Issue:** Raw webcam input often results in "jittery" keypoints, where the landmarks shake even if the user is standing still. This causes the angles to fluctuate rapidly.
- **Solution:** In a production environment, I would implement an **Exponential Moving Average (EMA) filter** to smooth the coordinates over time (smoothed = alpha * new + (1-alpha) * old). For this submission, I used strict angle thresholds to prevent flickering feedback.