**import cv2**

**import mediapipe as mp**

**import numpy as np**

**mp\_drawing = mp.solutions.drawing\_utils**

**mp\_pose = mp.solutions.pose**

**# Function to calculate angle**

**def calculate\_angle(a, b, c):**

**a = np.array(a) # First**

**b = np.array(b) # Mid**

**c = np.array(c) # End**

**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**

**# Function to count bicep curls**

**def count\_bicep\_curls(shoulder, elbow, wrist, stage, counter):**

**# Calculate angle**

**angle = calculate\_angle(shoulder, elbow, wrist)**

**# Visualize angle**

**cv2.putText(image, str(angle),**

**tuple(np.multiply(elbow, [640, 480]).astype(int)),**

**cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, (255, 255, 255), 2, cv2.LINE\_AA**

**)**

**# Curl counter logic**

**if angle > 160 and stage == 'down':**

**stage = 'up'**

**if angle < 30 and stage == 'up':**

**stage = 'down'**

**counter += 1**

**print("Bicep Curls Count:", counter)**

**return stage, counter**

**bicep\_counter = 0**

**bicep\_stage = 'down'**

**cap = cv2.VideoCapture(0)**

**with mp\_pose.Pose(min\_detection\_confidence=0.5, min\_tracking\_confidence=0.5) as pose:**

**while cap.isOpened():**

**ret, frame = cap.read()**

**if not ret:**

**break**

**image = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)**

**image.flags.writeable = False**

**results = pose.process(image)**

**image.flags.writeable = True**

**image = cv2.cvtColor(image, cv2.COLOR\_RGB2BGR)**

**try:**

**# Extract landmarks for bicep curls**

**landmarks = results.pose\_landmarks.landmark**

**left\_shoulder = [landmarks[mp\_pose.PoseLandmark.LEFT\_SHOULDER.value].x,**

**landmarks[mp\_pose.PoseLandmark.LEFT\_SHOULDER.value].y]**

**left\_elbow = [landmarks[mp\_pose.PoseLandmark.LEFT\_ELBOW.value].x,**

**landmarks[mp\_pose.PoseLandmark.LEFT\_ELBOW.value].y]**

**left\_wrist = [landmarks[mp\_pose.PoseLandmark.LEFT\_WRIST.value].x,**

**landmarks[mp\_pose.PoseLandmark.LEFT\_WRIST.value].y]**

**# Count bicep curls**

**bicep\_stage, bicep\_counter = count\_bicep\_curls(left\_shoulder, left\_elbow, left\_wrist, bicep\_stage, bicep\_counter)**

**except:**

**pass**

**# Render bicep curl counter**

**cv2.putText(image, 'Bicep Curls Count: ' + str(bicep\_counter),**

**(10, 30), cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (255, 255, 255), 2, cv2.LINE\_AA)**

**mp\_drawing.draw\_landmarks(image, results.pose\_landmarks, mp\_pose.POSE\_CONNECTIONS,**

**mp\_drawing.DrawingSpec(color=(245, 117, 66), thickness=2, circle\_radius=2),**

**mp\_drawing.DrawingSpec(color=(245, 66, 230), thickness=2, circle\_radius=2)**

**)**

**cv2.imshow('Mediapipe Feed', image)**

**if cv2.waitKey(10) & 0xFF == ord('q'):**

**break**

**cap.release()**

**cv2.destroyAllWindows()**