

Integrantes: Ruben Andre Cabrera Cermeño y Enrique Alejandro Orozco Mendoza

Nombre del profesor: Peter Jonathan Montalvo Garcia

Nombre de la materia: Visión computacional

Junio 2024

Índice

1. Tema escogido
2. Justificación del problema
3. Tecnologías implementadas
4. Problemas encontrados y como fueron solucionados por el equipo

1.Tema escogido

Detección de la ejecución correcta de ejercicios físicos(flexiones y sentadillas) utilizando Google Mediapipe

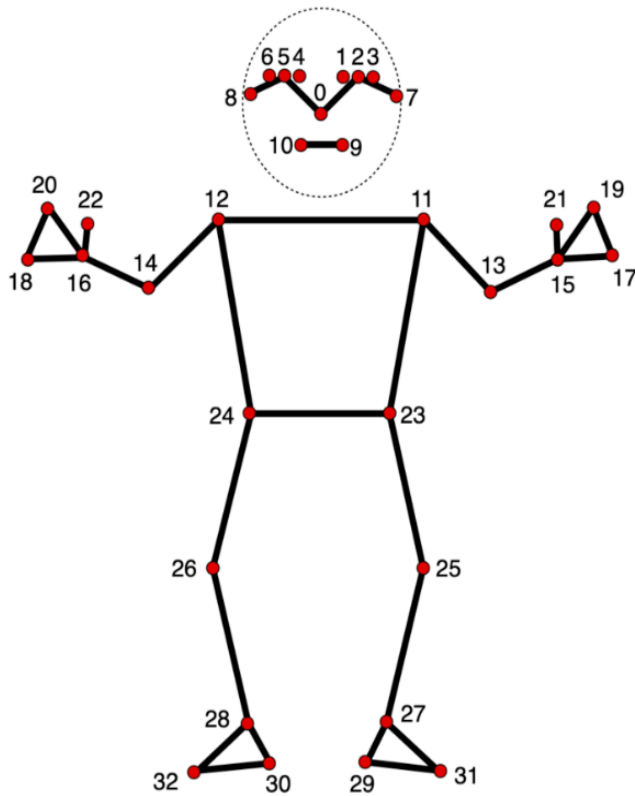
2. Justificación del problema

En el ámbito de la investigación en informática, específicamente en el campo de la visión por computadora, la detección precisa de posturas y movimientos humanos es un desafío técnico significativo. La capacidad de analizar automáticamente la postura durante la realización de ejercicios físicos tiene importantes aplicaciones prácticas, desde la prevención de lesiones hasta la mejora del rendimiento deportivo tanto en principiantes como en personas con un cierto grado de experiencia. Asimismo, se busca que el modelo brinde una retroalimentación instantánea, garantizando a los usuarios que están realizando los ejercicios de manera efectiva. En caso

contrario, el modelo identificará y mostrará los errores en la ejecución del ejercicio, permitiendo a los usuarios corregir su postura y técnica en tiempo real.. Al abordar este problema, se pueden desarrollar algoritmos avanzados, como el uso del algoritmo de detección de poses de MediaPipe que emplea una red neuronal CNN para detectar 33 puntos de referencia en el cuerpo humano y modelos de aprendizaje profundo que mejoren la precisión y eficiencia de los sistemas de visión por computadora.

3. Tecnologías implementadas

En nuestro proyecto vamos a implementar el uso del framework Mediapipe Pose, el cual permite detectar puntos de referencia en cuerpos humanos tanto en imágenes como en videos. Esto es esencial para analizar la postura, identificar ubicaciones clave del cuerpo y categorizar los movimientos, lo cual es crucial para evaluar la ejecución correcta de sentadillas y planchas, además de poder obtener una retroalimentación de forma inmediata. Mediapipe Pose utiliza una red neuronal convolucional (CNN) para detectar 33 puntos de referencia (landmarks) en el cuerpo humano. En nuestro proyecto, vamos a enfocarnos en los siguientes puntos de referencia específicos: hombros (izquierdo y derecho), codos (izquierdo y derecho), muñecas (izquierda y derecha), caderas (izquierda y derecha), rodillas (izquierda y derecha) y tobillos (izquierdo y derecho).



- 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

Por otra parte, OpenCV (Open Source Computer Vision Library) es una biblioteca de visión por computadora de código abierto que proporciona herramientas esenciales para el procesamiento de imágenes y videos. En nuestro proyecto, empleamos OpenCV para poder capturar y procesar secuencias de video en tiempo real, convertir imágenes de un formato de color a otro (específicamente de BGR a RGB), dibujar anotaciones y visualizaciones en los cuadros de video procesados, y mostrar los resultados en una interfaz gráfica en tiempo real. Esto permite una visualización en vivo del análisis de las sentadillas y planchas, además de proporcionar una retroalimentación inmediata sobre la postura y la forma.

En esa misma línea, NumPy es una biblioteca fundamental para la computación científica en Python que ofrece soporte para arrays y matrices multidimensionales, así como una amplia gama de funciones matemáticas para operar con estos datos de manera eficiente. En nuestro proyecto, NumPy se emplea para manipular y almacenar las coordenadas de los puntos de referencia detectados por MediaPipe, realizar los cálculos matemáticos necesarios para determinar los ángulos entre las articulaciones del cuerpo, y facilitar las operaciones vectoriales para el análisis de la postura. NumPy permite realizar estos cálculos de manera rápida y eficiente, lo cual es crucial para el procesamiento en tiempo real.

Así mismo, implementamos Matplotlib como una herramienta esencial para la visualización de datos del ejercicio. Específicamente, utilizamos Matplotlib para generar gráficos que muestran la variación de los ángulos de las articulaciones a lo largo del tiempo. Estos gráficos se actualizan y se muestran en la interfaz del usuario durante la ejecución del ejercicio, proporcionando una retroalimentación visual clara y detallada.

4. Problemas encontrados y como fueron solucionados por el equipo

Problema 1: Precisión en la Detección de Puntos Clave

Al principio, la detección de los puntos clave del cuerpo (landmarks) no era precisa en todos los frames, lo que afectaba la precisión del cálculo de ángulos y la retroalimentación.

Solución: Ajustamos los parámetros de confianza de Mediapipe para mejorar la precisión de la detección de puntos clave. Específicamente, configuramos `min_detection_confidence` y `min_tracking_confidence` a valores más altos en la inicialización de Mediapipe Pose.

Problema 2: Latencia en la Retroalimentación

Había una latencia notable en la retroalimentación visual, lo que dificultaba la experiencia del usuario en tiempo real.

Solución: Optimizamos el procesamiento de cada frame asegurando que solo se realizaban los cálculos necesarios. Procesamos únicamente los frames que contenían puntos clave detectados, evitando así cálculos innecesarios en frames sin detección de landmarks. Además, implementamos un manejo robusto de excepciones para capturar y registrar errores sin detener la ejecución del programa.

Problema 3: Visualización Progreso y la Precisión

La visualización de la barra de progreso y la precisión no era clara y a veces se superponía con otros elementos de la interfaz

Solución: Mejoramos el diseño visual de la interfaz, ajustando la posición y el tamaño de los elementos gráficos para asegurar una visualización clara y no obstructiva.

Problema 4: Variabilidad en la Iluminación

La variabilidad en la iluminación afectaba la detección de puntos clave en algunos frames, causando inconsistencias en la retroalimentación.

Solución: Para manejar la variabilidad en la iluminación que afectaba la detección de puntos clave, implementamos una conversión de la imagen de BGR a RGB, asegurando que los colores y la iluminación se manejen correctamente antes de procesar la imagen con Mediapipe. Esta conversión es crucial porque Mediapipe espera imágenes en formato RGB para una detección precisa. Además, verificamos la presencia de puntos clave (`results.pose_landmarks`) antes de



procesarlos y visualizarlos, lo que evita cálculos y retroalimentación basados en datos incompletos o incorrectos debido a problemas de iluminación.