

Detección de Postura Corporal en Tiempo Real

Introducción

Este programa de detección de postura en tiempo real permite al usuario verificar su postura a través de la cámara de su dispositivo. La herramienta utiliza MediaPipe para la detección de puntos clave en el cuerpo y calcula la distancia entre la nariz y el pecho, así como el ángulo entre los hombros. El usuario establece una postura inicial de referencia, y el programa evalúa la postura actual en función de la posición inicial, advirtiéndolo al usuario en caso de desvío. Es especialmente útil para aquellos interesados en mejorar la ergonomía o mantener una postura correcta al trabajar o realizar actividades sedentarias.

Prerrequisitos

Para ejecutar este programa correctamente, el entorno de trabajo necesita cumplir con algunos requisitos básicos:

1. **Instalación de Python:** El programa está desarrollado en Python 3.x, por lo que es necesario tener una versión compatible de Python instalada.
2. **Bibliotecas requeridas:** Las principales bibliotecas utilizadas son `cv2` (OpenCV), `mediapipe` y `math`. Estas pueden instalarse ejecutando los siguientes comandos en la terminal:

```
pip install opencv-python
```

```
pip install mediapipe
```

Herramientas Utilizadas

OpenCV (cv2): OpenCV es una biblioteca ampliamente utilizada en visión computacional que permite procesar y mostrar video en tiempo real. En este programa, se utiliza para capturar el video de la cámara y dibujar puntos y textos en la pantalla.

MediaPipe: MediaPipe es una biblioteca de aprendizaje automático que proporciona modelos pre entrenados para detectar y rastrear puntos clave en el cuerpo humano en tiempo real. Aquí, se utiliza el modelo **Pose**, que detecta puntos específicos en el cuerpo, como los hombros y la nariz, permitiendo calcular distancias y ángulos para evaluar la postura.

Math: Esta biblioteca estándar de Python permite realizar cálculos matemáticos avanzados, como la distancia euclidiana y el ángulo entre puntos, que son fundamentales para determinar la posición del usuario y la calidad de su postura.

Funcionamiento del Programa

El programa se ejecuta de manera continua, procesando cada cuadro capturado por la cámara para evaluar si la postura del usuario se mantiene en los parámetros deseados. A continuación, se describe en detalle cada sección del código:

1. Importación de Bibliotecas y Configuración de MediaPipe

```
import cv2
import mediapipe as mp
import math
```

Se importan las bibliotecas **cv2**, **mediapipe**, y **math** para utilizar sus funcionalidades de procesamiento de imágenes y cálculos.

Se inicializa **MediaPipe** para detectar los puntos clave en el cuerpo mediante el modelo **Pose**.

2. Cálculo de Distancias y Ángulos

El programa incluye dos funciones importantes:

- **calcular_distancia(p1, p2)**: Calcula la distancia entre dos puntos en el espacio utilizando la fórmula de distancia euclidiana.
- **calcular_angulo(p1, p2)**: Calcula el ángulo entre dos puntos. Este cálculo se utiliza para determinar la inclinación entre los hombros.

3. Captura de Vídeo

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

Se inicializa la captura de video desde la cámara web del dispositivo. A través de un bucle **while** continuo, el programa analiza cada cuadro del video en tiempo real.

4. Procesamiento del Cuadro y Detección de Puntos Clave

El programa convierte el cuadro de BGR a RGB (formato requerido por MediaPipe) y procesa el cuadro para detectar puntos clave en la postura del usuario:

```
rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
```

```
result = pose.process(rgb_frame)
```

MediaPipe identifica la posición de varios puntos del cuerpo. El programa usa tres puntos específicos:

- **Hombro Izquierdo**
- **Hombro Derecho**
- **Nariz**

Las coordenadas de estos puntos se utilizan para calcular el ángulo y la distancia necesarios para determinar la postura.

5. Determinación de la Postura Inicial

El programa permite al usuario adoptar una postura correcta y, al presionar la tecla 's', almacena los valores de referencia para la distancia y el ángulo:

Estos valores guardados actúan como base para evaluar la postura en tiempo real, permitiendo al programa determinar si la posición actual del usuario se desvía de los valores registrados.

```
if key == ord('s'):
```

```
    distancia_referencia = distancia_cara_pecho
```

```
    angulo_referencia = angulo_inclinacion
```

```
    print("Valores de referencia guardados.")
```

6. Evaluación de la Postura

El programa compara la postura actual con los valores de referencia y utiliza umbrales para determinar si la postura es incorrecta. En este caso, el programa alerta al usuario si detecta:

- **Inclinación excesiva de los hombros**
- **Distancia insuficiente entre la nariz y el pecho**

7. Visualización en Pantalla

En la pantalla, el programa muestra:

- **Ángulo de inclinación de los hombros**
- **Distancia entre la nariz y el pecho**
- **Mensajes de advertencia o confirmación de la postura**

```
cv2.putText(frame, 'Mala postura: Inclinacion excesiva!', (10, 150),  
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
```

8. Problemas y posibles soluciones

Algunos problema encontrados a la hora de realizar el proyecto son:

Entorno: Dependiendo de el entorno del usuario, principalmente, la disposición de su escritorio, si usa una cámara externa o propia del dispositivo, puede que no sea posible detectar toda la información necesaria para proporcionar una respuesta exacta, sin embargo, gracias a mediapipe se realizan unas aproximaciones de forma que el programa puede continuar aunque con una menor precisión.

Diversidad: Debido a que cada persona es diferente si usasemos unos parámetros absolutos, ciertas personas podrían ser excluidas, haciendo nuestro software inservible, para ello al iniciar el programa se pide adoptar una buena postura y pulsar una tecla para captar la posición del usuario y poder usar datos relativos.

Algoritmo: Como el objetivo del proyecto era desarrollar una solución a el problema de la postura se ha optado por un algoritmo sencillo para determinar si la postura es correcta o no en base a unos valores umbrales, sin embargo, al ser tan sencillo es posible que falle o, que si adoptamos una nueva postura que sea correcta pero difiere mucho de la captada nos indique que tenemos mala postura, mejorar el algoritmo para determinar estos casos nos daría la solución a este problema.

Conclusión

Este programa de detección de postura ofrece una forma sencilla y accesible para monitorear la postura en tiempo real usando técnicas avanzadas de visión por computadora y aprendizaje automático. Al permitir al usuario ajustar los valores de referencia basados en su postura ideal, el sistema es adaptable y proporciona una retroalimentación efectiva. Este programa es particularmente útil para quienes buscan mejorar sus hábitos posturales y mantener una ergonomía adecuada en situaciones como el trabajo de oficina o el estudio.