

Documentación del Proyecto: Webcam con Google Cloud Vision API (Python)

Descripción General

Aplicación en Python que utiliza una interfaz gráfica para capturar una imagen desde la webcam y analizarla con Google Cloud Vision API. Permite:

- Ver video en tiempo real.
- Capturar una imagen.
- Analizar la imagen para detectar etiquetas y rostros.

Tecnologías Usadas

Lenguaje: Python 3

Librerías:

- opencv-python: Captura de video.
- google-cloud-vision: Conexión con la Vision API.
- Pillow: Conversión de imágenes.
- tkinter: Interfaz gráfica.

Estructura Principal del Código (webcam_vision_gui.py)

- Credenciales: Se define la variable de entorno `GOOGLE_APPLICATION_CREDENTIALS` para conectar con Google Cloud.
- Clase WebcamApp:
- `update_video()`: Muestra el video en la interfaz.
- `capture_photo()`: Guarda un frame como imagen.
- `analyze_photo()`: Envía la imagen a la Vision API y muestra etiquetas y rostros.

Flujo de Uso

1. Se inicia la aplicación y se muestra el video en la ventana.
2. El usuario captura una imagen (guardada como `captured.jpg`).
3. El usuario analiza la imagen.
4. Los resultados se muestran en un cuadro de texto.

Requisitos

Python 3.6+

Conexión a internet.

Archivo de credenciales JSON válido de Google Cloud.

Instalación de dependencias:

`pip install opencv-python google-cloud-vision Pillow`

Comparación con C#

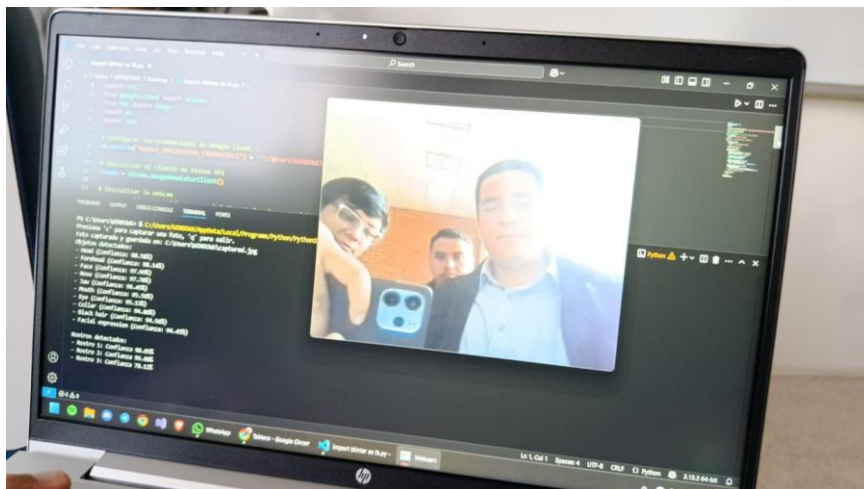
- Python usa tkinter en lugar de Windows Forms.
- opencv-python reemplaza a AForge.Video.
- Uso de pip facilita la gestión de librerías.

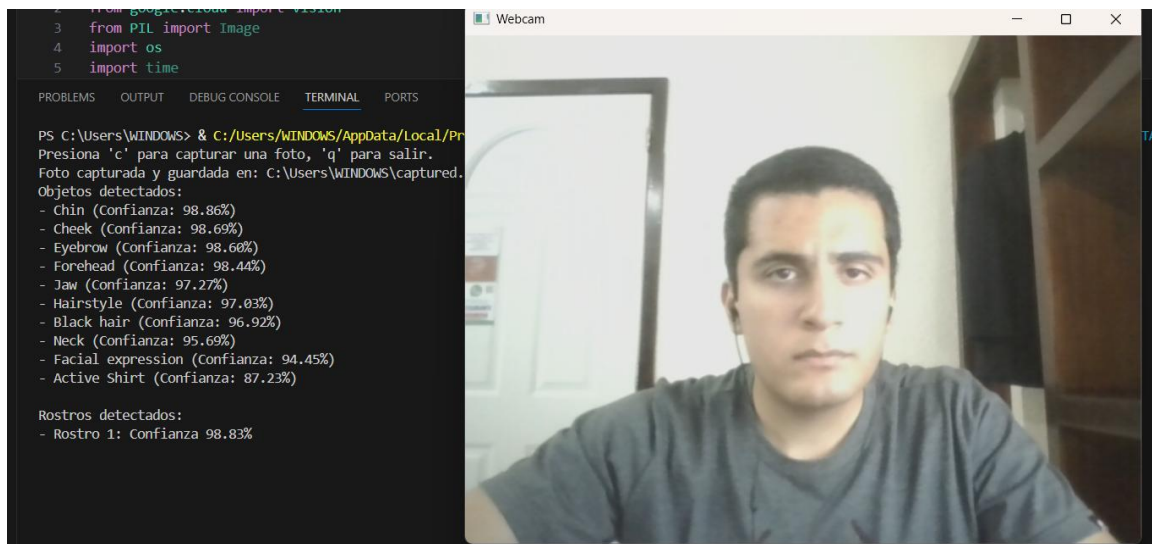
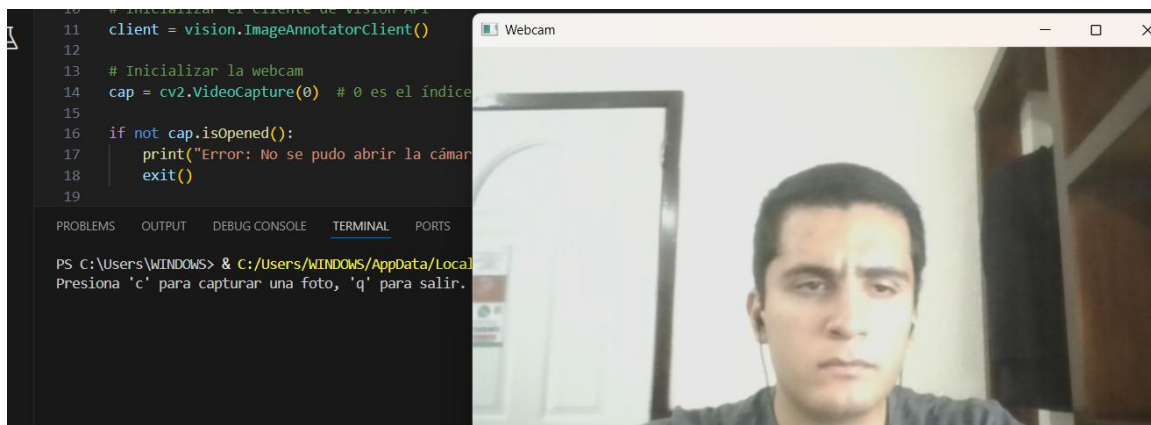
Escalabilidad

- Se puede extender para detectar texto u objetos específicos.
- Mejora de interfaz con bibliotecas como PyQt.

Documentación de imágenes y link del video de prueba:

<https://youtu.be/b8jXmUCQhQk?si=gxB2a2Yb3RzE7UqU>





Prompts Utilizados:

1. Crea una aplicación en Python con una interfaz gráfica usando Tkinter que capture video desde una webcam, permita tomar una foto con un botón, la guarde como 'captured.jpg', y analice la imagen con Google Cloud Vision API para detectar objetos y rostros, mostrando los resultados en un área de texto desplazable. Usa opencv-python para la captura de video, google-cloud-vision para interactuar con la API, y Pillow para manejar imágenes en la interfaz.
2. Configura el programa para usar Google Cloud Vision API con un proyecto existente (ID: friendly-path-459404-a8) usando un archivo JSON de credenciales ubicado en 'C:\Users\jorge\OneDrive\Desktop\ApiDeteccionface\friendly-path-459404-a8-cdbaf9e51a09.json', asegurándote de que el código cargue las credenciales y maneje errores si no se encuentran.

3. Proporcióname el código completo en un archivo Python, asegurándome de que incluya toda la lógica para capturar video, guardar imágenes, analizar con Google Cloud Vision API, y mostrar resultados, junto con la configuración de las credenciales.

Participantes:

- Abner Joel Mateo Hernández 0905-23-15220
- Jorge Mario Cano Cobon 0905-24-10433
- Eduardo Gabriel Visoni Morales 0905-22-1146