

Integrante

Alejandro campos Leonel

cruz lopez Fernando

vega

Introducción

El objetivo principal de este proyecto es realizar la detección de rostros en tiempo real utilizando la cámara web y Python. Para ello, se utiliza una librería de Python que permite acceder a la cámara y procesar las imágenes capturadas sin depender de servicios externos que puedan introducir latencia o costos adicionales.

El código desarrollado utiliza la librería MediaPipe para la detección facial en tiempo real, combinada con OpenCV para la captura y visualización de video desde la cámara web.

Equipo y Roles

- Alejandro: Encargado de la generación del código en lenguaje Python, utilizando la herramienta Gemini para facilitar la escritura y estructuración del código.
- Leonel: Responsable de la documentación del proyecto, asegurando que toda la información técnica y de desarrollo esté clara y organizada.
- Fernadon: Investigación y análisis de las APIs disponibles para determinar cuál es la más adecuada para la detección facial en el programa.

Herramientas y Tecnologías Utilizadas

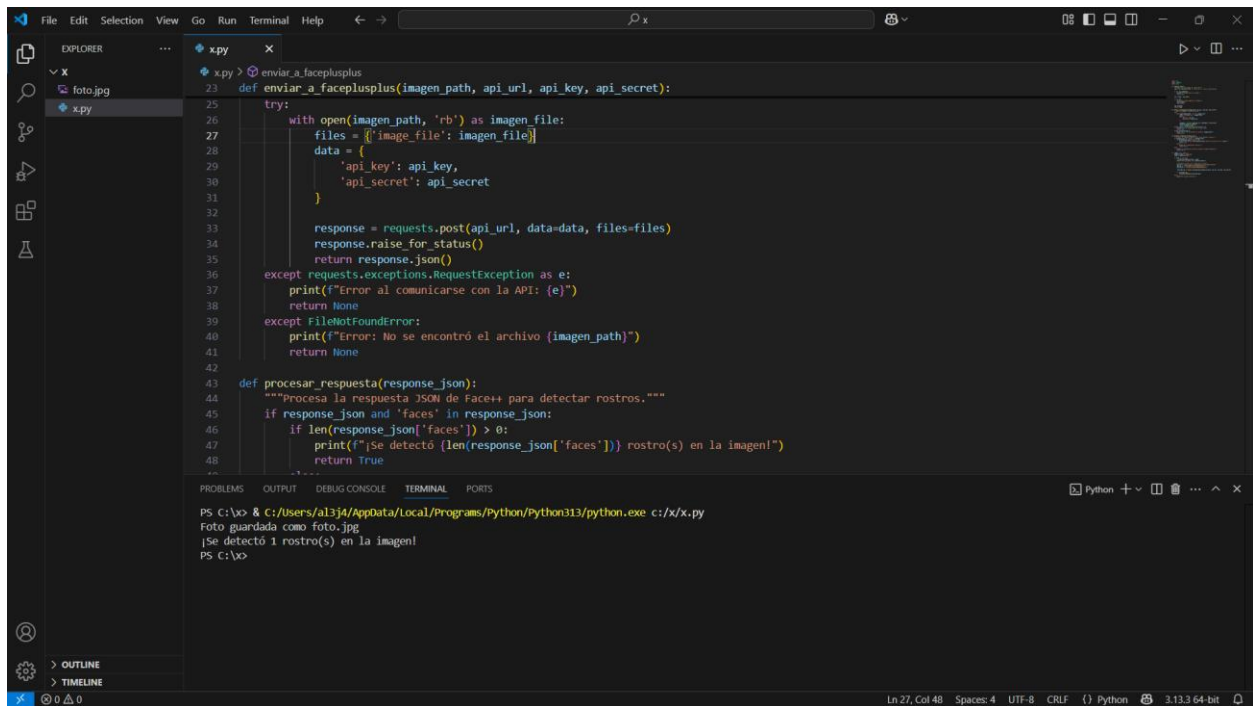
- Gemini: Inteligencia Artificial utilizada para generar el código en Python.
- MediaPipe: Librería para detección facial en tiempo real.
- OpenCV: Biblioteca para captura y procesamiento de video desde la cámara web.
- Face++ (F++): API investigada y utilizada para detección facial, aunque presentó problemas durante la ejecución.

Problemas Encontrados

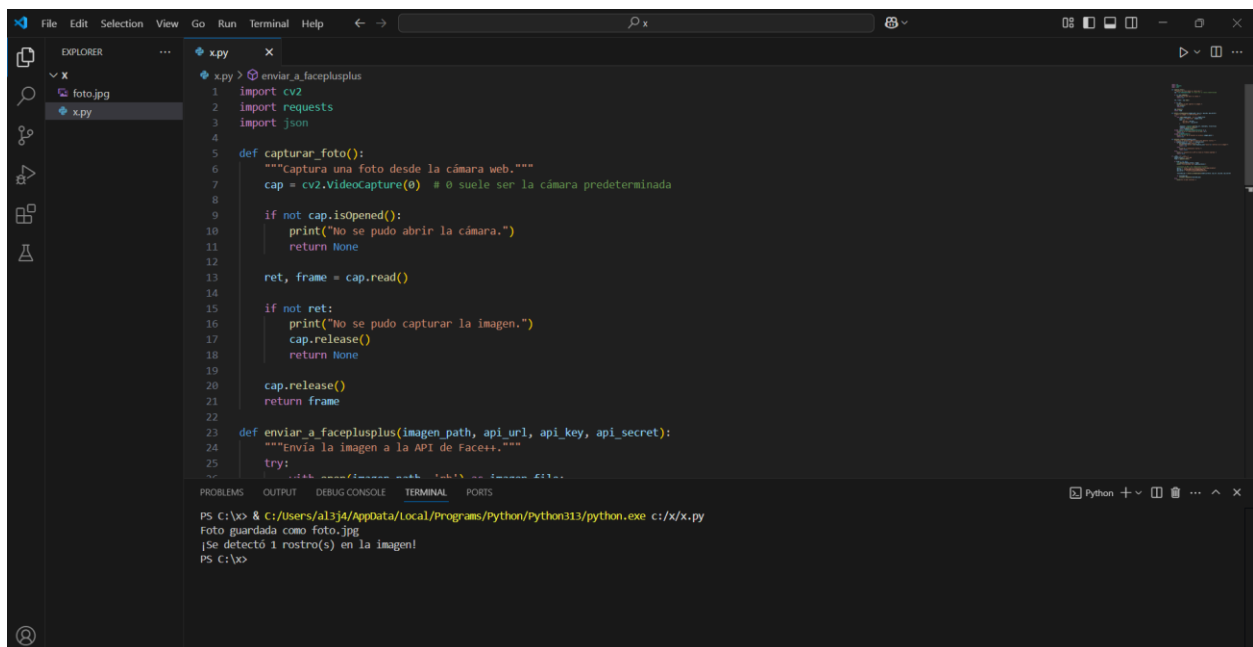
Durante el desarrollo, se presentaron dificultades con la instalación de las librerías necesarias, ya que inicialmente no estaban descargadas en el entorno de trabajo. Este inconveniente fue solucionado instalando correctamente las dependencias requeridas.

Además, se experimentaron problemas al ejecutar la API Face++, la cual fue la API seleccionada para la detección facial. Estos problemas afectaron la ejecución del programa, lo que motivó la búsqueda de soluciones alternativas como el uso de librerías locales (MediaPipe y OpenCV) para lograr la detección en tiempo real sin depender de servicios externos.

Esta son las imagen del programa funcional



```
File Edit Selection View Go Run Terminal Help
xpy
EXPLORER
x
foto.jpg
x.py
23 def enviar_a_faceplusplus(imagen_path, api_url, api_key, api_secret):
24     try:
25         with open(imagen_path, 'rb') as imagen_file:
26             files = {'image_file': imagen_file}
27             data = {
28                 'api_key': api_key,
29                 'api_secret': api_secret
30             }
31         response = requests.post(api_url, data=data, files=files)
32         response.raise_for_status()
33         return response.json()
34     except requests.exceptions.RequestException as e:
35         print(f"Error al comunicarse con la API: {e}")
36         return None
37     except FileNotFoundError:
38         print(f"Error: No se encontró el archivo {imagen_path}")
39         return None
40
41 def procesar_respuesta(response_json):
42     """Procesa la respuesta JSON de Face++ para detectar rostros."""
43     if response_json and 'faces' in response_json:
44         if len(response_json['faces']) > 0:
45             print(f"Se detectó {len(response_json['faces'])} rostro(s) en la imagen!")
46             return True
47     return False
48
49 if __name__ == '__main__':
50     imagen_path = 'foto.jpg'
51     api_url = 'https://api.faceplusplus.com/api/v2/face'
52     api_key = 'API_KEY'
53     api_secret = 'API_SECRET'
54     response = enviar_a_faceplusplus(imagen_path, api_url, api_key, api_secret)
55     if response:
56         procesar_respuesta(response)
57     else:
58         print("Error al enviar la imagen a la API de Face++.")
59
60 PS C:\> & C:/Users/al3j4/AppData/Local/Programs/Python/Python313/python.exe c:/x/x.py
61 Foto guardada como foto.jpg
62 ¡Se detectó 1 rostro(s) en la imagen!
63 PS C:\>
```



```
File Edit Selection View Go Run Terminal Help
xpy
EXPLORER
x
foto.jpg
x.py
1 import cv2
2 import requests
3 import json
4
5 def capturar_foto():
6     """Captura una foto desde la cámara web."""
7     cap = cv2.VideoCapture(0) # 0 suele ser la cámara predeterminada
8
9     if not cap.isOpened():
10         print("No se pudo abrir la cámara.")
11         return None
12
13     ret, frame = cap.read()
14
15     if not ret:
16         print("No se pudo capturar la imagen.")
17         cap.release()
18         return None
19     cap.release()
20     return frame
21
22 def enviar_a_faceplusplus(imagen_path, api_url, api_key, api_secret):
23     """Envía la imagen a la API de Face++."""
24     try:
25         with open(imagen_path, 'rb') as imagen_file:
26             files = {'image_file': imagen_file}
27             data = {
28                 'api_key': api_key,
29                 'api_secret': api_secret
30             }
31         response = requests.post(api_url, data=data, files=files)
32         response.raise_for_status()
33         return response.json()
34     except requests.exceptions.RequestException as e:
35         print(f"Error al comunicarse con la API: {e}")
36         return None
37     except FileNotFoundError:
38         print(f"Error: No se encontró el archivo {imagen_path}")
39         return None
40
41 def procesar_respuesta(response_json):
42     """Procesa la respuesta JSON de Face++ para detectar rostros."""
43     if response_json and 'faces' in response_json:
44         if len(response_json['faces']) > 0:
45             print(f"Se detectó {len(response_json['faces'])} rostro(s) en la imagen!")
46             return True
47     return False
48
49 if __name__ == '__main__':
50     frame = capturar_foto()
51     if frame is not None:
52         imagen_path = 'foto.jpg'
53         api_url = 'https://api.faceplusplus.com/api/v2/face'
54         api_key = 'API_KEY'
55         api_secret = 'API_SECRET'
56         response = enviar_a_faceplusplus(imagen_path, api_url, api_key, api_secret)
57         if response:
58             procesar_respuesta(response)
59         else:
60             print("Error al enviar la imagen a la API de Face++.")
61     else:
62         print("Error al capturar la imagen.")
63
64 PS C:\> & C:/Users/al3j4/AppData/Local/Programs/Python/Python313/python.exe c:/x/x.py
65 Foto guardada como foto.jpg
66 ¡Se detectó 1 rostro(s) en la imagen!
67 PS C:\>
```

