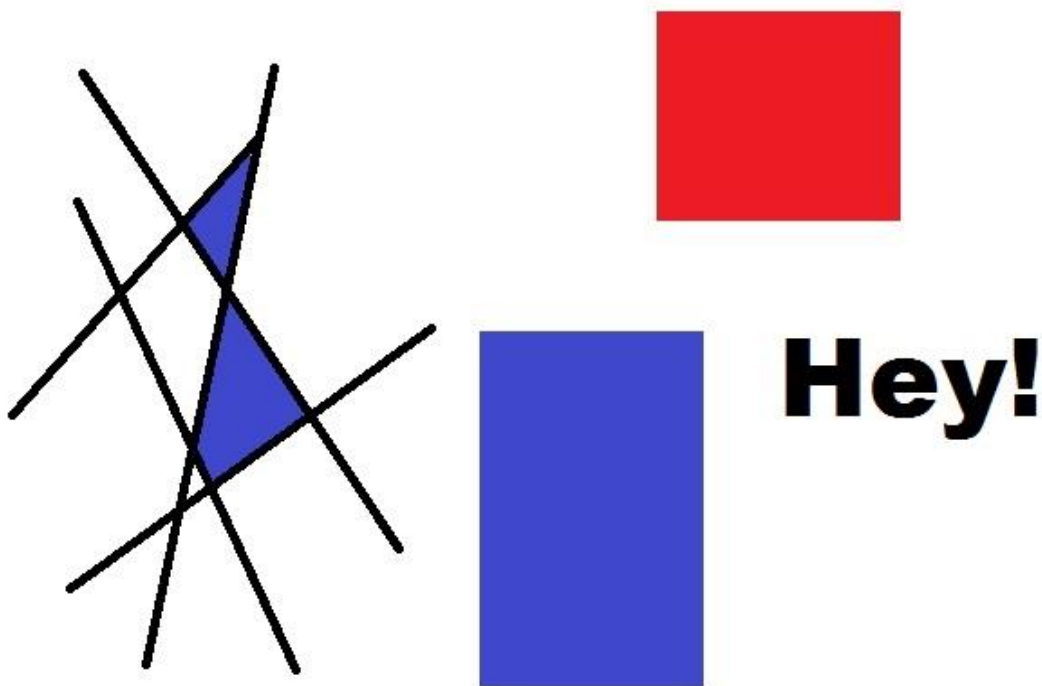


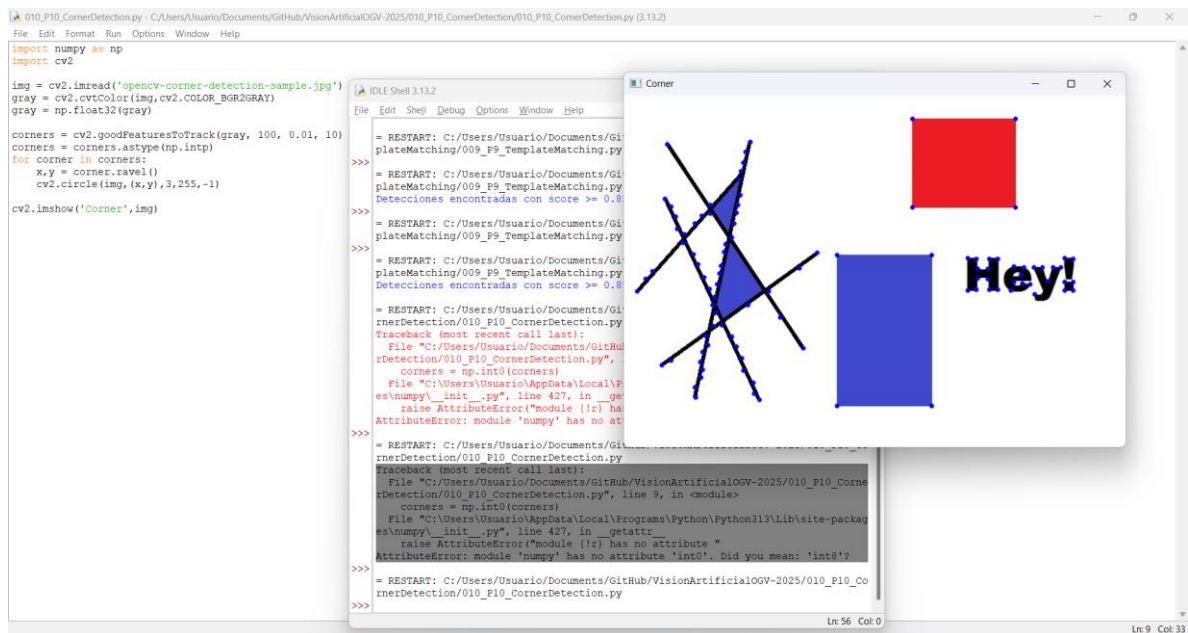
010_P10_Corner Detection

- Inicie sesión en mi Github
- Creé un nuevo archivo para la práctica 10 en Python
- Vi el tutorial del video de la práctica 10 de Python “Corner Detection OpenCV Python Tutorial”
- Puse el código que venía de ejemplo y vi el video, aparece que con este código puede detectar las esquinas y rastrear los movimientos cómo para hacer un modelado en 3D y ya con esto reconocer diferentes formas o patrones.
- Lo ejecuté y también tomé de ejemplo la imagen que viene por defecto de lo que usa de ejemplo.
- Al principio me marcó un error en la terminal de ejecutar porque una línea que el usaba ya no servía para mi versión del Python que tengo y le pedí ayuda a chat con esto que era antes de `corners = np.int0(corners)` y lo cambié por `corners = corners.astype(np.intp)` que ya este se ajusta al tamaño del puntero.
- Ya con esto me compiló con éxito y me salió justo cómo en el video de donde detectaba todas las esquinas o bordes de las figuras y tomé su respectiva evidencia.
- Después intenté con mi propia imagen y la hice en Paint con unas formas y ya hice el principal objetivo del profe que es la imagen que deseen separar por medio de un ROI el fondo de la imagen dejando únicamente el ROI al cual se le buscarán todas las esquinas, salió bien y ya también tomé su respectiva evidencia
- Ya para finalizar, creé el reporte. -Subí todos los archivos a mi GithubDesktop sobre la práctica 10 “Corner Detection”.

IMAGEN DE EJEMPLO:



PRUEBA 1:



MI IMAGEN DE EJEMPLO QUEN HICE:



PRUEBA 2:



MI CÓDIGO:

```
import numpy as np
import cv2

img = cv2.imread('F1.jpg')
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
gray = np.float32(gray)

corners = cv2.goodFeaturesToTrack(gray, 100, 0.01, 10)
corners = corners.astype(np.intp)
for corner in corners:
    x,y = corner.ravel()
    cv2.circle(img,(x,y),3,255,-1)

cv2.imshow('Corner',img)
```

ENLACE DE MI REPOSITORIO DE GITHUB:

<https://github.com/Ing-OscarValencia/VisionArtificialOGV-2025.git>