



Practica 2

Ingeniería en Mecatrónica
6to semestre

Arturo Alejandro Guzman Perez - 22110356

Objetivo

Aplicar operaciones aritméticas y lógicas en imágenes para combinarlas y superponerlas de forma selectiva.

Procedimiento

Carga de las imágenes:

```
import cv2
```

```
import numpy as np
```

```
img1 = cv2.imread('image1.png') img2 =  
cv2.imread('image2.png')
```

1. Suma básica de imágenes:

```
add = cv2.add(img1, img2)  
cv2.imshow('Addition', add)
```

2. Combinación ponderada:

- Mezclar imágenes con diferentes pesos:

```
weighted = cv2.addWeighted(img1, 0.7, img2, 0.3, 0) cv2.imshow('Weighted', weighted)
```

3. Superposición con máscara:

- Crea una máscara para superponer solo el contenido de interés:

```
img2_gray = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)
```

```
_, mask = cv2.threshold(img2_gray, 220, 255, cv2.THRESH_BINARY_INV)
```

```
mask_inv = cv2.bitwise_not(mask)
```

- Aplique la mascarilla para combinar las imágenes:

```
roi = img1[0:img2.shape[0], 0:img2.shape[1]] img1_bg =  
cv2.bitwise_and(roi, roi, mask=mask_inv) img2_fg =  
cv2.bitwise_and(img2, img2, mask=mask) dst = cv2.add(img1_bg,  
img2_fg) img1[0:img2.shape[0], 0:img2.shape[1]] = dst  
cv2.imshow('Result', img1)
```

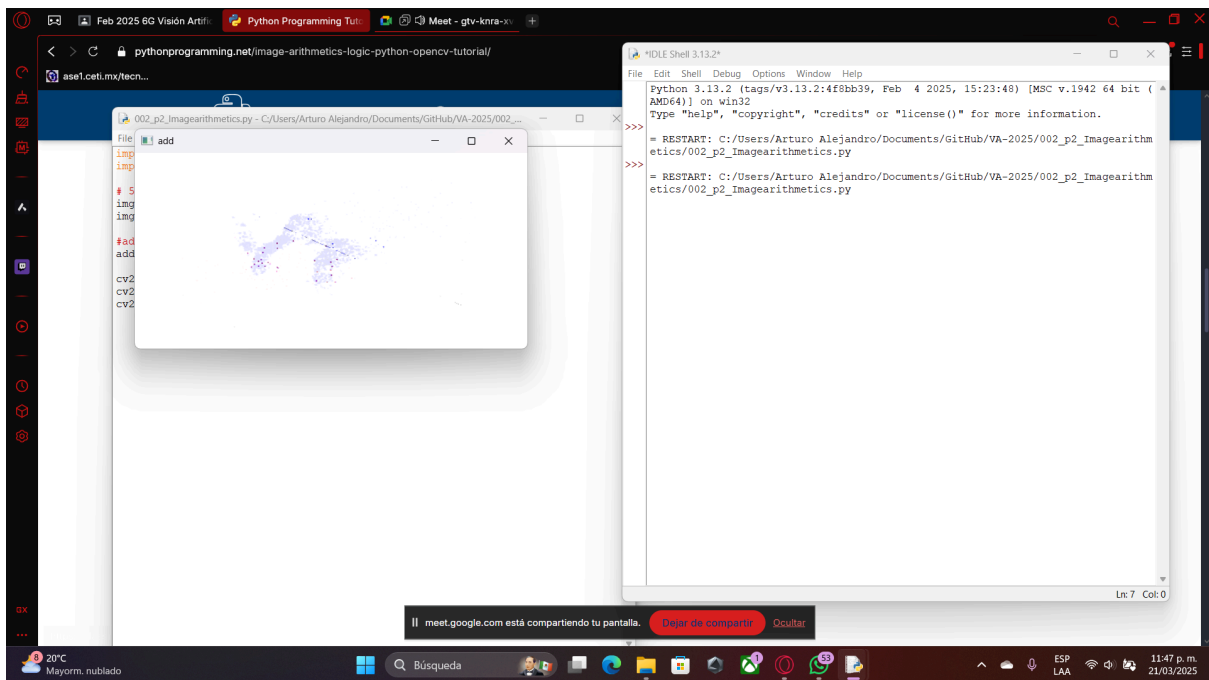
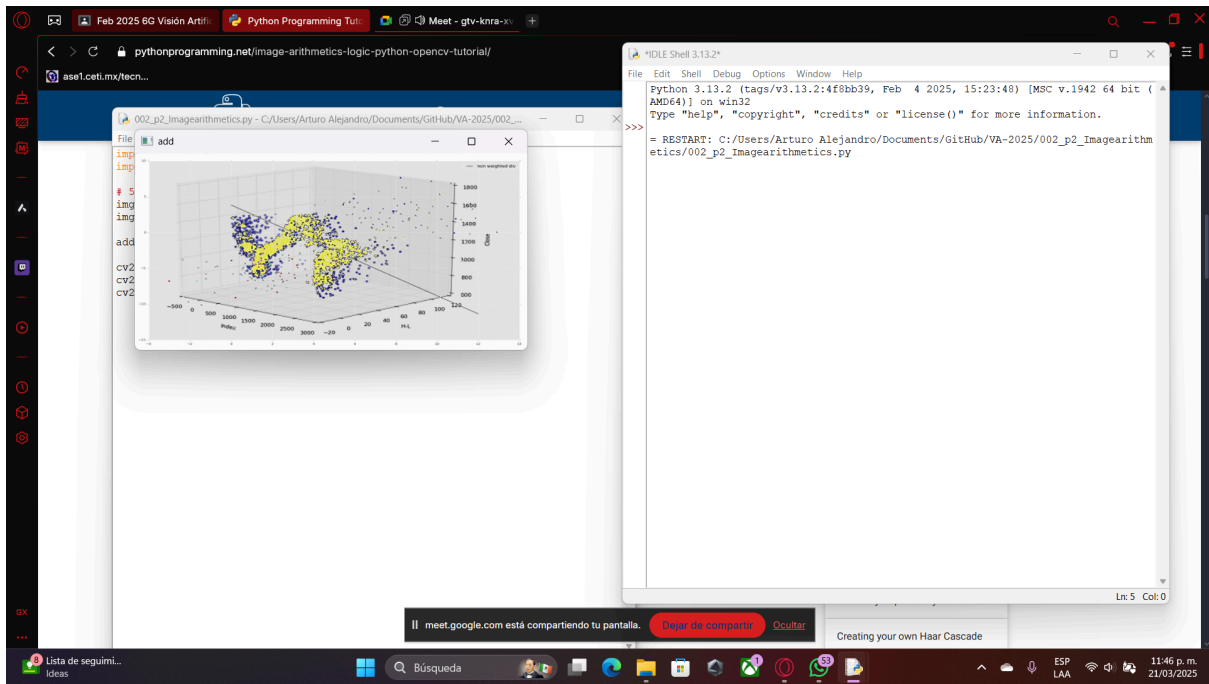
4. Finalizar y liberar

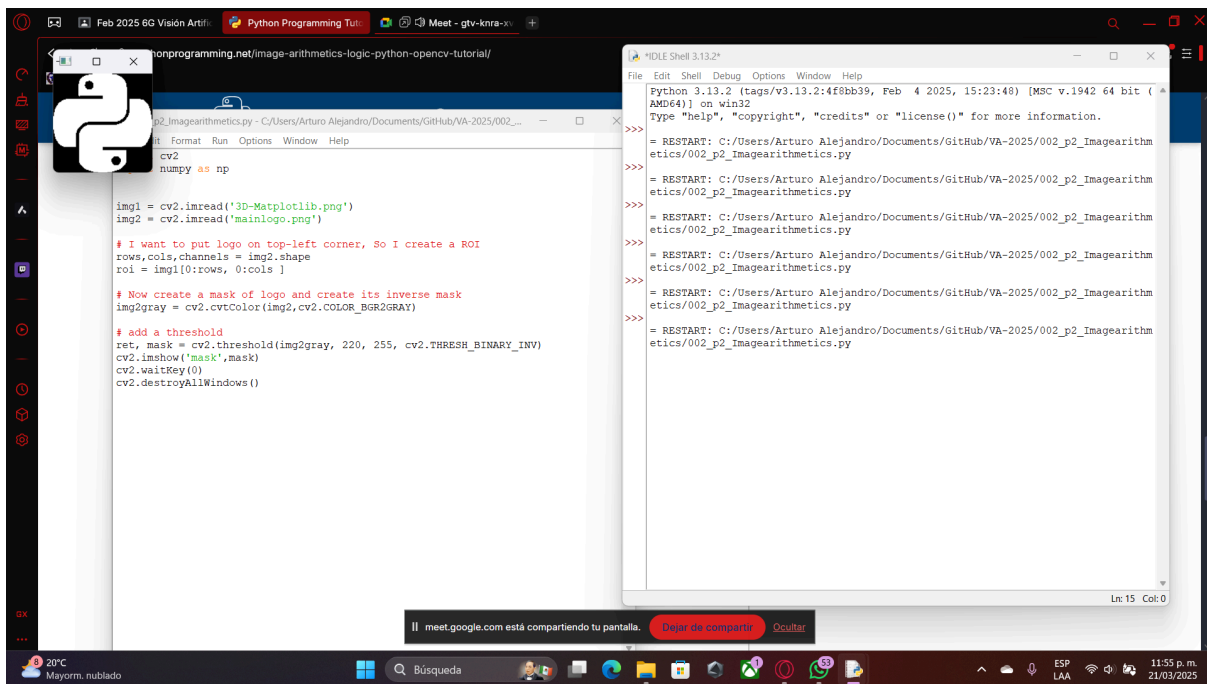
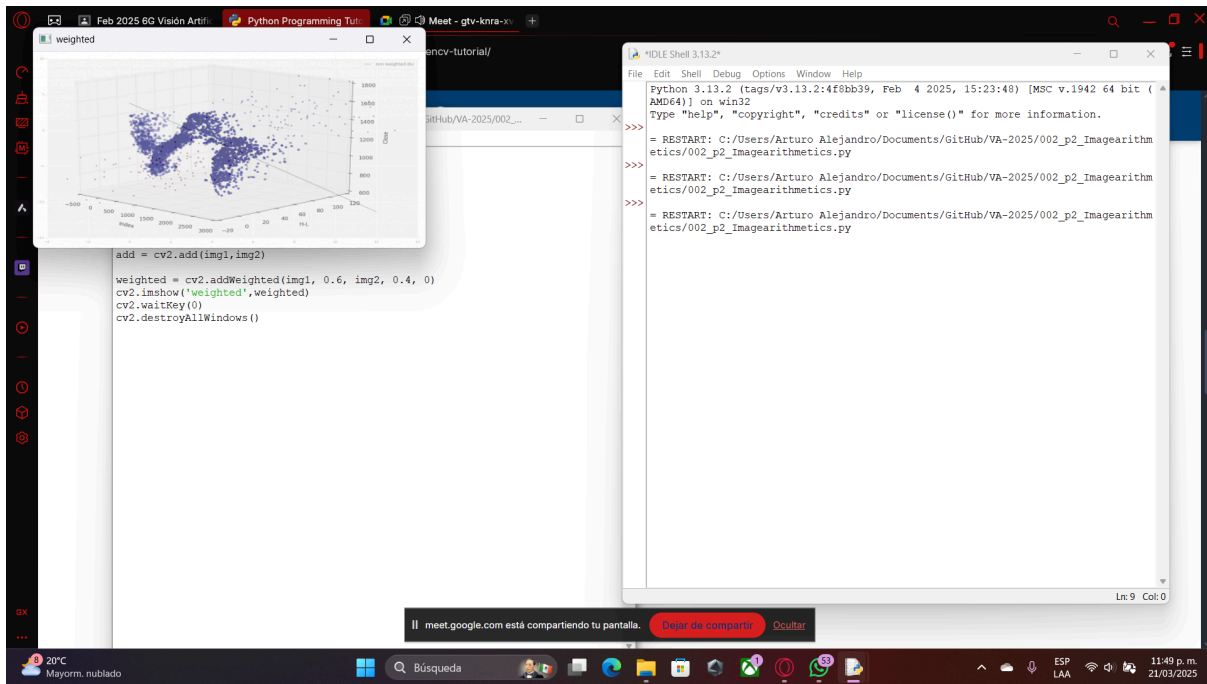
recursos: `cv2.waitKey(0)`

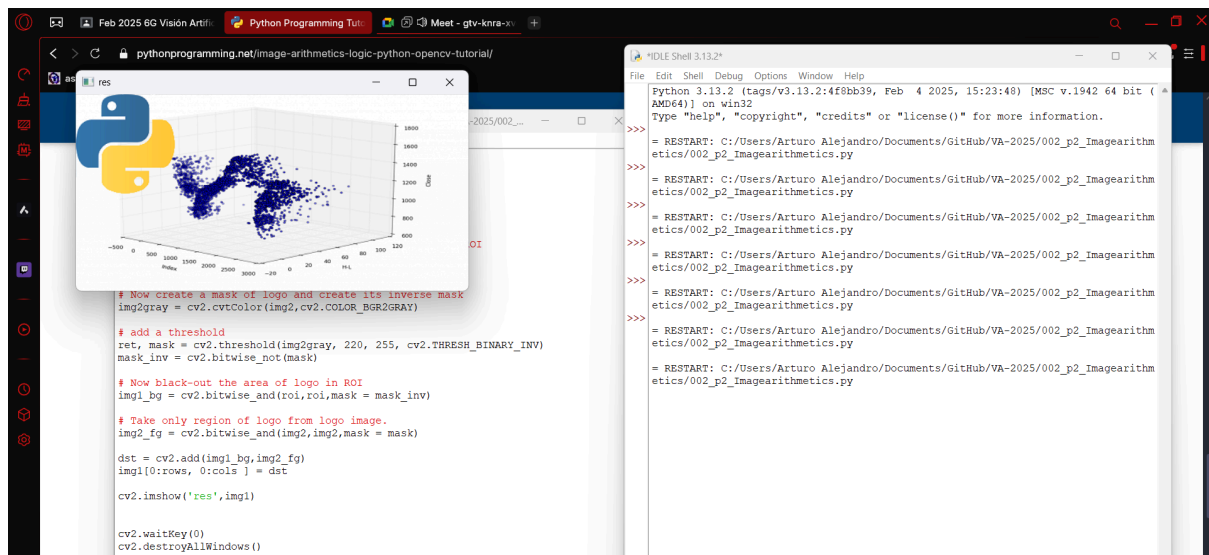
`cv2.destroyAllWindows()`

Resultados

Se visualizan imágenes combinadas de diferentes maneras: suma directa, combinación ponderada y superposición con máscara para eliminar el fondo no deseado.







Conclusión

Esta práctica muestra cómo realizar operaciones aritméticas y lógicas con imágenes en OpenCV, lo que es útil para superposición, mezcla y manipulación avanzada.