



[Practica 5]

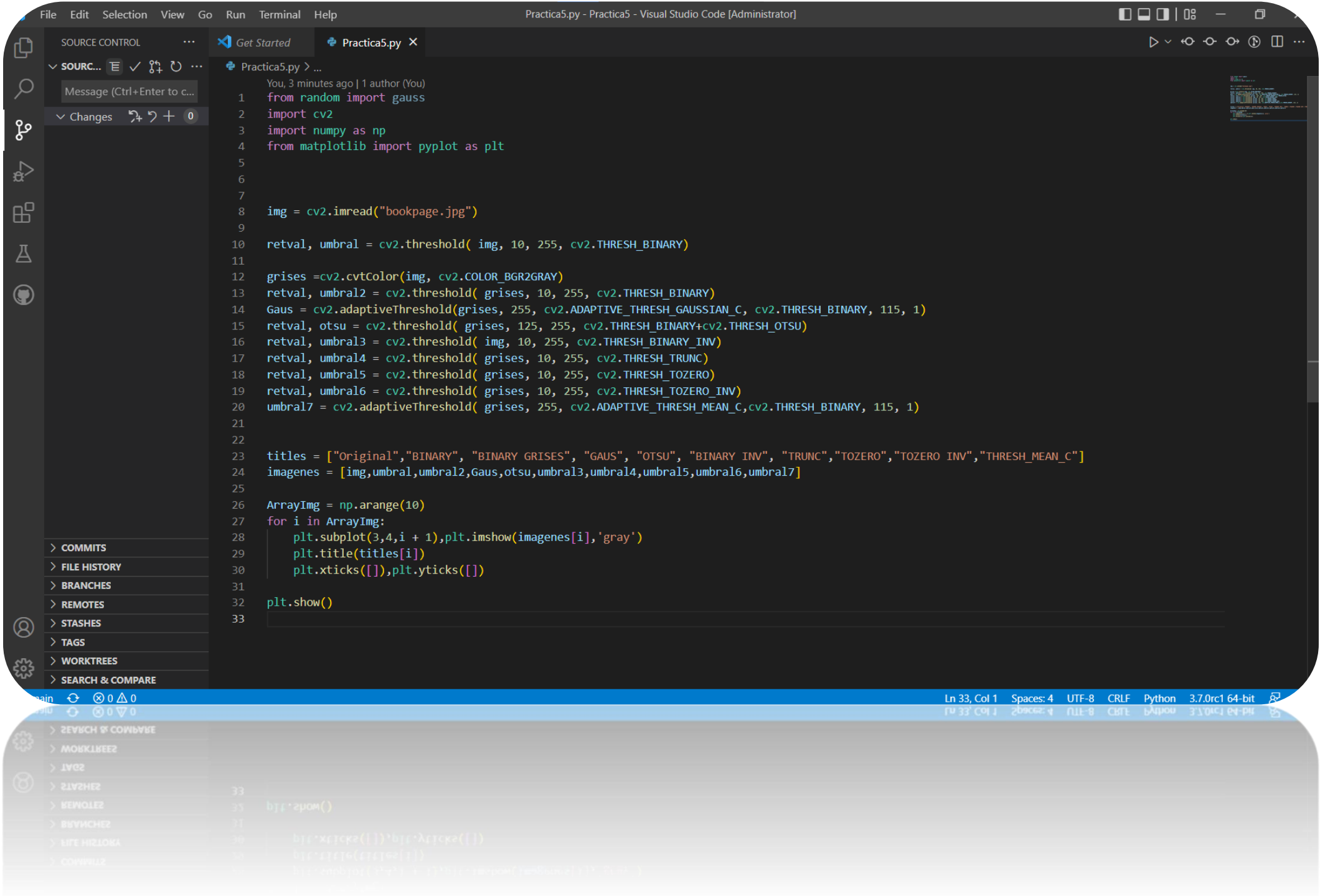
[Sistema de Visión Artificial]

Loera Contreras Gildardo Yair – 19110230 – 7E
PROFESOR [MAURICIO ALEJANDRO CABRERA ARELLANO]

PROBLEMA

Práctica 5 – Umbrales.
Objetivo: Utilizar las funciones de umbrales para la recuperación de información.
Threshold1 binary, b_inv, Trunc, To Zero, Tz_inv, Mean, Gaus, Otsu.

CODIGO



```
from random import gauss
import cv2
import numpy as np
from matplotlib import pyplot as plt

img = cv2.imread("bookpage.jpg")

retval, umbral = cv2.threshold( img, 10, 255, cv2.THRESH_BINARY)

grises =cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
retval, umbral2 = cv2.threshold( grises, 10, 255, cv2.THRESH_BINARY)
Gaus = cv2.adaptiveThreshold(grises, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C, cv2.THRESH_BINARY, 115, 1)
retval, otsu = cv2.threshold( grises, 125, 255, cv2.THRESH_BINARY+cv2.THRESH_OTSU)
retval, umbral3 = cv2.threshold( img, 10, 255, cv2.THRESH_BINARY_INV)
retval, umbral4 = cv2.threshold( grises, 10, 255, cv2.THRESH_TRUNC)
retval, umbral5 = cv2.threshold( grises, 10, 255, cv2.THRESH_TOZERO)
retval, umbral6 = cv2.threshold( grises, 10, 255, cv2.THRESH_TOZERO_INV)
umbral7 = cv2.adaptiveThreshold( grises, 255, cv2.ADAPTIVE_THRESH_MEAN_C,cv2.THRESH_BINARY, 115, 1)

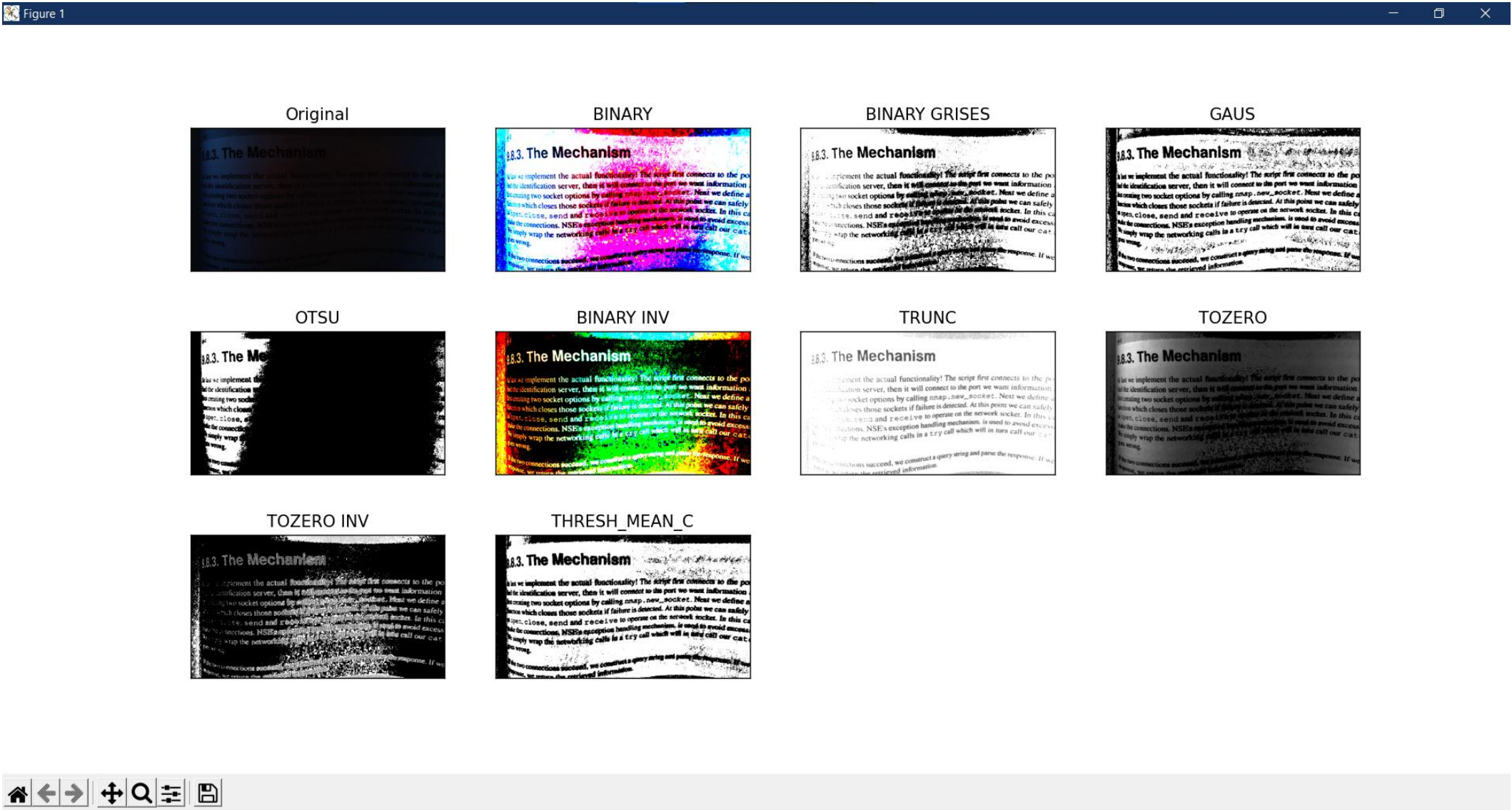
titles = ["Original","BINARY", "BINARY GRISES", "GAUS", "OTSU", "BINARY INV", "TRUNC","TOZERO","TOZERO INV","THRESH_MEAN_C"]
imagenes = [img,umbral,umbral2,Gaus,otsu,umbral3,umbral4,umbral5,umbral6,umbral7]
```

```
ArrayImg = np.arange(10)
for i in ArrayImg:
    plt.subplot(3,4,i + 1),plt.imshow(imagenes[i], 'gray')
    plt.title(titles[i])
    plt.xticks([],plt.yticks([]))

plt.show()
```

EVIDENCIAS

Al correr el programa nos mostrara el conjunto de imágenes en un Plot, con cada una de los diferentes umbrales que se utilizó para hacer la recuperación de la información de la imagen original.



GITHUB:
<https://github.com/Gylc87/Practica5.git>