



**INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO**

“Practica 3-Binarizacion”

-Tinoco Videgaray Sergio Ernesto

Grupo: 4BV1

Materia: Procesamiento digital de imágenes

INSTITUTO POLITÉCNICO NACIONAL



01/04/22

A la hora de procesar una imagen se requiere contrastar algún elemento dentro de la imagen permitiéndonos resaltar con mayor detalle los elementos que se encuentran dentro de la imagen.

Por ello existe el umbralado de imagen que nos permite obtener una imagen binarizada en únicamente 2 niveles de gris (blanco y negro) a partir de un valor que nos indicara el intervalo de grises que se van a contrastar.

En el desarrollo de esta práctica se va a realizar un programa en Python que obtendrá la imagen binarizada a partir de una imagen con elementos en blanco y negro para resaltar dichos elementos.

Para este caso vamos a trabajar con una imagen de las piezas de ajedrez.



- Desarrollo:

Primeramente, se van a importar las bibliotecas correspondientes:

Del paquete PILLOW se va a importar la biblioteca Image que nos va a permitir abrir un archivo de imagen de la computadora.

Se va a importar la Biblioteca Numpy asignándole como alias “np”.

De igual forma se va a importar la biblioteca Pyplot del paquete matplotlib para generar el grafico de dicha imagen.

```
from PIL import Image
import numpy as np
from matplotlib import pyplot
```

Posteriormente se va a abrir el archivo de imagen y se convierte a escala de grises:

```
img=Image.open("./ajedrez.png").convert("L")
```

Una vez convertida la imagen, se carga en una matriz usando la función "asarray" de la biblioteca Numpy

```
matriz=np.asarray(img)
```

Se obtienen los tamaños de alto y ancho de la imagen

```
alto=matriz.shape[0]  
ancho=matriz.shape[1]
```

Se solicita al usuario el nivel de umbral que se va a manejar:

```
m=int(input("Ingrese el valor del umbral: "))
```

Se recorre la matriz de la imagen evaluando cada nivel de gris presente en ella.

Si el nivel de gris es menor o igual al valor de umbral se asigna el nivel 0. De otra forma si resulta ser mayor a el valor de umbral se asigna el nivel mas alto, es decir 255.

```
i=0  
while(i<alto):  
    j=0  
    while(j<ancho):  
        if(matriz[i][j]<=m):  
            matriz[i][j]=0  
        else:  
            matriz[i][j]=255  
        j+=1  
    i+=1
```

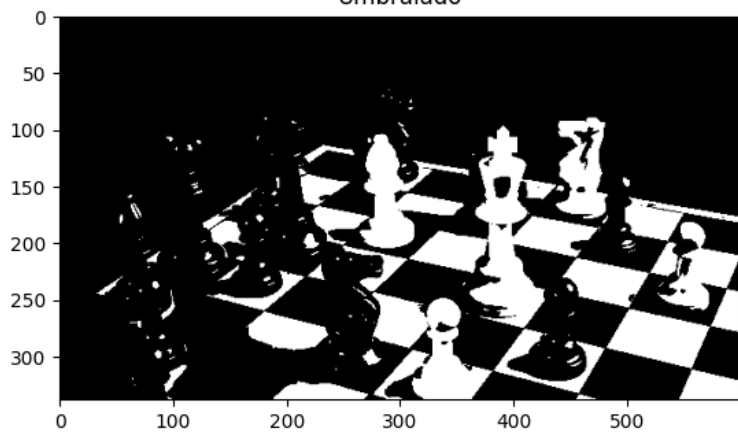
Cargamos los datos de la matriz resultante y lo mostramos por medio de un gráfico:

```
pyplot.imshow(matriz,cmap="gray")  
pyplot.show()
```

- Resultados:

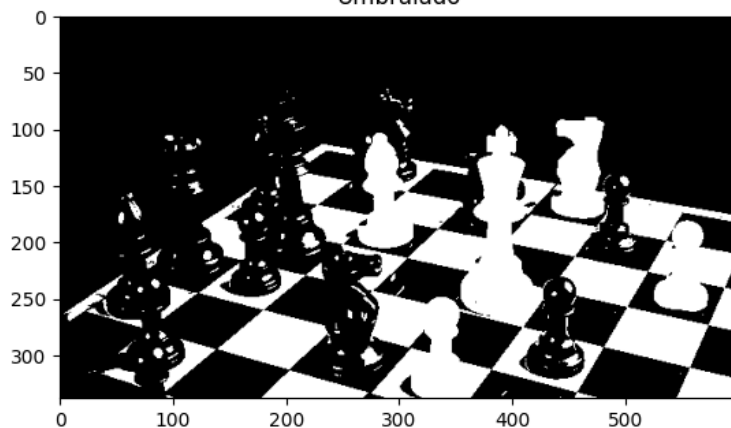
```
Ingrese el valor del umbral: 150
█
```

Umbralado



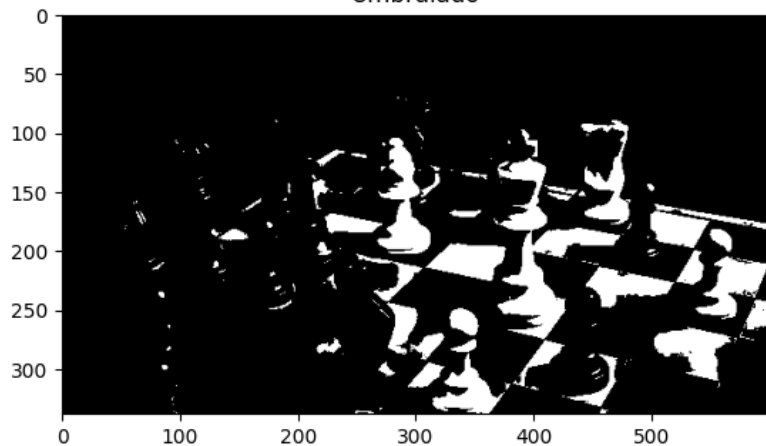
```
Ingrese el valor del umbral: 95
█
```

Umbralado



```
Ingrese el valor del umbral: 180
█
```

Umbralado



```
Ingrese el valor del umbral: 220
█
```

Umbralado

