

# Trabajo Práctico 1: Imágenes

## Objetivos:

- Hallar algoritmo para recorrer la imagen y encontrar la estrella más brillante.

Primero importamos las librerías correspondientes para procesar y analizar las imágenes

```
In [1]: %matplotlib inline

import numpy as np

import cv2 as cv

import matplotlib.pyplot as plt
```

## Primera imagen:

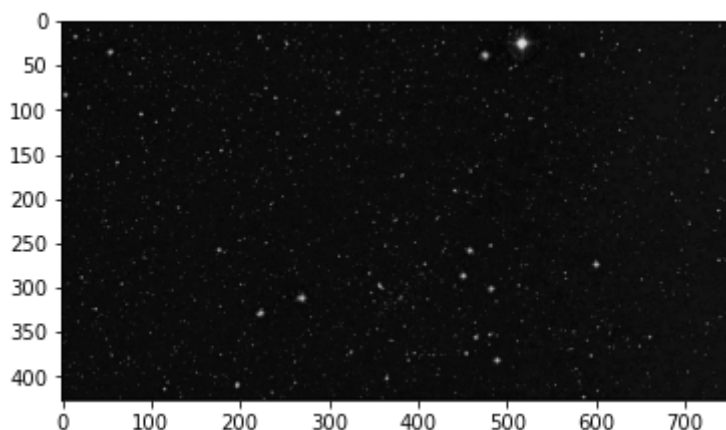
Comenzamos cargando y mostrando la primera imagen "Estrellas1.png" y su información.

```
In [2]: # Cargo la primera imagen
img1 = cv.imread('estrellas1.png', cv.IMREAD_GRAYSCALE)

# Dimensión de la imagen
print('Tamaño imagen: {} pix'.format(img1.shape))
print('Cantidad total de pixeles: {}'.format(img1.size))
print('Tipo de dato de cada pixel: {}'.format(img1.dtype))

plt.imshow(img1, cmap='gray')
plt.show()
```

Tamaño imagen: (426, 755) pix  
Cantidad total de pixeles: 321630  
Tipo de dato de cada pixel: uint8



Para poder cumplir la consigna pedida, definimos una función (ubicación\_pixel\_más\_brillante). Ésta función recibe como dato la imagen, luego define el punto  $a, b = 0, 0$ , la variable `pix`, y el alto y ancho de la imagen; recorre la imagen definiendo `pix_actual` y si `pix_actual` es mayor que `pix`, redefine a `pix = pix_actual`. Finalmente luego de recorrer toda la imagen devuelve la ubicación del pixel de mayor intensidad.

```
In [3]: #Funcion para buscar la ubicacion de la estrella mas brillante

def ubicacion_pixel_mas_brillante(img):
```

```

a,b = 0,0
pix = img.item(a,b) #intensidad del pixel
alto,ancho = img.shape

for i in range(0,alto):
    for j in range(0,ancho):
        pix_actual = img.item(i,j)

        if pix_actual > pix:
            pix = pix_actual
            a,b=i,j

return a,b,pix

```

Luego, llamamos a la función anteriormente definida, mostramos las coordenadas de su ubicación y la marcamos sobre una copia de la imagen con un círculo.

```

In [4]: #Busco la estrella mas brillante para la primer imagen
a,b,pix = ubicacion_pixel_mas_brillante(img1)

print("Ubicacion del pixel  a: ",a,"b: ",b)
print("Intensidad del pixel:",pix)

img_circle = img1.copy()
#cv.circle(imagen,centro,radio,color,ancho_linea,tipo_linea)
cv.circle(img_circle,(b,a),20,(255,255,255),2,cv.LINE_AA)

plt.title("Estrella mas brillante")
plt.imshow(img_circle, cmap='gray')

```

```

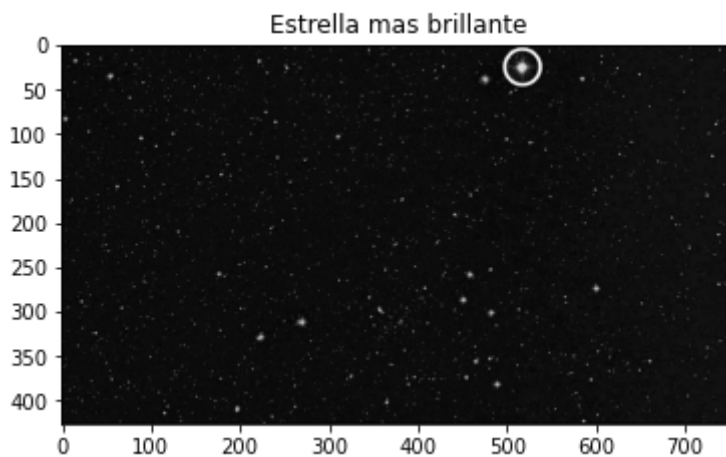
Ubicacion del pixel  a:  24 b:  518
Intensidad del pixel: 255

```

```

Out[4]: <matplotlib.image.AxesImage at 0x23f05ae5a60>

```



## Segunda imagen

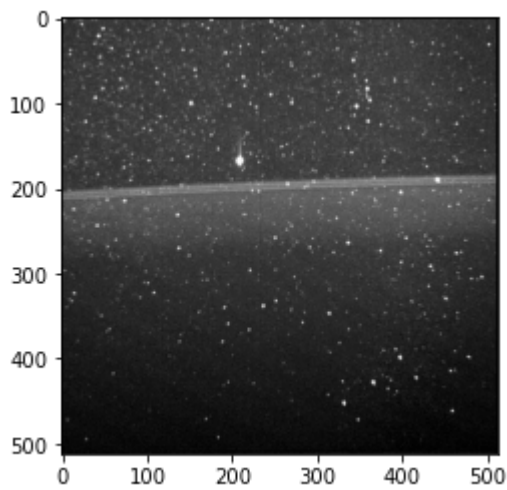
Repetimos el mismo proceso que para la primera imagen.

```

In [5]: # Cargo la segunda imagen
img2 = cv.imread('estrellas2.png', cv.IMREAD_GRAYSCALE)

plt.imshow(img2, cmap='gray')
plt.show()

```



```
In [6]: #Busco la estrella mas brillante para la segunda imagen
a,b,pix = ubicacion_pixel_mas_brillante(img2)

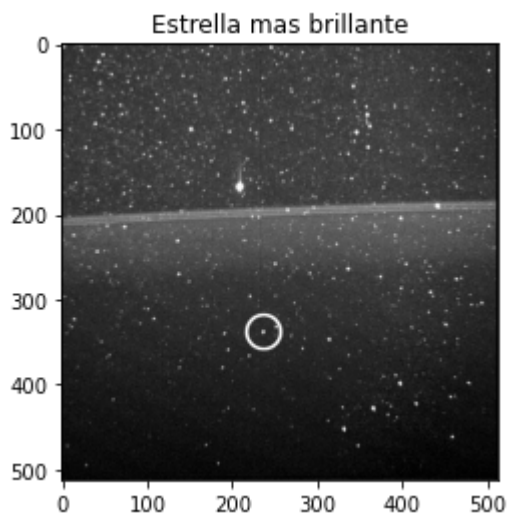
print("Ubicacion del pixel  a: ",a,"b: ",b)
print("Intensidad del pixel:",pix)

img_circle = img2.copy()
#cv.circle(imagen,centro,radio,color,ancho_linea,tipo_linea)
cv.circle(img_circle,(b,a),20,(255,255,255),2,cv.LINE_AA)

plt.title("Estrella mas brillante")
plt.imshow(img_circle, cmap='gray')
```

```
Ubicacion del pixel  a: 338 b: 237
Intensidad del pixel: 255
```

```
Out[6]: <matplotlib.image.AxesImage at 0x23f05e09d00>
```



## Análisis de resultados y conclusiones

Tanto como para la primera imagen y para la segunda logramos obtener la estrella más brillante y su ubicación. Sin embargo parecería a simple vista que para la primera imagen el resultado es el esperado, pero para la segunda no. Esto se debe a que para el ojo humano resalta más el conjunto de pixeles de alta intensidad, pero eso no implica que sea la más brillante puesto que con el análisis de datos se demuestra lo contrario; ya que en ambas imagenes la intensidad de pixel hallada es la máxima y su valor es de 255.