

# FUNDAMENTOS MATEMÁTICOS

## DE LA

### VISIÓN ROBÓTICA

## TRABAJO PRACTICO Nº 1: IMÁGENES

---

Vázquez Lareu, Román. Facultad de Ingeniería de la UBA

05/05/2020

### Introducción

En el siguiente informe, a partir de una imagen dada, se buscará obtener la estrella más brillante de la misma. Esto se hará generando un algoritmo que la recorra e identifique dicha estrella.

### Desarrollo

En primer lugar, se importarán las librerías necesarias para el análisis de la imagen: **numpy**, **cv2**, **matplotlib.pyplot**. A su vez, se evitará que las imágenes sean mostradas en una ventana emergente mediante el **"inline"**.

---

Listing 1: Importación de librerías

```
1 %matplotlib inline
2 import numpy as np
3 import cv2 as cv
4 import matplotlib.pyplot as plt
```

---

A continuación se leerá la imagen y se obtendrá su estructura (cantidad de canales y dimensiones; permitirán saber si se trata de una imagen a color o blanco y negro), cantidad total de píxeles y el tipo de cada pixel.

---

Listing 2: Carga de imagen y muestra de información

```
1 img = cv.imread('Estrellas.bmp')
2
3 print('Estructura de la imagen: {}'.format(img.shape))
4 print('Cantidad total de píxeles: {}'.format(img.size))
5 print('Tipo de dato de cada pixel: {}'.format(img.dtype))
6
```

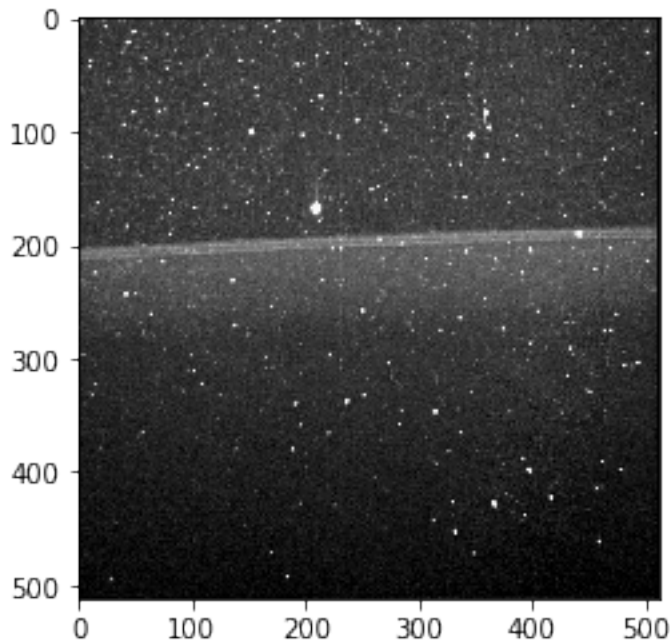
```
7 if len(img.shape)==3:
8     print('Imagen COLOR')
9 else:
10    print('Imagen BW')
11 plt.imshow(img[...,:-1])
```

---

Se obtendrá el siguiente output

---

```
1 Estructura de la imagen: (512, 512, 3)
2 Cantidad total de pixeles: 786432
3 Tipo de dato de cada pixel: uint8
4 Imagen COLOR
5 <matplotlib.image.AxesImage at 0x711f294d08>
6
7
```



Dado que se busca la estrella más brillante, se lee ahora la imagen en una escala de grises, con el objetivo de encontrar la "más blanca".

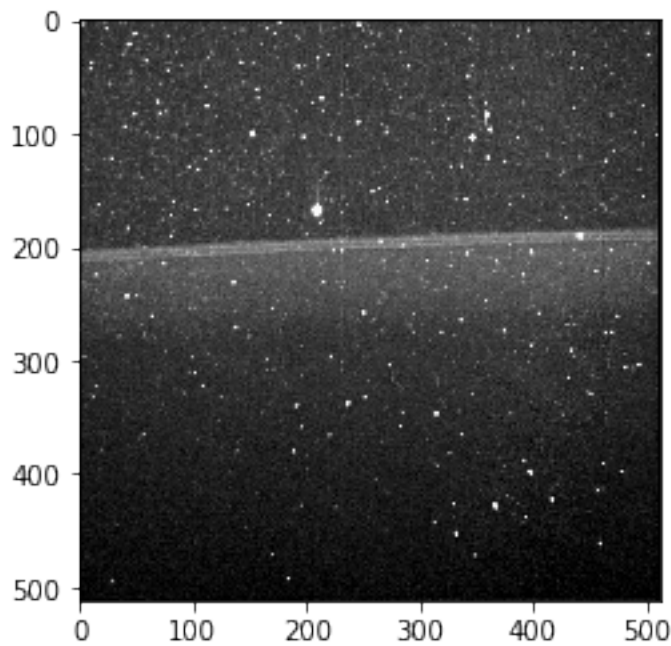
Listing 3: Lectura de imagen en escala de grises

---

```
1 imgGray = cv.imread('Estrellas.bmp',cv.IMREAD_GRAYSCALE)
2 plt.imshow(imgGray, cmap='gray', vmin=0, vmax=255)
3 plt.show()
```

---

Obteniendo de esta manera el siguiente output



10

---

La escala de grises en la que se basará al algoritmo al momento de hacer las comparaciones:



En cuanto al algoritmo, en primer se define una lista **posicionesMaximos** donde se almacenarán las posiciones de los máximos y una variable que indicará a que máximo hacen referencia esas posiciones **maximoValor**. La lista se inicializa vacía y el máximo en el valor mínimo posible.

---

Listing 4: Defino variables

```
1 posicionesMaximos = []  
2 maximoValor = 0;
```

---

A continuación se recorre la imagen usando ciclos anidados de **"for"** (fila y columna), y se

realizan dos comparaciones. En la primera, si el pixel es mayor al **maximoValor**, entonces se actualiza con el valor actual. A su vez se reinicia la lista y se almacena la posición actual. En la segunda, si el pixel es igual al **maximoValor**, únicamente se agrega la posición del maximo actual a la lista.

Listing 5: Algoritmo de recorrido de imagen

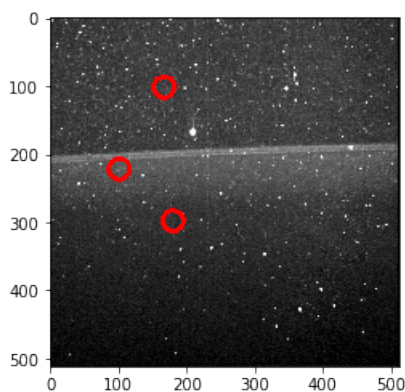
```
1 for fila in range(0,img.shape[0]):
2     for columna in range (0,img.shape[1]):
3
4         pixel = imgGray.item(fila,columna)
5
6         if (pixel == maximoValor):
7             posicion = (fila,columna)
8             posicionesMaximos.append(posicion)
9
10        if (pixel > maximoValor):
11            maximoValor = pixel;
12            posicionesMaximos = []
13            posicion = (fila,columna)
14            posicionesMaximos.append(posicion)
```

Finalmente, se recorre la lista **posicionMaximo**, y se dibuja un circulo alrededor de cada pixel obtenido como posicion de maximo

Listing 6: Marco máximos en la imagen

```
1 for posicion in posicionesMaximos:
2     cv.circle(img, posicion, 15, (0,0,255), thickness=5, lineType=8, shift↵
        =0)
3 plt.imshow(img[...,:-1])
```

La imagen final, con las estrellas mas brillantes marcadas es la siguiente:



11

---

## **Conclusión**

En conclusión, al momento de analizar las imágenes, a estas pueden tratarse como matrices de datos que pueden tener 2 (mono) o 3 (color) dimensiones.

## **Nota**

A simple vista creo no haber llegado al resultado esperado. Esto puede deberse a un error al momento de programar el algoritmo.