

# Introducción al procesamiento de Imágenes con Python y OpenCv

## Práctica 5

1ra parte

- LIBRERIA\_PYTHON → <https://pillow.readthedocs.io/en/stable/index.html>

```
from PIL import Image as im
import matplotlib.pyplot as plt
import numpy as np
I = im.open("alpes.png")
I2 = im.open("alpes2.jpg")
I3 = im.open("Hawaii.jpg")
I2 = I2.convert("L")
I33 = I3.convert("L")
I.show()
I3.show()
I33.show() #Mostrará en la pantalla la imagen seleccionada
```

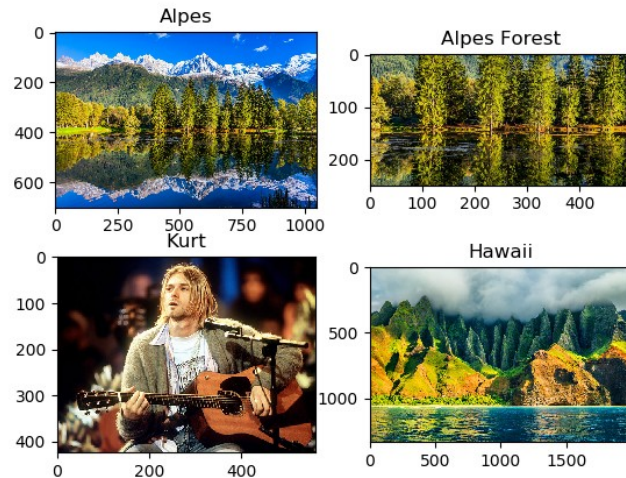
#Con los siguientes comandos, se imprimirá en la pantalla de la consola, las #características de cada imagen, como la resolución en pixeles, el tipo de escala de #color que tiene, siendo L para escala a grises y RGB a color, y el path o formato de imagen.

```
print("alpes.png -> ", I.size, I.mode, I.format)
print("alpes2.jpg -> ", I2.size, I2.mode, I2.format)
print("Hawaii.jpg -> ", I3.size, I3.mode, I3.format)
```

Debemos considerar la ruta de la imagen previa a su nombre si en el directorio en donde usted esté creando el Script correspondiente no se encuentre dicha imagen, de lo contrario, solo bastará con poner el nombre de la imagen con su path o extensión.

#Convertimos las imágenes en arrays de datos  
#(a,dtype,order,like)

```
a = np.asarray(I, dtype = np.uint8)
b = np.asarray(I2, dtype = np.uint8)
c = np.asarray(I3, dtype = np.uint8)
d = a[251:283, 317:349] #Tomamos un segmento
de la imagen.
```



#Creación de matriz de imágenes por medio de la librería matplotlib

```
plt.figure() #Crea una figura nueva
```

```
plt.subplot(221) #Agrega un eje a la figura actual
plt.imshow(a,cmap="gray", interpolation='nearest')
#Muestralos datos como imagen o como un raster en 2D
plt.title("kurt") #Agregamos un título a la imagen
```

```
plt.subplot(222)
plt.imshow(d,cmap="gray", interpolation='nearest')
plt.title("kurt_guitar")
```

```
plt.subplot(223)
plt.imshow(b,cmap="gray", interpolation='nearest')
plt.title("Alpes")
```

```
plt.subplot(224)
plt.imshow(c,cmap="gray", interpolation='nearest')
plt.title("Hawaii")
```

```
plt.show() #Muestra todas las gráficas/imágenes abiertas o creadas
```

# Extracción de contornos a imágenes con Python

## ¿Cómo extraer contornos a objetos de una imagen determinada con filtros en Python?

- Usted puede elegir la imagen que desee siempre y cuando esta sea lo suficientemente nítida para procesar.
- Evitar imágenes demasiado oscuras o con un alto contraste o brillo.
- Se aplicarán diferentes tipos de filtros a la misma imagen, y deberá comentar al respecto qué puede notar de diferente en cada uno, así como las posibles utilidades de estos cuando se trata de procesar imágenes geográficas.
- En esta práctica se empleará la librería cv2 (mejor conocida como openCV). Investigue como instalarla y su utilidad.

## LINKS DE INTERÉS

- <https://www.inesem.es/revistadigital/informatica-y-tics/opencv/>
- <https://pypi.org/project/opencv-python/>
- <https://opencv.org/>
- <https://pillow.readthedocs.io/en/stable/index.html>