

# Curso de imágenes y colores

A) El color es completamente un fenómeno de tu ojo

## 1. Introducción

### 1.1. Gráficos de computadora

1. Gráficos raster
2. Gráficos vectoriales

### 1.2. Imagen digital

1. Pixel [5.1](#)
2. Color o intensidad
3. Tamaño de la imagen [3.4](#)
4. Resolución

## 2. Librerías principales de Python

1. numpy
2. matplotlib
3. skimage

## 3. Librería numpy

B) ¿Por qué no una lista de Python?

### 3.1. Posición de numpy en Python

1. numpy proporciona una plataforma para manejar Big Data de manera eficiente
2. numpy en el stack científico de Python
3. Cálculos multi-dimensionales
4. Las matrices son una subclase de los arreglos

### 3.2. Creación de 1D ndarrays

1. `arange()` incluye el paso
2. `linspace()` incluye número de puntos
3. Vectorizar una función
4. `array()` vector de columna
5. `zeros()` y `ones()`



### 3.3. Creación de 2D ndarrays

1. reshape()
2. concatenación vertical y horizontal
3. convol2.py
4. Evaluar una función en una cuadrícula 2D [7](#)

### 3.4. Atributos de ndarrays

1. a.shape
2. a.size
3. a.itemsize
4. a.dtype
5. a.nbytes

### 3.5. Información de ayuda o help

1. Usando auto-completando
2. np.arange? np.linspace? [5.3](#)

### 3.6. Navegación de índices

1. Sistema de coordenadas de imagen
2. Manejo de índices [7](#)

## 4. Librería matplotlib

### 4.1. Varios gráficos en una figura

1. Dos interfaces diferentes para graficar
2. Ambas interfaces son OOP
3. Interface pyplot
4. ¿Qué es una figura?
5. ¿Qué es un subplot? [5.3](#)
6. Un grafico en un solo subplot
7. scatter - line\_with\_noise.py
8. Múltiples gráficos en el mismo plot()
9. Etiqueta y leyenda
10. Múltiples plot() en una figura



## 4.2. Leer una foto

1. `imread()`
2. Dimensiones de una foto
3. Canales de colores

## 4.3. Graficar la foto

1. Escoger un canal
2. `plt.imshow()`
3. `plt.colorbar()`
4. `plt.show(block=False)`

## 4.4. Elementos artísticos

1. `ax.legend()`
2. `Mathtext` in Matplotlib
3. `ax.text(0.70, 0.25, ecuación)`

# 5. Librería skimage

## 5.1. Analizar imágenes

1. `analizar-imagenes.py`
2. color a gris `rgb2gray()`
3. Convertir gris a color
4. `imsave()`

## 5.2. Transformar imágenes

1. `transform-images.py`
2. Compara imágenes al cambiar tamaño
3. `rescale`, `resize`, `downscale_local_mean`

## 5.3. Detectar bordes

1. `roberts?`
2. Diferentes filtros
3. `edge-detection.py`
4. `canny-detection.py`



## 5.4. Suavizar y convolucion

1. suavizar-filtrar.py
2. from skimage import filters
3. from scipy.ndimage import convolve

## 5.5. Deconvolucion

- 1.
- 2.

C) El arte y las Matemáticas te enseñan

# 6. Color

## 6.1. Historia del color

1. Desde Maxwell a Cruz Diez
2. Entender como se forma el color

## 6.2. Espacios de colores

1. Codificación de colores RGB
2. Código hexadecimal usando GIMP
3. Color en una etiqueta

## 6.3. Matiz, saturación, intensidad

1. complex-image.py

# 7. Crear imágenes

1. Crear imágenes virtuales
2. universe.py
3. field.py
4. Aplicar pseudo-colores

## 7.1. Conjunto de Mandelbrot

1. Crear una imagen del conjunto de Mandelbrot
2. Función de Mandelbrot
3. Recorte un pequeño cuadro de la imagen



## 8. Gráficos 3D y Animación

1. Gráficos 3D
2. Animación

## 9. Espectro de una imagen

- 1.
- 2.

## 10. Imágenes en medicina

- 1.
- 2.

