

# **Manipulación de imágenes, DICOM y color**

## **Laboratorio 1**

Dr. Alejandro Veloz

### **Objetivos**

Los objetivos de esta actividad son los siguientes:

- Familiarizarse con las bibliotecas NumPy, SciPy, y Matplotlib.
- Familiarizarse con el manejo básico de imágenes digitales.
- Familiarizarse con el estándar DICOM, utilizado para transmitir, almacenar, recuperar, imprimir, procesar y mostrar información relativa a las imágenes médicas.
- Aprender a utilizar la biblioteca pydicom.

### **Actividades**

1. Trabajo complementario (opcional)
2. DICOM
3. Mejora de contraste
4. Filtrado espacial
5. Visualización de mapas de activación
6. Instrucciones para la entrega

### **1. Trabajo complementario**

#### **1. Trabajo complementario**

- Instale la plataforma de desarrollo Anaconda, disponible de forma gratuita en [LINK](#). También puede usar Google Colab o Visual Studio Code.

- Estudie y haga los ejercicios de las siguientes secciones del documento “Scipy Lecture Notes”:
  - Sección 1.4. NumPy: creating and manipulating numerical data.
  - Sección 1.5. Matplotlib: plotting.
  - Secciones 1.6.1, 1.6.3, 1.6.5, 1.6.6 y 1.6.10. Scipy : high-level scientific computing.

## 1. Trabajo complementario

- Leer el siguiente material de referencia:
  - Why Does the DICOM Standard Exist?
  - Parte 1 “Introduction and Overview” del estándar DICOM.
  - Documentación de la biblioteca PyDicom.

## 2. DICOM

### 2. DICOM

- Se trabajará con dos conjunto de datos.
  - **data1**: imágenes de CT de distintos pacientes.
  - **data2**: Imágenes de CT de cortes de un mismo paciente.
- Los datos están disponibles en Github.

```
!git clone https://github.com/aavelozb/bme423-data.git
!unzip bme423-data/data1.zip -d bme423-data/
!unzip bme423-data/data2.zip -d bme423-data/
```

### 2. DICOM - actividades

1. Desarrolle un programa que genere un archivo CSV con los nombres y fechas de nacimiento de los pacientes de las imágenes contenidas en el directorio **data1**.
2. Desarrolle un programa que muestre en una sola figura todas las imágenes del directorio **data1**. Cada imagen debe estar identificada por el nombre del paciente.
3. Desarrolle un programa que anonimice todas las imágenes del directorio **data1**.
4. Implemente un programa que permita visualizar un corte coronal y sagital de las imágenes del directorio **data2**.

## 2. DICOM - actividades

5. Implemente un programa que permita visualizar un corte oblicuo coronal-sagital de las imágenes del directorio **data2**.
6. El proceso de *downsampling* consiste en re-muestrear una imagen. La forma más simple de realizar este proceso es conservar uno de cada  $n \in \mathbb{N}$  píxeles de la imagen.
  - Realice el proceso de *downsampling* de la imagen **p5.dcm**, conservando uno de cada 2, 3 y 4 píxeles.
  - Muestre la imagen original y las de tamaño reducido. Analice las diferencias entre éstas.

## 2. DICOM - actividades

7. Visualice las imágenes del directorio **data2** usando widgets de [Matplotlib](#), o [ipywidgets](#).

## 3. Contraste

### 3. Contraste - actividades

8. Agregue opciones a un widget para aplicar operaciones de mejora de contraste, en base a la transformación  $\gamma$  (Ec. 3.2-3, sección 3.2.3, página 110) y usando una función de transformación definida por tramos (Figura 3.10(a), página 116).
9. Genere situaciones que permitan apreciar en qué situaciones el contraste mejora, versus cuándo empeora. Se recomienda usar los histogramas de las imágenes resultantes, junto a los histogramas de las imágenes de entrada para su explicación.

### 3. Contraste - actividades

10. Implemente la ecualización (Ec. 3.3-8, página 126) y especificación de histogramas (Ec. 3.3-12, página 130). Explique los resultados (no es necesario usar widgets).

## 4. Filtrado espacial

La razón señal-ruido (SNR) es un indicador que permite averiguar el efecto de aplicar un filtro de suavizado. Consideraremos la siguiente fórmula para calcular la SNR:

$$SNR = \frac{\mu_{\text{sig}}}{\sigma_{\text{sig}}},$$

donde  $\mu_{\text{sig}}$  y  $\sigma_{\text{sig}}$  son el promedio y la desviación estándar, respectivamente, de las intensidades de una región de interés homogénea, seleccionada manualmente.

#### 4. Filtrado espacial - actividades

11. Implemente la operación de convolución (sección 3.4.2, página 148, ecuación 3.4-1).
12. Usando su implementación de la convolución:
  - Implemente los filtros de suavizado, incluyendo el filtro Gaussiano, (sección 3.5.1, página 152) y agudizado (sección 3.6, página 157).
  - Aplique estos filtros al volumen de CT de una misma persona (**data2**). Aplique la convolución en 2D y luego en 3D.
  - Compare con el resultado obtenido usando la función `scipy.ndimage.convolve` (para un filtro).

**Ayuda:** La siguiente sentencia permite construir la versión rellenada con ceros de la imagen:

```
pad_len = (mask.shape[0] - 1) // 2
img_padded = np.pad(img, ((pad_len, pad_len), (pad_len, pad_len)),
                     'constant', constant_values=0)
```

donde `mask` contiene la máscara del filtro (se asume que es cuadrada, con largo impar), e `img` es la imagen original.

#### 4. Filtrado espacial - actividades

13. Aplique los filtros de suavizado, filtros de agudizado y operaciones de mejora de contraste *de manera combinada* en la imagen **xray.dcm**.
  - Explique la pertinencia de los pasos aplicados y cuantifique la mejora en la calidad de la imagen a medida que aplica los pasos considerados.
  - Use el ejemplo de la sección 3.7 (pág. 169) como inspiración.
14. Resuelva el mismo problema de la pregunta anterior para el caso 3D con el volumen **data2**.

#### 4. Filtrado espacial - actividades

**Nota:** La operación de convolución, los filtros, y los métodos de mejora de contraste debe implementarlos Ud., sin utilizar librerías específicas de procesamiento de imágenes.

## 5. Visualización de mapas de activación

### 5. Visualización de mapas de activación - actividades

15. Desarrolle un programa que permita visualizar un mapa como el siguiente:

Dispone de los archivos **anat.nii**, **zmap\_voice.nii.gz** y **zmap\_nonvoice.nii.gz**.

Note que los archivos están en formato *.nii*, los cuales pueden ser leídos usando la librería **nibabel**.

## 5. Instrucciones para la entrega

**Entregas:**

- *apellido\_nombre\_l1e1.ipynb* y *apellido\_nombre\_l1e1.pdf*
  - Fecha: **hoy**
  - Respuestas desde la 1 hasta la 7.
  - Medio: Aula virtual
- *apellido\_nombre\_l1e2.ipynb* y *apellido\_nombre\_l1e2.pdf*
  - Fecha: 10/09/2025
  - Respuestas desde la 8 en adelante.
  - Medio: Aula virtual

Las versiones **.pdf** en las entregas corresponden a los archivos **.ipynb** con las salidas de las celdas visibles.

No respetar las instrucciones significará un descuento de 10 pts.