

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

Manejo Básico de Imágenes con MATLAB

N° de práctica: 2

Nombre completo del alumno		Firma
N° de cuenta:	Fecha de elaboración:	Grupo:

Elaborado por:	Revisado por:	Autorizado por:	Vigente desde:
Dr. Ernesto Moya Albor			Agosto 2015

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

1. Seguridad en la ejecución

	Peligro o Fuente de energía	Riesgo asociado
1		
2		
3		

2. Objetivo

Con el desarrollo de esta práctica el alumno conocerá las funciones básicas de *MatLab*[®] para abrir, desplegar y guardar imágenes en distintos formatos.

3. Introducción

❖ Marco teórico

MatLab[®] incluye un conjunto de funciones (algoritmos) y herramientas gráficas enfocadas al procesamiento, análisis y visualización de imágenes, este conjunto es denominado *toolbox* de imágenes. Muchas de las funciones del *toolbox* de imágenes se encuentran escritas en el lenguaje de *MatLab*[®], lo que permite inspeccionar los algoritmos usados.

En esta práctica se tocarán los puntos de entrada, salida y visualización de imágenes en diferentes formatos usando las funciones del *toolbox*.

Representación de los Tipos de imágenes en MatLab[®]

MatLab[®] soporta varios tipos de imágenes, las cuales se enumeran a continuación:

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

- imágenes Indexadas:

Consisten de una matriz de datos y una paleta de colores. La paleta de colores es un arreglo de m filas por 3 columnas la cual contiene valores de punto flotante en el rango de $[0,1]$. Cada fila de la paleta especifica las componentes roja, verde y azul para un color único. El color de cada pixel es determinado usando el correspondiente valor en la matriz de datos como un índice dentro de la paleta asignada. El valor 1 apunta a la primer fila de la paleta, el valor 2 a la segunda fila, etc.

- Imágenes de intensidad:

Consisten de una matriz de datos, cuyos valores representan intensidades dentro de algún rango, cada elemento en la matriz corresponde a un pixel en la imagen representando niveles de gris, donde la intensidad 0 usualmente representa al negro y la intensidad 1, 255 o 65536 usualmente representa total intensidad, o blanco.

- Imágenes RGB o a color:

Una imagen RGB, a veces es referida como una imagen de color verdadero, en *MatLab*[®] es almacenada como una matriz de $m \times n \times 3$ (m filas por n columnas y una profundidad de 3) que define las componente roja, verde y azul para cada pixel individual. Las imágenes RGB no usan paleta de colores, el color de cada pixel es determinado por la combinación de las intensidades roja, verde y azul almacenadas en cada plano de color en la ubicación del pixel. Los formatos gráficos almacenan las imágenes RGB como imágenes de 24 bits, donde las componentes roja, verde y azul son cada una de 8 bits, esto da un potencial de 16 millones de colores. Un pixel cuyas componentes de color son (0,0,0) es desplegado como negro y un pixel con las componentes (1,1,1) es desplegado como blanco.

❖ Conceptos clave

Abrir imágenes con MatLab[®]

Una imagen digital es un mosaico de elementos llamados pixeles, dichos elementos se encuentran almacenados en archivos digitales en un determinado formato definido por el tipo de imagen.

Se consideran las imágenes que estan almacenadas en un archivo con un formato determinado, por ejemplo: BMP, JPG, TIF, PNG, GIF, etc.

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

MatLab[®] posee la función **imread** encargada de abrir imágenes de diferentes formatos dentro del **workspace**, su sintaxis básica es la siguiente:

```
A=imread('nombre_imagen', 'tipo_imagen')
```

donde:

A.- Es el nombre de la variable donde se guardaran los valores de los pixeles de la imagen.

nombre_imagen.- Especifica el nombre completo del archivo que contiene a la imagen, puede incluir la ruta relativa o absoluta del fichero.

tipo_imagen.- Indica el formato de la imagen, por ejemplo: bmp, gif, jpg, png, tiff, etc. Se usa para forzar la lectura de un formato en particular sin importar la extensión del archivo.

Para el caso de las imágenes indexadas *MatLab*[®] regresa dos valores, uno para la matriz de valores y otro para la paleta o mapa de colores asociado:

```
[A, mapa]=imread('nombre_imagen', 'tipo_imagen')
```

Visualizar imágenes con MatLab[®]

Para desplegar en pantalla los datos contenidos en una matriz se utiliza el comando **image**, además para interpretar de manera correcta los datos para el caso de imágenes de intensidad e indexadas se usa el comando **colormap**, a continuación se muestra la sintaxis de ambas funciones:

```
figure(1); image(A); colormap(mapa)
```

donde:

1.- Indica el número de ventana donde se visualizará la imagen.

mapa.- Es el arreglo de m filas por 3 columnas que contiene la paleta de colores que interpretará los colores de la imagen.

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

Histograma de una imagen con MatLab®

El *histograma de una imagen* es una herramienta visual de gran aceptación y utilidad para el estudio de imágenes digitales. Con una simple mirada puede proporcionar una idea muy aproximada de la distribución de niveles de gris, el contraste que presenta la imagen y alguna pista del método más adecuado para manipularla.

La *forma* del histograma proporciona informaciones importantes como la *intensidad media* y la *dispersión* de los valores de nivel de gris, siendo esta última, la medida de *contraste* de la imagen. Cuanto mayor es la dispersión a lo largo del eje de los niveles de gris, mayor es el contraste de la imagen y es entonces cuando el sistema visual humano consigue una máxima respuesta en su apreciación de la imagen.

El histograma de una imagen digital con L niveles de gris en el rango $[0, L-1]$ es una función discreta de la forma:

$$h(r_k) = n_k$$

donde:

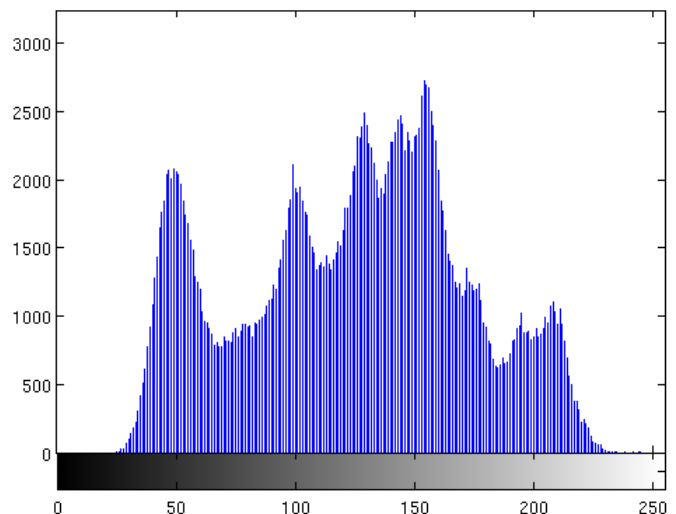
r_k es el k -ésimo nivel de gris

n_k es el número de píxeles en la imagen con el nivel de gris r_k

$k = 0, 1, 2, \dots, L-1$ niveles de gris

Para desplegar en pantalla el histograma de una imagen se utiliza el comando **imhist**:

```
figure(2); imhist(A)
```



	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

4. Ejercicios a realizar

❖ Ejercicio 1

- 1) Leer y desplegar la imagen del archivo “**corte.bmp**” así como su histograma.
- 2) Leer la imagen del archivo “**rx cerv.pcx**” y desplegarla usando primero el comando **image** y posteriormente el comando **imshow**.
- 3) Leer y desplegar la imagen “**abdomen.png**” usando el comando **image** y usando posteriormente el comando **imagesc**.
- 4) Abrir y visualizar la imagen RGB “**magriclonRGB.jpg**” y cada una de sus componentes de color por separado usando la paleta de colores roja, verde y azul.
- 6) Abrir y desplegar la imagen RGB “**retinaRGB.jpg**” y convertirla a niveles de gris.
- 7) Visualizar la imagen “**intestinoRGB.jpg**” con diferentes paletas de colores predefinidas de *MatLab*®.
- 10) Visualizar la imagen indexada “**cta_scan_index.bmp**”.

❖ Ejercicio 2

Histograma de una imagen de 8 bits

Implementa una función que calcule y muestre el histograma de imágenes de 8 bits con la siguiente sintaxis:

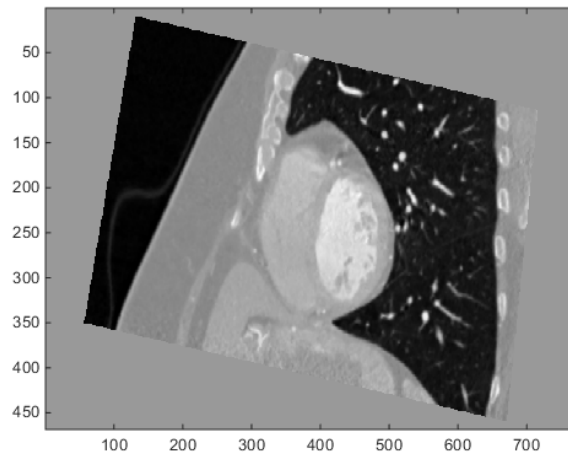
```
myhist(A);
```

	<h1 style="text-align: center;">Manual de Prácticas Procesamiento Digital de Imágenes Médicas</h1>	
División de Ingeniería Eléctrica	Laboratorio de Cómputo para el Procesamiento de Señales	

❖ Ejercicio 3

Imágenes de CT

Se cuenta con 11 imágenes (carpeta corte_45) de tomografía por computadora (CT) del **corte 45** del eje corto de un estudio del corazón, cada imagen representa un tiempo del ciclo cardíaco, del 0% al 90%.



a) Realiza un script que lea las 11 imágenes y muestre en una misma ventana las 11 imágenes de tal manera que se pueda simular el movimiento del corazón. Utiliza ciclos while o for y el comando pause.

5. Referencias

- ❖ Digital Image Processing, González, R.C , Woods, P., Addison Wesley, 1992
- ❖ MATLAB Documentation:
<http://www.mathworks.com/help/matlab/>