

## PRÁCTICA 1

### “Introducción al tratamiento de imágenes digitales en Matlab”

#### Objetivos.

Familiarizarse con el entorno, la programación y el tratamiento matricial de imágenes en Matlab.

#### Documentación y material disponible:

##### 1. Sobre Matlab

- a. *Apuntes Breves Notas de Matlab.*
- b. *Aprenda Matlab 7.0 como si estuviera en primero.*
- c. [www.mathworks.com](http://www.mathworks.com)
- d. Ayuda de la herramienta

##### 2. Sobre Image Processing

- a. Ayuda de la herramienta

**Imágenes a utilizar:** “P1\_1.jpg”.

**Instrucciones a utilizar:** `imfinfo`, `whos`, `imread`, `max`, `imtool`, `imshow`, `imwrite`, `stem`, `imhist`

#### Ejercicios

1. Sigue el documento ***Breves notas de Matlab*** y ejecuta sus ejemplos para familiarizarte con el entorno y la programación en Matlab.
2. Crea una carpeta en el escritorio donde guardarás todos los ficheros de la sesión. Selecciona esta carpeta como directorio de trabajo de Matlab.
3. Guarda en el directorio de trabajo un fichero de imagen.

4. Con la instrucción `imfinfo` de Matlab obtén la siguiente información de la imagen: anchura en píxeles, altura en píxeles, tipo de imagen y profundidad de bit.
5. Lee la imagen del fichero anterior y guárdala en una variable de Matlab (matriz) de nombre `Imagen1`. Para ello, haz uso de la instrucción `imread`.
6. Visualiza esta imagen con la instrucción `imtool` y con la instrucción `imshow`. Familiarízate con los entornos gráficos de salida de cada una de estas instrucciones.
7. Con la instrucción `whos` obtén la siguiente información de la variable Matlab `Imagen1`: tipo de dato y rango. Compara estos valores con los obtenidos en el ejercicio 4. Calcula el mayor elemento de la variable `Imagen1` (máximo nivel de intensidad). Utiliza para ello la instrucción `max`.
8. Calcula en Matlab la imagen complementaria de `Imagen1`, denominándola `Imagen2`. Visualiza esta imagen con la instrucción `imtool`. Guarda esta imagen en un fichero de imagen del mismo formato que la imagen original empleando la instrucción `imwrite`.
9. Crea una matriz, de nombre `Imagen3`, con los niveles de rojo de la imagen `Imagen1`. Esta nueva matriz es una imagen en niveles de gris. Visualízala con `imtool`. Guárdala en un fichero de imagen de otro formato (.bmp o .jpg) y lee la información correspondiente a este fichero.
10. Utiliza la función `imadjust` con la configuración `ImagenSalida=imadjust (ImagenEntrada, [], [], gamma)` para, mediante la modificación del parámetro `gamma`, obtener una imagen `Imagen4` más clara y una imagen `Imagen5` más oscura que `Imagen3`. Visualiza estas imágenes. Representa el histograma de estas tres imágenes mediante la función Matlab `imhist`. Interpreta cualitativamente los resultados de la operación realizada.
11. Utiliza la función `imabsdiff`, para crear una nueva imagen `Imagen6` que refleje la diferencia absoluta de `Imagen4` e `Imagen5`. Interpreta los resultados. Realiza la

misma operación sin utilizar la función `imabsdiff` y comprueba que obtienes los mismos resultados.

12. A partir del fichero de imagen inicial, desarrolla un script en Matlab que:
  - a. Calcule el histograma de una imagen en niveles de gris formada por los niveles de verde de la imagen inicial. El histograma se debe almacenar en un vector.
  - b. Visualice el histograma anterior mediante la instrucción `stem`.
  - c. Calcule de nuevo el histograma mediante la función Matlab `imhist`.
  - d. Compruebe que los dos resultados son idénticos.