



HERRAMIENTAS COMPUTACIONALES **EN EL LABORATORIO**

Parte 2: MATLAB **Examen de Febrero de 2010**

Escribir un programa Matlab que implemente un filtro de búsqueda de bordes en una imagen. El programa deberá leer una imagen guardada en un fichero, modificarla para resaltar los bordes y representar ambas imágenes (la original y la filtrada).

Cada imagen se puede entender como una matriz de píxeles, donde cada valor de la matriz corresponde a un número entre 0 y 255 que representa el nivel de gris de cada píxel. Así por ejemplo, una imagen de 20x50 píxeles (20 píxeles horizontales y 50 píxeles verticales) se almacenará en una matriz de 50 filas y 20 columnas.

Se suministran ya implementados los siguientes comandos, que nos permiten leer una imagen de un fichero o pintar imágenes en una ventana:

```
leeImagen(nombre)
% Lee la imagen contenida en el fichero cuyo nombre se pasa como parámetro.
% El comando devuelve una matriz con los valores de cada píxel de la imagen.
% Ejemplo: A=leeImagen('diatoms.jpg');

pintaImagenes(Imagen1, Imagen2)
% Crea una ventana donde pinta las imágenes contenidas en las dos matrices que
% se pasan como parámetros. Pinta la Imagen1 a la izquierda y la Imagen 2 a la
% derecha.
% Ejemplo: pintaImagenes(A,B);
```

Además, se suministran las siguientes imágenes para probar el programa:

- screw.jpg 30 x 30 píxeles
- diatoms.jpg 512 x 486 píxeles
- fibers.jpg 394 x 297 píxeles
- screws.jpg 524 x 400 píxeles
- spain.jpg 400 x 280 píxeles
- tinoncarbon.jpg 325 x 321 píxeles

El algoritmo de búsqueda de bordes se basa en la aplicación de dos filtros alrededor de cada píxel de la imagen original. Cada píxel de la imagen filtrada se calcula como una función de los 8 píxeles que tiene alrededor en la imagen original, según los siguientes pesos:

$$M_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \text{y} \quad M_2 = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Es decir, el nuevo píxel se calcula en función de los píxeles originales a partir de:

$$f_1 = a_{i-1,j-1} + 2a_{i-1,j} + a_{i-1,j+1} - a_{i+1,j-1} - 2a_{i+1,j} - a_{i+1,j+1}$$

$$f_2 = a_{i-1,j-1} + 2a_{i,j-1} + a_{i+1,j-1} - a_{i-1,j+1} - 2a_{i,j+1} - a_{i+1,j+1}$$

y tomando el valor del nuevo píxel como la raíz cuadrada de la suma de los cuadrados de esos dos valores:

$$b_{i,j} = \sqrt{f_1^2 + f_2^2}$$

Si el valor obtenido fuera mayor que 255 se igualaría a 255.

Para simplificar, no se tratarán los píxeles de los bordes de la imagen (ni la primera ni la última fila, ni la primera ni la última columna). Deben tratarse todos los píxeles restantes.

El programa deberá hacer lo siguiente:

- Leer la imagen del fichero en una matriz A
- Crear otra matriz B de igual tamaño que A con todos los valores a 0.
- Para cada píxel de la nueva imagen B, calcular su nuevo valor en función de los píxeles de la imagen original, aplicando el algoritmo descrito.
- Pintar en pantalla las dos imágenes A y B.