



# GRADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

## Compresión de Video

### Prácticas de Aula. Bloque 1. Tema 1

---

Nombre: \_\_\_\_\_

Fecha: \_\_\_\_\_

1. Construir un tablero de ajedrez de 8x8 cuadrados blancos y negros de 32x32 pixels cada cuadrado.
2. Se define la tasa de compresión (*compresión ratio*) de una imagen comprimida como la relación entre el tamaño de la imagen sin comprimir y la imagen comprimida. Obtener esa relación cuando la imagen 'peepers.png' se comprime en formato jpeg con un factor de calidad de 65.
3. La imagen 'electronic-circuit.tif' ha sido captada con una cámara fotográfica con una resolución de 450x450 dpi (dots per inch). Se quiere convertir y guardar esta imagen en formato tif, sin compresión y con una resolución de 200x200 dpi.

☺ Consultar el help de `imwrite` (`doc imwrite`) y buscar las opciones de compresión formato tif.

4. Después de la siguiente instrucción `I=imread('peppers.png')`, se ejecuta

```
K=im2bw(mat2gray(I),0.3);
```

Comentar las operaciones que realiza esa instrucción. Mostrar la imagen resultante.

5. Después de la siguiente instrucción `I=imread('coins.png')`, se ejecuta la orden `plot(I(125,:))`; ¿Qué representa esa gráfica?
6. Ejecutar el siguiente código:

```
I=imread('coins.png');  
mesh(double(I));
```

Seguir las siguientes instrucciones:

1. *Insert*→*Colormap*.
2. Click botón derecho del ratón sobre la barra del ColorMap. Seleccionar *Standard Colorbar*→*gray*.
3. *Tools*→*Rotate 3D*. Rotar la imagen para verla desde diferentes perspectivas.

7. La imagen 'Misc\_pollen.tif' es muy oscura. Indicar algún procedimiento para mejorar el contraste y resulte más clara.
8. Para la imagen 'lightPollen.jpg' observar las diferencias entre de mejora de contraste usando las funciones `histeq(I,256)` y `adapthisteq(I, 'Distribution', 'uniform')`. Comparar los resultados y dar una opinión personal.
9. Utilizando la función `imrotate`, girar 90°, -90° y 180° la imagen 'moon.tif'.
10. Crear la imagen negativa de 'Misc\_pollen.tif' manipulando el valor de los pixels. Repetir el ejercicio basado en la orden `imcomplement`.
11. La herramienta `angleMeasurement_tool` permite medir ángulos trazados sobre una imagen. Esta herramienta permite medir el ángulo entre dos líneas que pueden ser desplazadas y giradas por el usuario. Utilizando esta opción, medir los ángulos correspondientes a las vigas de la imagen 'gantrycrane.png'. Para ello, ejecutar el siguiente comando:


```
angleMeasurement_tool(imread('gantrycrane.png'));
```

12. Comprobar el resultado en el procesado de la imagen 'moon.tif' cuando se ejecuta este código.

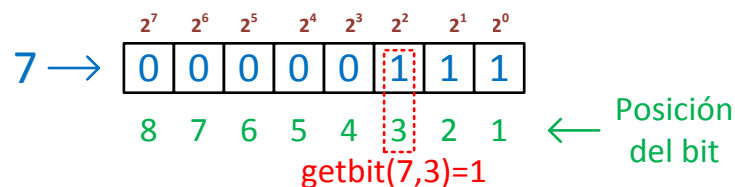
```
I=imread('moon.tif');  
I1=imadd(I,50);  
I2=immultiply(I,1.2);
```

13. Generar una imagen resultante de sumar dos imágenes: 'cameraman.tif' y 'texto.png'.
14. La aplicación `glsdemo` permite personalizar una transformación de intensidad modificando los puntos de trazada. Para ejecutar esta herramienta escribir `glsdemo` en la línea de comandos de Matlab. En la gráfica central con un click se añaden tantos puntos verdes como se desee. Estos puntos pueden ser arrastrados para dar la forma deseada. Utilizar esta herramienta para observar el efecto de cambios en el contraste.
15. El código del `identificacionTexturas.m` permite identificar cambios de texturas en una imagen. Ejecutar este código para observar el resultado.
16. Existe la función que permite visualizar dos imágenes, original y procesada, para realizar su comparación visual. Ejecutar la siguiente orden para arrancar la herramienta y visualizar las diferencias entra dos imágenes:

```
compareImagesTool('maskCircle.tif','maskCircleSmooth.tif');
```

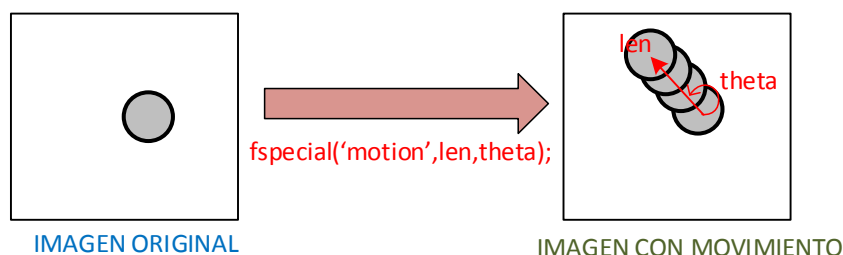
Se observa que se despliegan dos ventanas: *Overview* y *My Image Compare Tool*. En la ventana *Overview*, se puede ampliar o reducir el tamaño de las imágenes usando los botones  situados en la parte superior izquierda. Si se amplía mucho la imagen, se despliega un recuadro *pan* para seleccionar la parte visible de las imágenes.

17. Para la imagen ‘moon.tif’ se ha generado una imagen máscara ‘maskCircleSmooth.tif’. Realizar operaciones lógicas binarias entre los datos de ambas imágenes permiten generar nuevas imágenes. Para ello, aplicar las siguientes operaciones: `bitand`, `bitor`, `bitxor`, y visualizar la imagen resultante. Si es necesario, usar al help de Matlab para encontrar más información sobre estas funciones. ¿Cuál es el resultado de aplicar la función `bitcmp` a la imagen ‘moon.tif’?
18. La función `getbit` permite extraer el valor a nivel de bit de un número. Por ejemplo, `getbit(7,3)` permite extraer el valor del bit situado en la posición 3 del número 7. Gráficamente se muestra en la siguiente figura:



Si se tiene una imagen *I*, la orden `getbit(I,n)` genera otra matriz que tiene como contenido el valor del bit situado en la posición *n*. El código `planosBit.m` contiene una descripción de como desdoblar en planos de bits la imagen ‘baboon.tif’. Observar que los bits de mayor peso (más significativos) contienen mayor información de forma que los de menor peso (menos significativos) podrían ser eliminados sin alterar sustancialmente la información de la imagen. Se pide generar 8 imágenes obtenidas por combinación de suma de planos de bit: Imagen 1, planos  $2^7$ , Imagen 2, planos  $2^7+2^6$ , Imagen 3, planos  $2^7+2^6+2^5$ , etc. Observar que a partir de una imagen, la mejora de la calidad es insignificante.

19. El filtro ‘motion’ permite simular el movimiento de una cámara resultando una imagen borrosa. Usa como parámetros `len`, número de pixels desplazados, y `theta`, ángulo en grados en la dirección de desplazamiento. Gráficamente se muestra en la siguiente figura.



Escribir el siguiente código y comprobar el resultado. Probar diferentes valores enteros de **len** y **theta**.

```
% imshow(I);  
% title('Original Image (courtesy of MIT)');  
  
LEN = 17;  
THETA = 33;  
  
PSF = fspecial('motion', LEN, THETA);  
Iblurred = imfilter(I, PSF, 'conv', 'circular');  
  
subplot(1,2,1), imshow(I), title('ORIGINAL ');  
subplot(1,2,2), imshow(Iblurred), title('BLURRED');
```