

Tarea 4
IELE-4017 Análisis Inteligente de Señales y Sistemas
Profesor: Luis Felipe Giraldo Trujillo
2021-I

1. (15 puntos) Considere la imagen en escala de grises `paisaje.tiff`. Para obtener puntos en este ejercicio tiene que resolver todos los enunciados en este problema.
 - a) Del libro Digital Image Processing, de Gonzalez, lea y entienda el procedimiento para realizar la equalización de una imagen en escala de grises. Cualquier edición del libro le sirve. En este informe, explique brevemente cada paso para ecualizar el histograma de una imagen en escala de grises.
 - b) Equalice la imagen utilizando Matlab o Python. Puede utilizar la función de Matlab `histeq` o su equivalente en Python. Grafique la imagen original y la imagen resultante con sus respectivos histogramas y funciones de distribución acumulada, y analice los resultados del procedimiento de equalización. Asegúrese que la imagen leída esté en escala de grises.
NO puede utilizar las funciones de Matlab o Python que hacen estos procedimientos para calcular las funciones de distribución acumulada e histograma. Es decir, tiene que hacer la rutina que cuenta la cantidad de píxeles por cada valor de nivel de gris para el histograma y la suma para la distribución acumulada.

2. (30 puntos) El archivo `comprimida.txt` contiene los primeros 50x100 coeficientes de la transformada del coseno de una imagen en escala de grises de tamaño 630x945. Para obtener puntos en este ejercicio tiene que resolver todos los enunciados en este problema.
 - a) Utilizando alguno de los comandos `imagesc` o `bar3` visualice los coeficientes de la transformada.
 - a) Escriba una rutina en Matlab o Python para reconstruir la imagen original a partir de estos coeficientes, y muestre la imagen resultante (por ejemplo, puede utilizar la función `imshow` de Matlab). Lo más probable es que esta rutina tarde varios minutos en correr. Trate de identificar lo que contiene la imagen. NO puede utilizar funciones preestablecidas de Matlab o Python para hacer esta reconstrucción. Aquí tiene que ser una implementación clara de las ecuaciones de reconstrucción de la imagen a través de transformada del coseno.

3. (25 puntos) Considere la imagen en escala de grises `ladron.jpg`. Para obtener puntos en este ejercicio tiene que resolver todos los enunciados en este problema.
 - a) Implemente en Matlab o Python una rutina que realice la convolución en imágenes vista en clase dado un filtro o kernel de tamaño $n \times n$. NO puede utilizar funciones preestablecidas de Matlab o Python para filtrar imágenes.
 - b) Pruebe su rutina con la imagen `ladron.jpg` y dos filtros de tamaño 3×3 cada uno que realicen las siguientes operaciones: borrosado (blurring) y detección de bordes (edge detección). Muestre los filtros elegidos y las imágenes resultantes después del filtrado.

- c) Utilice las funciones `fft2`, `fftshift`, y `imagesc` de Matlab, o sus equivalentes en Python, para graficar la imagen de la magnitud de la transformada de Fourier de las tres imágenes: original, borrosa, y con bordes detectados. Asegúrese de que la frecuencia 0 de la transformada esté en el centro de la imagen. Analice los resultados teniendo en cuenta el impacto que hace cada filtro sobre la imagen desde una perspectiva frecuencial (por ejemplo, si atenúa o amplifica frecuencias altas o bajas o bandas de frecuencia).

Antes de graficar la transformada de Fourier, haga una prueba de su rutina al transformar una imagen negra con una región rectangular blanca. El resultado tiene que ser una imagen tipo sinc con magnitud máxima en el centro de la imagen.

4. (30 puntos) Considere las imágenes en escala de grises `ladron.jpg` y `ladronborroso.jpg`. Resulta que la segunda imagen es el resultado de aplicar un filtrado lineal (a través de la convolución) utilizando un kernel (es decir, un filtro) de 3×3 desconocido. Lo único que se conoce de este kernel es que los elementos de la segunda fila son cero. Para obtener puntos en este ejercicio tiene que resolver todos los enunciados en este problema.

- Intente explicar intuitivamente qué tipo de filtro se utilizó, y el efecto que está generando.
- Formule matemáticamente, implemente, y resuelva un problema de optimización que encuentre los coeficientes del kernel de tal manera que el resultado de filtrar la imagen `ladron.jpg` con ese filtro corresponde a la imagen `ladron borroso.jpg`. Vea la ayuda abajo para recomendaciones de implementación.
- Muestre el kernel que resulta del proceso de optimización y analice el porqué este kernel produce ese efecto de filtrado.
- Grafique la imagen que resulta de filtrar `ladron.jpg` con el kernel estimado, compárela con `ladron borroso.jpg`. ¿Existen diferencias?

Ayuda: Sea U una matriz asociada a la imagen `ladron.jpg`, V una matriz asociada a la imagen `ladron borroso.jpg`, y sea h un kernel de 3×3 . Denotamos la operación de filtrado entre la imagen U y el kernel h como $H\{U\}$. El problema de optimización se podría formular como encontrar los parámetros de h tal que la función $\|V - H\{U\}\|_F^2$ es minimizada, sujeto a que los elementos de la segunda fila de h son cero. La operación $\|\cdot\|_F^2$ corresponde a la norma Frobenius de una matriz. Para resolver este problema, se recomienda utilizar alguna de las funciones de Matlab `fminsearch`, `fminunc`, o `fmincon` o su equivalente en Python. Debido al tiempo de cómputo que este procedimiento de optimización puede tomar, también se recomienda utilizar la función `imfilter` en Matlab (o su equivalente en Python), la cual recibe una imagen y un kernel, y realiza el filtrado de forma eficiente. Sin embargo, usted también puede utilizar la rutina implementada en el problema 3 si así lo desea. Vean la rutina `ejemploFiltrarImagen.m` como ejemplo a seguir.