

# IEE239 - Procesamiento de Señales e Imágenes Digitales

## Laboratorio 5 - Guia Práctica

### Primer Semestre 2018

Martes, 12 de junio del 2018

#### Horario 08M1

- Duración: 2 horas, 30 minutos.
- La evaluación es **estrictamente** personal.
- Está permitido el uso de material adicional.
- **Está prohibido copiar código externo** (ejemplos de clase, material en línea, etc.).

1. (4 puntos) Se cuenta con las imágenes **crater.jpg** y **chest.png**<sup>1</sup>, en las que se requiere mejorar el contraste entre objetos a partir de técnicas de *Image Enhancement*.
  - a. Obtener la DFT 2D con origen centrado de ambas imágenes a partir de `fft2()`, `fftshift()` y `unwrap()` para  $(M, N) = (2P, 2Q)$ , donde  $P$  y  $Q$  corresponden a las dimensiones de la imagen de interés. Obtener su espectro de magnitud y fase a partir de `abs()` y `angle()` y graficarlos en una misma figura para cada caso. Qué transformación de intensidad ilustraría mejor las componentes de magnitud? Aplicarla y graficarla para ambos casos.
  - b. Realizar *Image Enhancement* sobre **crater** a partir del Laplaciano como una operación en frecuencia. Para ello, generar  $\nabla^2(u, v)$  con origen centrado para los valores  $M, N$  usados anteriormente y generar las matrices que correspondan a los pares  $(u, v)$  a partir de `meshgrid()`. Graficar su espectro de magnitud y, de acuerdo a su ganancia, indicar de qué tipo de filtro se trata. La Figura 1a muestra el espectro a obtener. Es de respuesta isotrópica? Incluir su respuesta en comentarios.

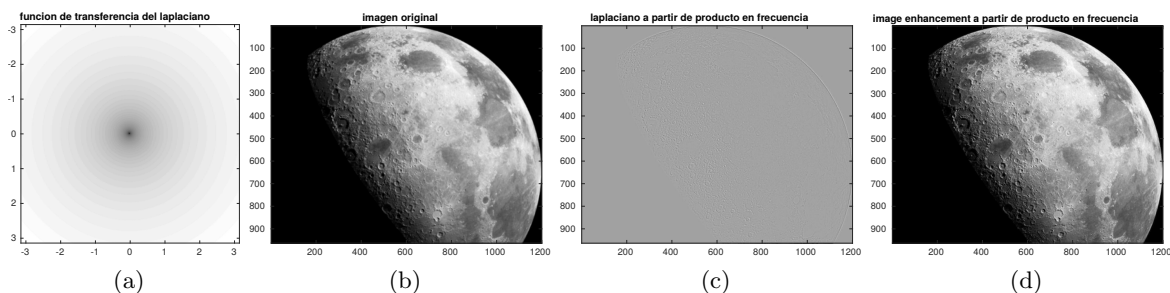


Figura 1: *Image Enhancement* a partir de producto en frecuencia. (a) Función de transferencia del Laplaciano. (b) Imagen original. (c) Laplaciano de imagen. (d) Imagen de bordes enfatizados.

- c. Realizar el producto en frecuencia entre el espectro de **crater** y  $\nabla^2(u, v)$ . Obtener la imagen resultante a partir de `ifftshift()` e `ifft2()`. Para ello, realizar lo siguiente:

---

<sup>1</sup>Las imágenes están almacenadas en la carpeta /laboratorio/lab05/08m1/guia/.

- i. Obtener  $\nabla^2(u, v) = -(u^2 + v^2)$ .
- ii. Determinar  $\nabla^2\{f(x, y)\} = \mathcal{F}^{-1}\{\nabla^2(u, v)F(u, v)\}$ .
- iii. Obtener  $g(x, y) = f(x, y) + c\nabla^2\{f(x, y)\}$  para  $c = -1$ .

Mostrar en una misma figura, con rótulos adecuados,  $f(x, y)$ ,  $\nabla^2\{f(x, y)\}$  y  $g(x, y)$ .<sup>2</sup> Las Figuras 1b a 1d muestran la imágenes a obtener. Es el efecto resultante el esperado? Justificar claramente su respuesta.

- d. Diseñar un filtro *Butterworth* pasabajos de orden  $n = 6$  y frecuencia de corte  $D_0 = \frac{2\pi}{32}$ .<sup>3</sup> Obtener  $H_{BW}(u, v)$  de origen centrado a partir de `meshgrid()` y `unwrap()`. Graficar en una misma ventana su espectro de magnitud y fase a partir de `abs()` y `angle()`. Las Figuras 2a a 2b muestra el espectro a obtener. Usar la herramienta `data cursor` para analizar la ganancia del filtro en puntos  $(u, v)$  con distancia  $D_0$  respecto al origen e incluir dicho valor en comentarios.

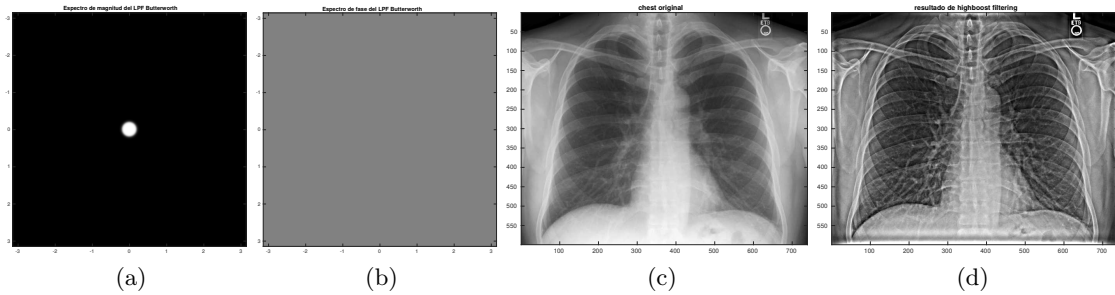


Figura 2: *Highboost Filtering* a partir de producto en frecuencia. (a) Espectro de magnitud del filtro Butterworth. (b) Espectro de fase del filtro Butterworth. (c) Imagen original. (d) Imagen de bordes enfatizados.

- e. Considerando un factor de ganancia para bajas frecuencias  $k_1 = 1$  y uno para altas frecuencias  $k_2 = 5$ , aplicar el método de *Highboost Filtering* a **chest** de acuerdo a la siguiente relación y las rutinas `ifftshift()`, `ifft2()`:

$$g(x, y) = \mathcal{F}^{-1}\{k_1 + k_2[1 - H_{BW}(u, v)]F(u, v)\}.$$

Graficar en una misma ventana, con rótulos adecuados,  $f(x, y)$  y  $g(x, y)$ . Verificar que el resultado enfatiza detalles y preserva las regiones planas en la imagen original. Las figuras 2c a 2d muestran las imagenes resulantes.

2. (3 puntos) Se cuenta con las imagenes **build.jpg** y **build\_ang.jpg**<sup>4</sup>, las cuales corresponden a una misma escena pero con un cambio de orientación. Adicionalmente, se sabe que **build\_ang** sufre distorsiones de intensidad de acuerdo al siguiente modelo:

$$b(x, y) = u(x, y) + \eta(x, y),$$

donde  $u(x, y)$  es la imagen con cambio de orientación de factor desconocido, y  $\eta(x, y) = \cos(\frac{\pi}{16}x + \frac{\pi}{32}y)$  es una señal superpuesta.

<sup>2</sup>Para evitar cambios de escala respecto a la imagen de original de 8 bits, realizar a un cast a `uint8` para realizar la gráfica de  $g(x, y)$ .

<sup>3</sup>Ya que el número de muestras en frecuencia de la imagen a tratar es de  $(M, N)$ , el espacio de frecuencia del filtro a diseñar también debe cubrir el mismo dominio para poder ser multiplicado punto a punto.

<sup>4</sup>Las imágenes están almacenadas en la carpeta `/laboratorio/lab05/08m1/guia/`.

---

**Algoritmo 1** Estimar el cambio de orientación entre imágenes.

---

**Entrada:**  $f(x, y)$ : imagen a transformar,  $r(x, y)$ : imagen de referencia,  $\Theta$ : vector de dimensiones  $P \times 1$  con los ángulos a analizar (en radianes).

**Salida:**  $\theta_{\text{opt}}$ : ángulo que obtiene la mayor métrica de similitud entre imágenes.

```

for  $i = 0$  to  $P - 1$  do
     $\theta \leftarrow \Theta_i$ 
     $\hat{f}(x, y) \leftarrow \text{imrotate}\{f(x, y), \theta\}$                                  $\triangleright$  Aplicar rotación. Revisar imrotate().
     $\varepsilon(i) \leftarrow d_2\{\text{vec}(\hat{f}(x, y)), \text{vec}(r(x, y))\}$                  $\triangleright$  Evaluar similitud. Revisar norm().
end for

 $i_{\text{true}} \leftarrow \underset{i}{\text{argmin}} \quad \varepsilon(i)$                                  $\triangleright$  Índice del ángulo óptimo. Revisar min().
 $\theta_{\text{opt}} \leftarrow \Theta_{i_{\text{true}}}$ 

```

---

- a. Para estimar el cambio de orientación entre imágenes, realizar una búsqueda exhaustiva de acuerdo al Algoritmo 1.<sup>5</sup> Para ello, considerar a **build** como la imagen a transformar, a **build\_ang** como la imagen de referencia y a  $\Theta = \{\frac{\pi}{18}, \frac{\pi}{9}, \dots, \frac{\pi}{2}\}$  como los ángulos a evaluar. Con lo obtenido, graficar en una misma ventana la imagen **build\_ang** y su versión con cambio de orientación según el ángulo estimado con el algoritmo. Las Figuras 3a y 3b muestran los resultados esperados.

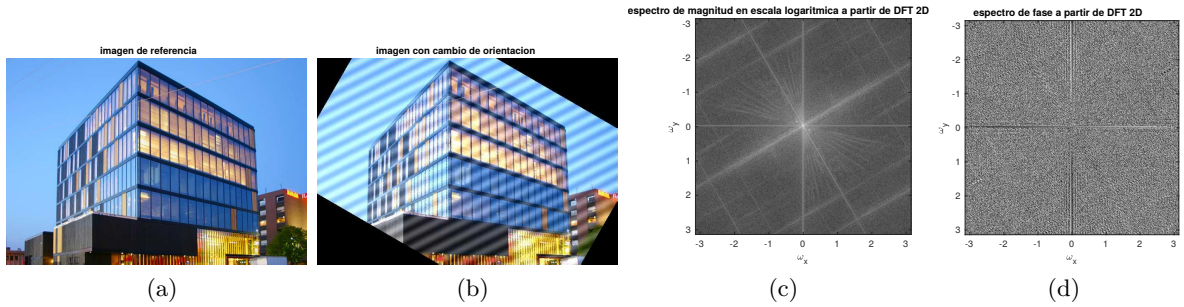


Figura 3: Transformaciones de espacio e intensidad. (a) Imagen original. (b) Imagen distorsionada con cambio de orientación corregido. (c) Espectro de magnitud de imagen distorsionada. (d) Espectro de fase de imagen distorsionada.

- b. Considerando la capa azul, obtener la DFT 2D de **build\_ang** aprovechando la propiedad de separabilidad. Con los valores anteriores de  $M$  y  $N$ , calcular la DFT 2D como dos DFT 1D a partir de `dftmtx()`, `fftshift()` y `unwrap()`. Obtener y graficar su espectro de magnitud y fase a partir de `abs()` y `angle()`<sup>6</sup>. Considerando la frecuencia de  $\eta$ , es posible estimar el cambio de orientación a partir de su DFT 2D? En caso sea cierto, indicar el procedimiento e incluir su valor estimado en comentarios. Las Figuras 3c y 3d muestran el espectro de magnitud y espectro de fase de la imagen, respectivamente.

- c. A partir de lo obtenido en el inciso 2b, calcular la DFT 2D inversa aprovechando la

---

<sup>5</sup>Para generar cambios de orientación de un factor arbitrario que preserve el tamaño original de la imagen, considerar la rutina `imrotate()` y la bandera `crop`.

<sup>6</sup>Aplicar una transformación logarítmica para aumentar su contraste.

propiedad de *dualidad* de la DFT:

$$\mathcal{F}^{-1}\{G(u, v)\} = MN\mathcal{F}^*\{G^*(u, v)\},$$

donde  $*$  denota la operación complejo conjugado. Verificar lo obtenido a partir de `real()`, `ifftshift()` e `ifft2()`. Graficar ambas imagenes resultantes y verificar que son iguales.

- d. Aplicar el método de *Marr-Hildreth* para obtener el mapa de bordes de **build**. Obtener la máscara de LoG empleando `fspecial()` y considerar la rutina `edge()` con bandera `'zerocross'` para los umbrales  $T \in \{0.35g_{\max}, 0.05g_{\max}\}$ , donde  $g_{\max}$  denota el valor máximo de la respuesta ante el filtro Laplaciano de Gaussiano. Mostrar el espectro de magnitud de  $\nabla^2 G_{x,y}$  y determinar de qué tipo de filtro se trata. Luego, mostrar los resultados para ambos umbrales adecuadamente rotuladas y en una misma ventana. La Figura 5 describe la función de transferencia del operador LoG, así como los mapas de bordes a obtener. Cuál es el efecto del valor umbral en la robustez ante distorsiones en la imagen?

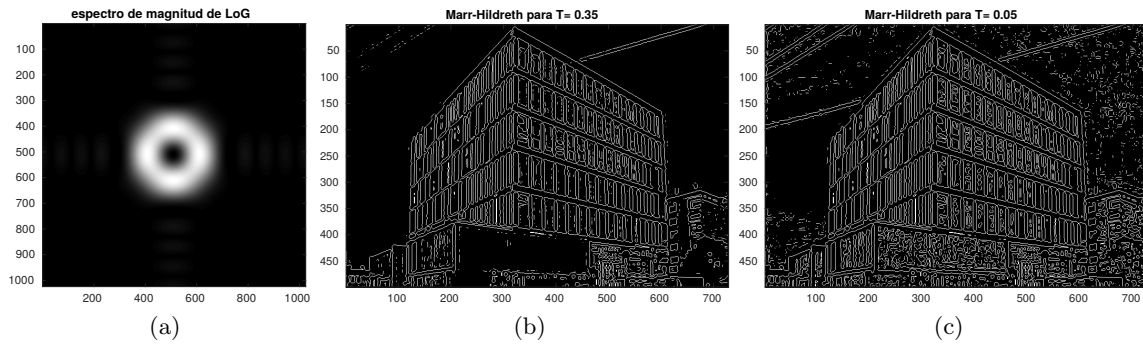


Figura 4: Método de Marr-Hildreth. (a) Espectro de frecuencia del Laplaciano de Gaussiano. (b) Mapa de bordes para  $T = 0.35g_{\max}$ . (c) Mapa de bordes para  $T = 0.05g_{\max}$ .

3. (3 puntos) Se cuenta con la imagen **candy.jpg**<sup>7</sup>, en el espacio de colores **RGB**. Se requiere realizar una segmentación a color de los objetos en ella.
  - a. Cargar la imagen a partir de `imread()` y verificar que se trata de una variable sin signo de 8 bits de resolución de intensidad empleando `whos()`. La Figura 5a muestra la imagen de interés. Luego, obtener su versión en escala de grises a partir de `rgb2gray()` y obtener su histograma empleando `hist()`. Graficar el vector resultante y rotularlo adecuadamente a partir de `bar()`. Se puede clasificar su distribución como unimodal, bimodal o multimodal?
  - b. A partir del inciso 3a, determinar el valor umbral  $T$  como la semisuma de las modas del histograma. Luego, determinar la imagen binaria  $m(x, y)$ :

$$m(x, y) = \begin{cases} 1, & f_{\text{grises}}(x, y) > T \\ 0, & \text{otros casos} \end{cases},$$

donde  $f_{\text{grises}}(x, y)$  corresponde a **candy** en escala de grises. Luego, usar  $m(x, y)$  para enmascarar cada capa de la imagen original. Graficar la imagen inicial y resultante en una misma ventana y rotular adecuadamente. La Figura 5b muestra la imagen a colores umbralizada. Es el umbral obtenido un parametro adecuado para separar objetos de diferentes colores? Justificar claramente su respuesta en comentarios.

<sup>7</sup>Las imágenes están almacenadas en la carpeta /laboratorio/lab05/08m1/guia/.

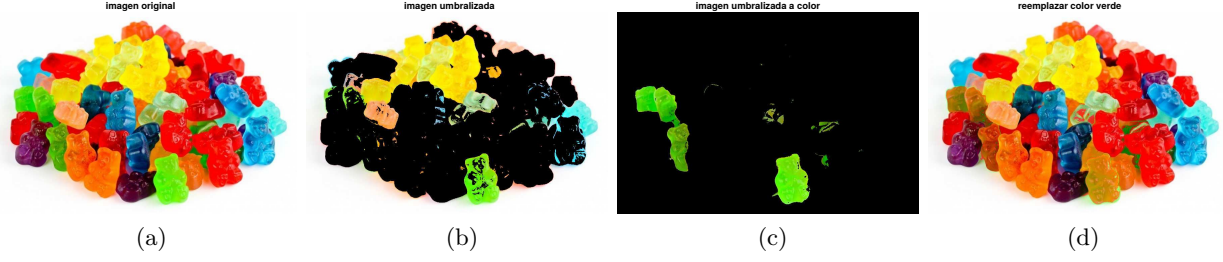


Figura 5: Segmentación de múltiples capas. (a) Imagen original. (b) Segmentación basada en escala de grises. (c) Segmentación basada en intensidad RGB de interés. (d) Cambio de color en imágenes RGB.

- c. Considerar una segmentación realizada directamente en el espacio de colores RGB. Para ello, definir un color de interés  $\mathbf{c} = \{90, 220, 0\}$  y una distancia  $D_{\min} = 105$ . Realizar un cast a `double` sobre la imagen y obtener la imagen segmentada  $z(x, y)$  a partir de la siguiente expresión, donde  $D_2$  corresponde a la distancia Euclidiana:

$$z(x, y) = \begin{cases} f(x, y), & D_2\{f(x, y), \mathbf{c}\} < D_{\min}, \\ 0, & \text{otros casos} \end{cases},$$

$$D_2\{f(x, y), \mathbf{c}\} \triangleq \left( \sum_{i=0}^2 (f_i(x, y) - c_i)^2 \right)^{\frac{1}{2}}.$$

Finalmente, mostrar las imágenes original y resultante en una misma figura, adecuadamente rotuladas, empleando `imshow()`. La Figura 5c muestra la imagen a colores segmentada según valores cercanos a  $\mathbf{c}$ . A comparación del inciso anterior, qué método permite seleccionar de manera más precisa el color a preservar en la imagen original? Justificar claramente su respuesta en comentarios.

- d. A partir de la máscara  $m(x, y)$ , modificar la tonalidad de los elementos de color verde. Para ello, aumentar en 100 la capa roja y reducir en 100 la capa verde de los píxeles que cumplen con la condición de acuerdo a  $m(x, y)$ . Finalmente, graficar la imagen resultante a partir de `imshow()`. La Figura 5d muestra la imagen a colores modificada según  $m(x, y)$ . Cuál es el efecto del brillo en el método de segmentación? Incluir su respuesta en comentarios.