

IEE239 - Procesamiento de Señales e Imágenes Digitales

Laboratorio 5 - Aplicación

Primer Semestre 2018

Martes, 12 de junio del 2018

Horario 08M1

- Duración: 1 hora.
- La evaluación es **estrictamente** personal.
- Está permitido el uso de material adicional.
- **Está prohibido copiar código externo (ejemplos de clase, material en línea, etc.).**

(5 puntos) Se desea segmentar imágenes RGB integrando la información de sus capas. Para ello, se propone el método de *k-medias* (*k-means clustering*), el cual asocia las intensidades de todos los píxeles a un conjunto limitado de intensidades. Con el propósito analizar la imagen original y aplicar la segmentación descrita, considerar el siguiente procedimiento:

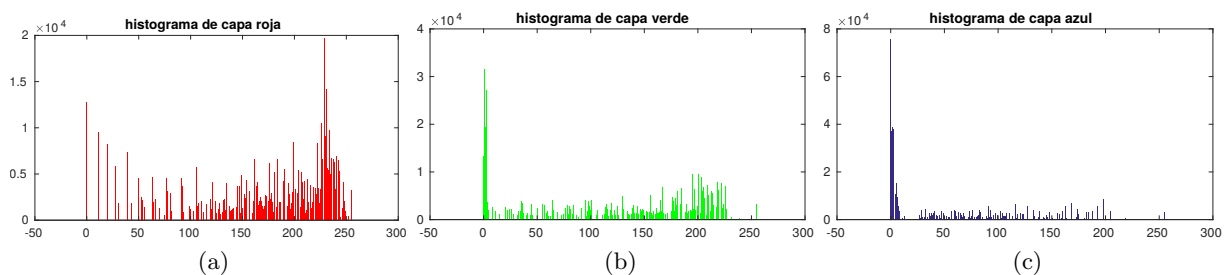


Figura 1: histogramas de la imagen de interés en RGB. (a) Histograma de la capa roja. (b) Histograma de la capa verde. (c) Histograma de la capa azul.

- (0.5 puntos) Se cuenta con la imagen **fruits.png**¹ en el espacio de color RGB. Cargar la imagen a partir de `imread` y graficarla empleando `imshow()`. Usar el comando `whos` para obtener el espacio en memoria que ocupa e incluir dicho valor en comentarios.
- (1 punto) Determinar el histograma de cada una de las capas de la imagen empleando `hist()` y graficarlas en una misma ventana usando `bar()` y rotulando adecuadamente. Es posible atribuirle alguna distribución a algún histograma? Incluir su respuesta en comentarios. La Figura 1 muestra los vectores a obtener.
- (1 punto) Obtener la versión en escala de grises de **fruits** a partir de `rgb2gray()`. Luego, a partir del concepto de derivadas de primer orden por *forward difference*, obtener la magnitud y ángulo de $\nabla f(x, y)$, así como la siguiente máscara de bordes:

$$M(x, y) = \begin{cases} 1, & |\nabla f(x, y)| > \alpha \cdot \max\{|\nabla f(x, y)|\}, \\ 0, & \text{otros casos} \end{cases},$$

¹La imagen está almacenada en la carpeta `/lab05/08m1/aplicacion/`.

Algoritmo 1 Método de *k-medias*

Entrada: $D \in \mathbb{R}^{PQ \times 3}$: elementos de la imagen RGB ordenados como vectores 3d, **iter**: número de iteraciones, K : número de clases a emplear.

Salida: $S \in \mathbb{R}^{PQ \times 1}$: índices que asocian cada elemento de D a una de las clases, m : conjunto final de medias obtenidas.

```
 $m_i \leftarrow \mathcal{U}(0, 1), i \in [0, K - 1]$   $\triangleright$  Iniciar medias con distribucion uniforme entre 0 y 1.  
 $N \leftarrow PQ$   
for  $i = 0$  to iter-1 do  
  for  $j = 0$  to  $N - 1$  do  
     $S_j \leftarrow \underset{k}{\operatorname{argmin}} \{d_2\{D_j, m_k\}\}$   $\triangleright$  Hallar media mas cercana a cada punto a partir de  
    distancia euclidiana ( $d_2$ ).  
  end for  
  for  $j = 0$  to  $K - 1$  do  
     $G_j \leftarrow$  Indices correspondientes a la clase  $j$   
     $m_j \leftarrow \frac{1}{\# \text{elementos en } G_j} \sum_{x_l \in G_j} x_l$   $\triangleright$  Hallar nuevas medias para cada clase.  
  end for  
end for
```

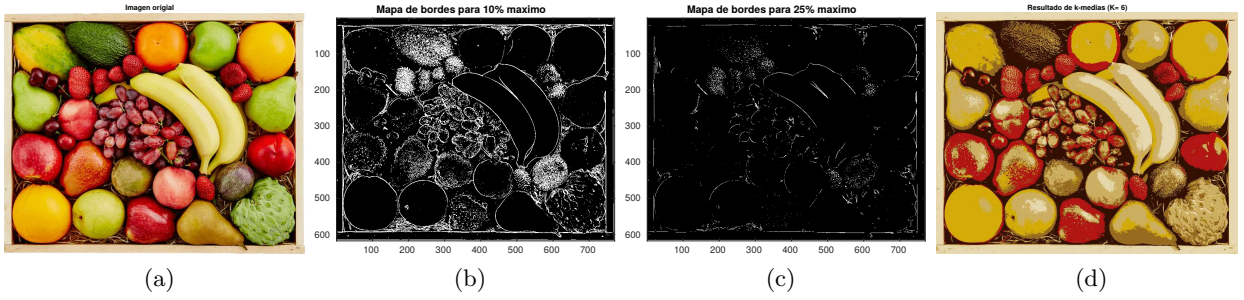


Figura 2: segmentación de imágenes en RGB. (a) Imagen original. (b) Mapa de bordes para un umbral del 10 % máximo. (c) Mapa de bordes para un umbral del 25 % máximo. (d) Imagen segmentada según *k-medias*.

considerando los valores $\alpha \in \{0.1, 0.25\}$. En una misma ventana, graficar ambos mapas. Es alguno de ellos lo suficientemente preciso para segmentar los objetos presentes en la imagen? Cuál es el efecto de la textura propia de la imagen sobre el mapa de bordes? Incluir su respuesta en comentarios. Las Figuras 2b y 2c muestran los mapas de bordes a obtener.

- d. (2.5 puntos) Crear la subrutina `func_kmeans()` con el método de *k-medias*, descrito en el Algoritmo 1, y aplicarlo sobre la imagen a colores **fruits**. Para ello, considerar $K = 6$ clases, 25 iteraciones e inicializar de forma aleatoria sus medias en el espacio de características. Luego, generar una imagen a colores en las cuales cada pixel RGB tome el valor de la media más cercana a su intensidad RGB. Graficar la imagen original y la resultante en una misma ventana. Las Figuras 2a y 2d muestran las imagenes a obtener.