

Tratamiento Discreto de Señales 1

Práctica 1-Entorno Matlab para procesado de imagen

28 de febrero de 2005

1 Objetivos

El objetivo de la práctica consiste en la familiarización con el entorno de trabajo que se empleará durante el desarrollo de la asignatura. Para ello se propondrá la programación de tres breves rutinas que, tratando con los conceptos vistos en la parte teórica de la misma, permitan al alumno un primer acercamiento a las principales características y funcionalidades del toolbox de imagen del entorno Matlab (el manual del toolbox de imagen se encuentra en el fichero `images_tb.pdf`)

2 Bandas de Mach

- Contruya un script de Matlab que cree la imagen del experimento de las Bandas de Mach. Para ello, determine un número de filas y columnas razonable (típicamente 256×256), un número de bandas a observar, y elija los niveles de gris de forma que el rango de intensidades comúnmente empleado (de 0 a 255, correspondiendo el 0 al negro y el 255 al blanco) quede barrido de forma más o menos homogénea. Dote al programa de flexibilidad para poder cambiar el número de bandas a voluntad.
- Observe el resultado mediante el uso de la orden `imshow`.
- Considere los dos formatos de almacenamiento de imágenes empleado por Matlab, a saber, el modo *double* y el modo *uint8*. Si A es una imagen en modo *double*, y B lo es en modo *uint8*, la relación entre ellas puede ser

```
A = double(B)/255;  
B = uint8(round(A*255));
```

Considere, en relación con la imagen anteriormente creada, ambos formatos de imagen, observe las diferencias y compare los tamaños en memoria de la imagen en ambos casos.

- Asocie ahora a cada banda (por tanto, a cada nivel de gris) un color. Para ello, cree un mapa de colores, esto es, una matriz de tantas filas como bandas haya creado, y con tres columnas; cada fila contendrá las tres componentes de color en el espacio RGB empleado por Matlab. A partir de este momento, y para que las cosas tengan sentido, los valores de intensidad de las bandas de Mach deben interpretarse como índices al mapa de colores, es decir, el valor de cada pixel de la imagen indicará el índice de la fila de la cual extraer el color del pixel. Haga una nueva llamada a la orden `imshow` haciendo los cambios oportunos para poder observar la imagen en color deseada¹.

¹También podríamos haber creado el mismo efecto mediante una matriz de tres dimensiones, las dos primeras

3 Creación de una señal sinusoidal

- Se pretende ahora que cree una imagen consistente en una señal sinusoidal acorde con la expresión

$$I(x, y) = A(y)\cos(\omega(x)x)$$

con $A(y)$ y $\omega(x)$ funciones monótonas crecientes.

- Para mostrar la imagen creada por pantalla puede olvidar cuestiones de normalización de rangos de intensidad empleando un segundo argumento en la orden `imshow` de la forma siguiente; `imshow(A, [])`, donde se ha considerado que la imagen se encuentra almacenada en la matriz `A`.
- Compruebe el cumplimiento de los fenómenos visuales asociados a este experimento.

4 Creación de una animación

- Se pretende ahora que escriba un script que permita crear una animación a partir de una imagen original.
- Para ello, haciendo uso de la orden `imread`, lea la imagen `cameraman.tif`.
- La animación consistirá en la creación de una imagen *multiframe*, esto es, una matriz de cuatro dimensiones, las dos primeras coincidentes con el tamaño de la imagen original, la tercera igual a uno (en nuestro caso, dado que es una imagen en tonos de gris. Si la imagen original hubiese sido una imagen RGB, esta dimensión haría sido igual a 3), y en la cuarta se almacenarán las diferentes instantáneas de la imagen en movimiento.
- Cree una imagen en movimiento, por ejemplo, haciendo que las columnas de la matriz de desplacen hacia la derecha, en modo circulante (esto es, que cuando una columna salga por la derecha de los límites de la imagen, vuelva a entrar por la izquierda)².
- Una vez creada la imagen multiframe, debe crear, a partir de la anterior, una imagen de índices y su mapa correspondiente. Para ello, haga uso de la instrucción `gray2ind`; cuando ambos hayan sido generados, puede crear la película mediante la orden `immovie`;
- Experimente con los parámetros de su código hasta conseguir una sensación de movimiento razonablemente continua.

dimensiones coincidentes con el tamaño de la imagen, y la tercera dimensión de magnitud igual a 3. A través de esta tercera dimensión almacenaríamos, para cada pixel, su tripleta de valores del sistema RGB.

²Recuerde que a la matriz

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

se le denomina *matriz circulante*.