



GRADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

Compresión de Video

Ejercicios Prácticos Bloque 1. Tema 2

Nombre: _____

Fecha: _____

➤ Lectura de parámetros de un video.

```
info = mmfileinfo('testVideo.mpg') % Lectura básica de información de video
info.Audio
info.Video

vidObj = VideoReader('testVideo.mpg') % Lectura de un video

imshow('testVideo.mpg'); % Reproductor de video de Matlab
%% Tools-> Maintain Fit to Window
```

Ejemplo 1: Lectura de parámetros de un video

```
vidObj = VideoReader('testVideo.mpg');
fprintf('\n BitsPerPixel=%d', vidObj.BitsPerPixel);
fprintf('\n Duration=%f', vidObj.Duration);
fprintf('\n FrameRate=%f', vidObj.FrameRate);
fprintf('\n Height=%d', vidObj.Height);
fprintf('\n Width=%d', vidObj.Width);
fprintf('\n NumberOfFrames=%d', vidObj.NumberOfFrames);
fprintf('\n VideoFormat=%s\n', vidObj.VideoFormat);
```

Ejemplo 2: Lectura y reproducción de un video en formato mpeg

```
% Lectura del video
vidObj = VideoReader('testVideo.mpg');

% Precarga la estructura de datos.
mov(1:vidObj.NumberOfFrames) = ...
    struct('cdata', zeros(vidObj.Height, vidObj.Width, 3, 'uint8'),...
        'colormap', []);

% Lectura frame a frame.
for frame = 1 : vidObj.NumberOfFrames-1
    mov(frame).cdata= read(vidObj, frame);
end

% Visualización del video
movie(mov, 1, vidObj.FrameRate);
```

Ejercicio prácticos 1.

1. Utilizando el ejemplo anterior, visualizar en una figura la frame número 18.
2. Comprobar lo que sucede si en la reproducción del video se utiliza esta orden
`movie(mov, vidObj.NumberOfFrames:-1:1, vidObj.FrameRate)`
3. Reemplazar la orden de lectura frame a frame por la siguiente orden:
`mov(frame).cdata= imrotate(read(vidObj, frame), 90);`

➤ Conversión de espacios de colores: RGB a YCbCr.

```
ycbcrmap = rgb2ycbcr(Y); % Conversión RGB a YCbCr
YCBCR = rgb2ycbcr(RGB); % Conversión YCbCr a RGB
```

Ejemplo 3: Conversión RGB a YCrCb 4:4:4

```
% Lectura de la imagen
RGB = imread('peppers.png');
[fil col dim]=size(RGB);
whos RGB

% Se convierte RGB a YCbCr.
YCbCr = rgb2ycbcr(RGB);
whos YCbCr

% Observar que la imagen RGB y YCrCb ocupan el mismo espacio

% Visualización de las componentes RGB.
close all;
hFig = figure;
set(hFig, 'Position', [20 200 600 600]);
subplot(2,2,1), imshow(RGB), title('ORIGINAL')
subplot(2,2,2),
imshow(cat(3,RGB(:,:,1),zeros(fil,col,'uint8'),zeros(fil,col,'uint8'))),
title('RED');
subplot(2,2,3),
imshow(cat(3,zeros(fil,col,'uint8'),RGB(:,:,2),zeros(fil,col,'uint8'))),
title('GREEN');
subplot(2,2,4),
imshow(cat(3,zeros(fil,col,'uint8'),zeros(fil,col,'uint8'),RGB(:,:,3))),
title('BLUE');

% Se muestran las componentes Y, Cr y Cb.
hFig = figure;
set(hFig, 'Position', [650 200 600 600]);
subplot(2,2,1), imshow(RGB), title('ORIGINAL')
subplot(2,2,2), imshow(YCbCr(:,:,1)), title('Y Component');
subplot(2,2,3),
imshow(cat(3,zeros(fil,col,'uint8'),zeros(fil,col,'uint8'),YCbCr(:,:,2))),
title('Cb Component');
subplot(2,2,4),
imshow(cat(3,YCbCr(:,:,3),zeros(fil,col,'uint8'),zeros(fil,col,'uint8'))), title('Cr Component');
```

➤ Trabajando con video YUV

stefan@90@352x288.yuv es un video en formato YUV 4:2:0 de 90 frames de tamaño CIF(352x288). Para visualizar este video usar la aplicación *yuvplayer.exe*. Al abrir la aplicación seleccionar *File* → *Open Video 1* el ese video con *Video Size* CIF y el *Video format* en 4:2:0.

La visualización de ese video también se puede realizar desde Matlab. Para ello, se ha desarrollado la función `readYUV.m` de lectura de videos en formato YUV es:

```
[yuvMovie, YUV] = readYUV(filename, numOffFrames, fileFormat);
```

El parámetro `fileFormat` tiene que ser uno de estos valores:

NTSC:	Y: 720 x 480 ; U,V: 360 x 240
PAL:	Y: 720 x 576 ; U,V: 320 x 288

```

MPEGYUV      Y: 704 x 480 ; U,V: 352 x 240
CIF_NTSC:    Y: 352 x 240 ; U,V: 176 x 120
CIF_PAL:     Y: 352 x 288 ; U,V: 176 x 144
QCIF_NTSC:   Y: 176 x 120 ; U,V: 88 x 60
QCIF_PAL:    Y: 176 x 144 ; U,V: 88 x 72

```

Ejemplo 4: Visualización de un video en formato YUV

```

% Lectura del video YUCV 4:2:0
[yuvMovie, YUV] = readYUV('stefan@90@352x288.yuv', 90, 'CIF_PAL');

% Visualización del video
imshow(yuvMovie);

```

➤ Video entrelazado.

Ejemplo 5: Generación de frames entrelazadas.

```

% Lectura del video
[yuvMovie, YUV] = readYUV('stefan@90@352x288.yuv', 3, 'CIF_PAL'); % Se leen 3 frames

% size(YUV) % Se comprueba las dimensiones de YUV

% Se lee la luminancia de la frame 2
Ycurr=YUV(:,:,1,2);
[fil, col]=size(Ycurr);

% Se crean imagenes de ceros
Yodd=zeros(fil,col);
Yeven=zeros(fil,col);

% Se seleccionan las filas pares e impares por separado
Yodd(1:2:fil,:)=Ycurr(1:2:fil,:);
Yeven(2:2:fil,:)=Ycurr(2:2:fil,:);

% Visualización de las imágenes
subplot(1,3,1), imshow(uint8(Ycurr)), title('ORIGINAL Y');
subplot(1,3,2), imshow(uint8(Yodd)), title('Imagen Líneas impares');
subplot(1,3,3), imshow(uint8(Yeven)), title('Imagen Líneas pares');

```

Ejemplo 6: Efecto de interlaced frames.

```

% Lectura de un video YUV 4:2:0
[yuvMovie, YUV] = readYUV('stefan@90@352x288.yuv', 3, 'CIF_PAL'); % Se leen 3 frames

% size(YUV) % Se comprueba las dimensiones de YUV

% Se lee la luminancia de la frame 2
Ycurr=YUV(:,:,1,2);
[fil, col]=size(Ycurr);
% Se lee la luminancia de la frame 3
Ynext=YUV(:,:,1,3);

% Se crean imagenes de ceros
Yodd=zeros(fil,col);

```

```

Yeven=zeros(fil,col);

% Se seleccionan las filas pares e impares por separado
Yodd(1:2:fil,:)=Ycurr(1:2:fil,:);
Yeven(2:2:fil,:)=Ynext(2:2:fil,:);

% Imagen reconstruida
Yrec(1:2:fil,:)=Ycurr(1:2:fil,:);
Yrec(2:2:fil,:)=Ynext(2:2:fil,:);

% Visualización de las imágenes
subplot(3,2,1), imshow(uint8(Ycurr)), title('Frame 2: Original');
subplot(3,2,2), imshow(uint8(Ynext)), title('Frame 3: Original');
subplot(3,2,3), imshow(uint8(Yodd)), title('Frame 2: Líneas impares');
subplot(3,2,4), imshow(uint8(Yeven)), title('Frame 3: Líneas pares');
subplot(3,2,5), imshow(uint8(Yrec)), title('Interlaced: Frame2 + Frame3');

```

➤ Programación de la webcam.

Ejemplo 7: Programación de la webcam.

```

imagreset % Reset de la Imagen Aquisition Toolbox
% imaqttool

% Configuración de la webcam
vid = videoinput('winvideo', 2, 'YUY2_176x144');
src = getselectedsource(vid);
vid.ReturnedColorspace = 'grayscale'; % rgb 'YCbCr'

vid.FramesPerTrigger = 1;

% LAzo infinito
while(1)

start(vid);
frame = getdata(vid); % getsnapshot(vid); % Captura una imagen
stop(vid);

% Visualización de la frame capturada
subplot(1,2,1), imshow(frame), title('ORIGINAL');
subplot(1,2,2), imshow(imcomplement(frame)), title('NEGATIVA');
%subplot(1,2,2), imshow(edge(frame)), title('DETECCIÓN DE CONTORNO');
end

```