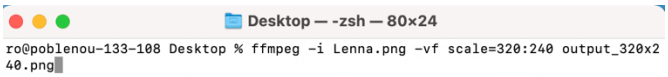
Roger Viñeta I Mañé
216694
16/11/2021

AUDIO AND VIDEO ENCODING SYSTEMS – P1

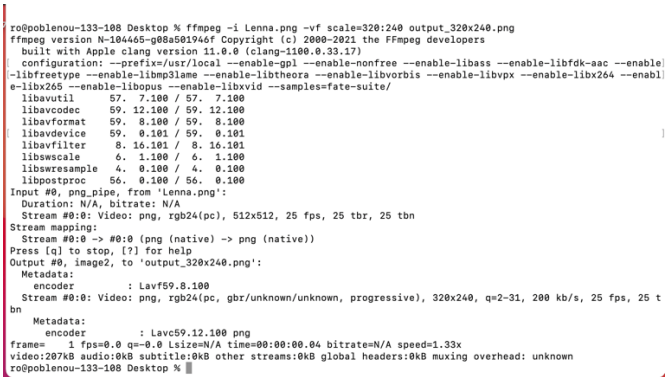
EX1 Start a script called rgb_yuv.py and create a translator from 3 values in RGB into the 3 YUV values, plus the opposite operation.
We have created a script where you introduce either some RGB or YUV values and it converts them to the other format. You can introduce them via keyboard in your computer. We put the rgb_yuv.py file on the zip.

EX2 Use ffmpeg to resize images into lower quality.
In this exercise, we scaled an image through ffmpeg. We have taken a photogram from Lenna that was 512x512 and rescaled to 320x240.

The command line used for resize transformation is:
ffmpeg -i Lenna.png -vf scale=320:240 output_320x240.png



OUTPUT



EX3 Use FFMPEG to transform the Lenna image into b/w. Do the hardest compression you can and comment the results
In this exercise, we took a lenna color photo and used the command below in order to convert the image to black and white. We have obtained the results that we show below.

The command line used for black and white transformation is:
ffmpeg -i *Lenna.png* -vf format=gray bw_lenna.png

The command line used compression is:
ffmpeg -compression_level 50 -i lenna.png lenna_comp.png

```
ro@poblenou-133-108 P1 % ffmpeg -i Lenna.png -vf format=gray bw_lenna.png
ffmpeg version N-104465-g08a501946f Copyright (c) 2000-2021 the FFmpeg developers
  built with Apple clang version 11.0.0 (clang-1100.0.33.17)
  configuration: --prefix=/usr/local --enable-gpl --enable-nonfree --enable-libass --enable-libfdk-aac --enable
-libfreetype --enable-libmp3lame --enable-libtheora --enable-libvorbis --enable-libvpx --enable-libx264 --enabl
e-libx265 --enable-libopus --enable-libxvid --samples=fate-suite/
  libavutil      57.  7.100 / 57.  7.100
  libavcodec     59. 12.100 / 59. 12.100
  libavformat    59.  8.100 / 59.  8.100
  libavdevice    59.  0.101 / 59.  0.101
  libavfilter     8. 16.101 /  8. 16.101
  libswscale      6.  1.100 /  6.  1.100
  libswresample   4.  0.100 /  4.  0.100
  libpostproc    56.  0.100 / 56.  0.100
Input #0, png_pipe, from 'Lenna.png':
  Duration: N/A, bitrate: N/A
  Stream #0:0: Video: png, rgb24(pc), 512x512, 25 fps, 25 tbr, 25 tbn
Stream mapping:
  Stream #0:0 -> #0:0 (png (native) -> png (native))
Press [q] to stop, [?] for help
Output #0, image2, to 'bw_lenna.png':
  Metadata:
    encoder         : Lavf59.8.100
  Stream #0:0: Video: png, gray(pc, progressive), 512x512, q=2-31, 200 kb/s, 25 fps, 25 tbn
    Metadata:
      encoder         : Lavc59.12.100 png
frame=    1 fps=0.0 q=-0.0 Lsize=N/A time=00:00:00.04 bitrate=N/A speed=1.32x
video:218kB audio:0kB subtitle:0kB other streams:0kB global headers:0kB muxing overhead: unknown
ro@poblenou-133-108 P1 %
```

We can see that if we try to compress the image with the following command the results we get are the following: Codec AVOption compression_level () specified for input file #0 (Lenna.png) is not a decoding option.

We have been investigating and realized that it is not possible to compress more a .png file.

```
ro@Rogers-MacBook-Pro P1 % ffmpeg -compression_level 50 -i Lenna.png lenna_comp.
png
ffmpeg version N-104465-g08a501946f Copyright (c) 2000-2021 the FFmpeg developer
s
  built with Apple clang version 11.0.0 (clang-1100.0.33.17)
  configuration: --prefix=/usr/local --enable-gpl --enable-nonfree --enable-liba
ss --enable-libfdk-aac --enable-libfreetype --enable-libmp3lame --enable-libtheo
ra --enable-libvorbis --enable-libvpx --enable-libx264 --enable-libx265 --enable
-libopus --enable-libxvid --samples=fate-suite/
  libavutil      57.  7.100 / 57.  7.100
  libavcodec     59. 12.100 / 59. 12.100
  libavformat    59.  8.100 / 59.  8.100
  libavdevice    59.  0.101 / 59.  0.101
  libavfilter     8. 16.101 /  8. 16.101
  libswscale      6.  1.100 /  6.  1.100
  libswresample   4.  0.100 /  4.  0.100
  libpostproc    56.  0.100 / 56.  0.100
Input #0, png_pipe, from 'Lenna.png':
  Duration: N/A, bitrate: N/A
  Stream #0:0: Video: png, rgb24(pc), 512x512, 25 fps, 25 tbr, 25 tbn
Codec AVOption compression_level () specified for input file #0 (Lenna.png) is n
ot a decoding option.
ro@Rogers-MacBook-Pro P1 %
```

EX4 Create a script which contains a function which applies a run-lenght encoding from a series of bytes given.

We have created a script that runs the run-length encoding from a series of bytes given. We can check that on the script in the zip file.

EX5 Create a script which can convert, can decode (or both) an input using the DCT. Not necessary a JPG encoder or decoder. A script only about DCT is OK too.

We have attached two scripts, one that only computes the DCT and the IDCT and other that makes all the process. It's kind of low the one that takes all the process and we have just let it run for 6/7 minutes.