UNIVERSIDADE DE AVEIRO
DEPARTAMENTO DE ELECTRÓNICA, TELECOMUNICAÇÕES E INFORMÁTICA

Information and Coding (2023/24)

Lab work nº 3 — Due: 7 Jan 2024

Intro

Using the Golomb coding algorithm, you have to implement a video codec for grayscale video sequences. The final codec should rely on block based motion compensation and spatial predictive coding. You should use uncompressed videos, such as those available in `https://media.xiph.org/video/derf/`, represented in the uncompressed YUV4MPEG format (see, for example, `https://wiki.multimedia.cx/index.php/YUV4MPEG2` and `https://linux.die.net/man/5/yuv4mpeg` for a specification of this format). Because the codec will operate only on grayscale, first you will have to appropriately extract the luminance component (Y component) from the original videos.

Part I

1. Implement a lossless image codec for grayscale images, based on Golomb coding of the prediction residuals. This codec should provide good compression, using appropriate predictors and values of m for the Golomb codes. 
2. Adapt the image codec in order to encode videos as simple sequences of images (i.e., intra-frame coding only). Remember that the file with the encoded video should contain all the parameters needed for decoding.

Part II

3. Expand your codec to support inter-frame (temporal) prediction (also known as motion compensation). The codec will have to comply with the following requisites:
   (a) The periodicity of the intra-frames (also known as I or key frames) should be an input parameter of the encoder (the inter-frames are also called P frames).
   (b) The block size and the search area for inter-frame coding should be an input parameter of the encoder.
   (c) For P frames, it will have to estimate if the current block should be encoded in intra or inter mode (based on the bitrate produced).

Part III

4. Implement a lossy version of the video codec, still based on Golomb coding of the prediction residuals, i.e., by simple quantization of those residuals. To help in assessing the quality of the lossy compressed video sequences, develop a `video_cmp` program, that compares two video sequences in terms of the peak signal to noise ratio (PSNR), given by

$$\text{PSNR} = 10 \log_{10} \frac{A^2}{e^2},$$

where A is the maximum value of the signal (typically 255), e^2 is the mean squared error between the reconstructed frame, $\tilde{f}$, and the original frame, f,

$$e^2 = \frac{1}{NM} \sum_{r=1}^{N} \sum_{c=1}^{M} [f(r,c) - \tilde{f}(r,c)]^2,$$

and where N and M denote, respectively, the number of rows and columns of the video frames.

Part IV

5. Elaborate a report, where you describe all the relevant steps and decisions taken in all the items of the work. When appropriate, include also measures of processing time, compression ratios and corresponding errors introduced by the compression/decompression process. For this, use several video examples. Also, it is important to compare your results with those obtained with some existing video codecs.

6. Create a video presentation, of at most 5 minutes, to "sell" what you have done in this lab work.