CSE 408 Multimedia Information Systems

Phase #2

(Due November 1 2015, midnight)

Description: In this project, you will

- experiment with video data
- experiment with lossy and lossless encoding techniques

Tasks:

The input to the project will be a video file and a 10 pixel by 10 pixel region. You will operate only on the Y channel.

Task I (Temporal Predictive coding): Implement a program which treats the video region as 100 separate signals (one for each pixel) and includes the following predictive coding (PC) schemes (that will be applied separately on each signal):

- *PC Option 1:* No PC (use original values).
- *PC Option 2:* Predictive encoding with the predictor $s_i[t] = s_i[t-1]$.
- *PC Option 3:* Predictive encoding with the predictor $s_i[t] = \frac{s_i[t-1]+s_i[t-2]}{2}$.
- *PC Option 4:* Predictive encoding with the predictor $s_i[t] = \alpha_1 \times s_i[t-1] + \alpha_2 \times s_i[t-2]$, where α_1 and α_2 are two non-negative values such that $\alpha_1 + \alpha_2 = 1.0$ and

$$\alpha_1 \times s_i[t-2] + \alpha_2 \times s_i[t-3] = s_i[t-1]$$

$$\alpha_1 \times s_i[t-3] + \alpha_2 \times s_i[t-4] = s_i[t-2]$$

  In case there are not sufficient observations to compute α_1 and α_2, use $\alpha_1 = \alpha_2 = 0.5$ as default.

The program should output its result into a text file named as

$X\_Y.tpc$

where X is the name of the video file and y is the option number. The program should output the total (absolute) prediction error.

Task II (Spatial Predictive coding): Implement a program which treats the video as $numframes$ separate 2D signals (one for each frame) and includes the following predictive coding (PC) schemes (that will be applied separately on each 2D signal):

- *PC Option 1:* No PC (use original values).
- *PC Option 2:* Predictive encoding with the predictor A.

- *PC Option 3:* Predictive encoding with the predictor B.
- *PC Option 4:* Predictive encoding with the predictor C.
- *PC Option 5:* Predictive encoding with the predictor $\alpha_a \times A + \alpha_b \times B + \alpha_c \times C$, where α_1, α_2, and α_3 are three non-negative values such that $\alpha_1 + \alpha_2 + \alpha_3 = 1.0$ and would correspond to zero-error predictions for the most recent three past predictions.

  In case there are not sufficient observations to compute α_1, α_2, and α_3 use $\alpha_1 = \alpha_2 = \alpha_3 = \frac{1}{3}$ as default.

The program should output its result into a text file named as

$$X\_Y.spc$$

where X is the name of the video file and y is the option number. The program should also output the total (absolute) prediction error.

Task III: You will implement the following error quantization schemes (that will be applied after temporal or spatial predictive coding):

- *Error Quantization Option 1:* No quantization.
- *Error Quantization Option 2:* Quantization of the error into 2^m uniform bins for a given $m \geq 1$.

The program should output its result into a text file labeled as

$$X\_Y\_Z.tpq$$

or

$$X\_Y\_Z.spq$$

where for option 1, $Z = 0$ and for option 2, $Z = m$.

Task IV: You will implement the following encoding schemes that generates a bit stream given the output of Task III:

- *Encoding Option 1:* No compression (non-integer entries should be stored as double),
- *Encoding Option 2:* Variable-length encoding with Shannon-Fano coding
- *Encoding Option 3:* Dictionary encoding with LZW coding (for a given dictionary bit length)
- *Encoding Option 4:* Arithmetic coding

The result should be written into a binary output file

$$X\_Y\_Z\_C.tpv$$

or

$$X\_Y\_Z\_C.spv$$

where C is the compression option. The total amount of distortion (signal-to-noise ratio) between the original video and the encoded video and the size of the encoded video should also be printed.

Task V: Implement also a viewer that reads the given binary file and displays the decoded video.

Deliverables:

- Your code (properly commented) and a README file.
- A report describing your work and the results on a sample image.

Please place your code in a directory titled "Code", the outputs to a directory called "Outputs", and your report in a directory called "Report"; zip or tar all off them together and submit it through the Blackboard.