EE565: Video Communications

Prof. Kondi

Spring 2007

Homework #2

Due: Thursday, March 8, 2007

In this assignment, you are asked to implement a simplified Motion-Compensated DCT based video encoder. You are asked to do the following:

1. Load into Matlab frames 0 and 1 of the "Foreman" sequence (luminance only) to be found in the course web page. The files `frame0.raw` and `frame1.raw` are in "raw" format as explained in class. To load, for example, file `frame0.raw` into Matlab, you can issue the commands: `fin=fopen('frame0.raw','r');`, `frame0=fread(fin,[176,144])';`, `fclose(fin);`.

2. Frame 0 is to be encoded in INTRA mode while frame 1 is to be encoded in INTER mode. Partition frame 1 into 16×16 macroblocks and perform motion estimation between frame 1 and frame 0 to find one motion vector per macroblock. Use the Sum of Absolute Differences (SAD) criterion to select the motion vectors. You are allowed to select the search window to use.

3. Partition frame 0 into 8×8 blocks, take the DCT of each block using Matlab function `dct2` and quantize according to the H.263 standard. Calculate the entropy of the output of the quantizer (quantization levels, ac coefficients only). You do not have to perform a zigzag scan or calculate the zero runs as in H.263. Then reconstruct frame 0 by performing "inverse quantization" followed by inverse DCT using function `idct2`. Calculate the PSNR of reconstructed frame 0.

4. Perform motion compensation and construct a prediction for frame 1 using the *reconstructed* frame 0 and the motion vectors. Calculate the prediction error (displaced frame difference) and encode it using DCT and quantization. Calculate the entropy of the output of the quantizer. Reconstruct frame 1 by performing "inverse quantization" and inverse DCT on the prediction error and adding it to the motion-compensated reconstructed frame 0. Calculate the PSNR of reconstructed frame 1.

Steps 3 and 4 need to be repeated three times for three different Quantization Parameters (QP). The QP to be used are: 10, 20, 31. Submit a printout of the Matlab code, printouts of the reconstructed frames 0 and 1, the entropies of the quantization levels of frames 0 and 1 and the PSNR of frames 0 and 1.

Regarding the quantization and "inverse quantization" use the following information:

QP may take integer values from 1 to 31. The quantization step size is then $2 \times QP$.

The following definitions of divisions are made:

- /: Integer division with truncation towards zero.
- //: Integer division with rounding to the nearest integer. (examples: 3//2=2, -3//2=-2).

Thus, if COF is a transform coefficient to be quantized and $LEVEL$ is the absolute value of the quantized version of the transform coefficient, the quantization is done as follows:

For intra blocks (except for the dc coefficient):

$$LEVEL = |COF|/(2 \times QP). \tag{1}$$

For inter blocks (all coefficients, including the dc):

$$LEVEL = (|COF| - QP/2)/(2 \times QP). \tag{2}$$

For the dc coefficient of an intra block:

$$LEVEL = COF//8. \tag{3}$$

For the inverse quantization, if COF' is the reconstructed transform coefficient and $LEVEL$ is the absolute value of the quantized version of the transform coefficient, the inverse quantization is done as follows:

For inter blocks and intra blocks, except for the dc coefficient:

$$|COF'| = \begin{cases} 0 & \text{if } LEVEL = 0 \\ 2 \times QP \times LEVEL + QP & \text{if } LEVEL \neq 0,\ QP \text{ is odd} \\ 2 \times QP \times LEVEL + QP - 1 & \text{if } LEVEL \neq 0,\ QP \text{ is even} \end{cases} \tag{4}$$

The sign of COF is then added to obtain COF':

$$COF' = \text{Sign}(COF) \times |COF'| \tag{5}$$

For the dc coefficient of an intra block:

$$COF' = LEVEL \times 8 \tag{6}$$