CS3570 Introduction to Multimedia Technology

Homework #1

Due: 11:59pm, March 24, 2012

1. **Color quantization**

   Color quantization is a process that reduces the total number of distinct colors to represent an image. The given mat file 'colors.mat' contains eight quantized colors in 8x3 RGB matrix format from 'clash1.png'. Please apply color quantization to the following color image by using these eight index colors.

   (a) Show the eight color patches (8 patches)
   (b) Implement the color quantization and show the quantized image (1 image)
   (c) Calculate the frequencies of these eight colors and use Shannon's entropy equation to calculate the optimum number of bits to encode each color, and then calculate the average bits for a pixel. You have to show these values in report.
   (d) Implement Shannon-Fano algorithm to apply entropy coding for coding the quantized colors. Show the entropy code for each color and the average bits for representing a pixel in this case. You have to implement your own code, describe your implementation and show the results in the report.
   (e) Compare the coding efficiency in (c) & (d).

   Hint: popularity algorithm in unit2 and entropy coding in unit1.
   Matlab functions you can use in this problem: reshape, repmat, min, numel.
   Matlab functions you can NOT use in this problem: kmeans.

2. **DCT image compression**

   Transform the image ('clash2.png') from spatial domain to frequency domain by DCT, and then reconstruct the image with inverse DCT for three different cases with reduced numbers of DCT coefficients.

   The first step is to divide image into blocks with 8x8 pixels for each block, followed by applying 2D DCT for each block. Then, for each block, only keep the lower-frequency, i.e. upper-left n-by-n, coefficients in the 2D DCT domain by setting the remaining coefficients to zero. The final step is to reconstruct the image by taking inverse 2D DCT with the modified DCT coefficients for each block.

   (a) Implement the above simplified DCT compression process for n = 2, 4, and 8 and apply it to the following grayscale image. Show the reconstructed images for these three different cases. (3

images)

(b) Compute the PSNR values of the three reconstructed images in (a). Discuss what the PSNR value means here.

Hint: $\text{PSNR} = 10\ log_{10}(\frac{MAX^2}{MSE})$,

$\text{MAX} = \text{maximum pixel value},\ \text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2$

Hint: when you calculate PSNR, remember to normalize images to 0~255 or 0~1.

Matlab functions you can NOT use in this problem: psnr, dctmtx, dct, idct2.

3. Image interpolation

Implement the following two image interpolation methods (in (a) and (b)) to upsample the following image ('clash3.png') to 4 times of its height and width sizes, respectively.

(a) Nearest-neighbor (NN) interpolation (1 image)

(b) Bilinear interpolation (1 image)

(c) Discuss the results in (a) & (b)

Matlab functions you can NOT use: imresize

You should submit {student_ID}_report.pdf, the output result images, and codes in {student_ID}.zip

Your report should contain at least how you implement the methods and how to implement your codes.