**CS3570 Introduction to Multimedia**

Homework #1

Due: 11:59pm, 2017/03/27

## 1. DCT image compression (30%)

Transform the image, *lalaland.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.

- Divide image into blocks with $8 \times 8$ pixels for each block, followed by applying 2D DCT for each block.
- 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.
- 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 attached image. Show the reconstructed images for these three different cases. [3 images] Compute the PSNR values of the three reconstructed images and discuss what the PSNR value means here.

(b) Use the same process in (a) with image transformed to YIQ color model and show the reconstructed image in RGB space. [3 images] Compute the PSNR values of the three reconstructed images and discuss what the PSNR value means here.

(c) Compare the differences between the results in two color spaces in (a) and (b).

**Note:**

- You should not use the Matlab built-in functions, such as dctmtx, dct, dct2, idct, idct2, rgb2ntsc, ntsc2rgb, psnr, etc.
- *Definition of PSNR:*

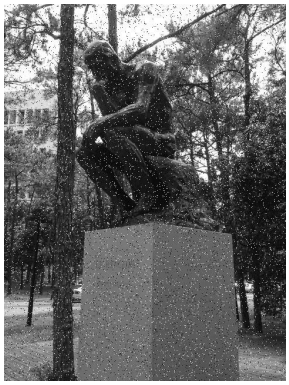
$$PSNR = 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$
$$= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right)$$
$$= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE)$$

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

$I$: original image, $K$: compressed image

## 2. Image filtering (30%)

Create convolution masks with different sizes and perform convolution with these masks on the gray scale image with noises, *thinker_gray_noised.jpg*, and show the results after applying the following image filters. Discuss what you observe in each questions.



(a) Apply Gaussian blur filter on the provided image with two mask sizes $3 \times 3$ with $\text{sigma} = 0.3$ and $9 \times 9$ with $\text{sigma} = 1.0$. Compare these two results. [2 images]

(b) Apply median filter on the provided image with filter sizes $3 \times 3$ and $9 \times 9$. Compare the outputs, describe what you observe from the results. [2 images] Also discuss on the differences between the results with median filtering and the ones with Gaussian blurring in (a).

**Note:** Implement 2D convolution and filtering on your own. You can use fspecial function in Matlab to create Gaussian masks, but you should not use the Matlab built-in functions, such as conv2, imfilter, filter2, imgaussfilt, medfilt2, etc.

## 3. Interpolation (40%)

Implement the image interpolation function to upsample an image to four times the original width and height. Implement the following two different interpolation methods and show the 4× upsampled images.

(a) Apply nearest-neighbor interpolation on the low resolution image, *img_LR.png*, and compute the PSNR with the original high resolution image, *img_HR.png*. [1 image]

(b) Apply bilinear interpolation on the low resolution image and also compute the PSNR with the high resolution image. [1 image]

(c) Compare and discuss the results from the above two methods and give the meaning of PSNR values to these results.

**Note:**

- You should not use the Matlab built-in functions, such as imresize, psnr, etc.

## Reminder:

- Matlab built-in functions listed in problem description are prohibited.
- Your code should work correctly and generated results (display or output files) must be consistent to your results in report.
- Report format can be in Word, PowerPoint or others that can describe your work and result clearly.
- In report, should contain at least how you implement the methods and discussion to the output results.
- **Pack {student_ID}_report.pdf, the output result images, and codes in {student_ID}.zip. Your package should also contain a README file about how to execute your program.**