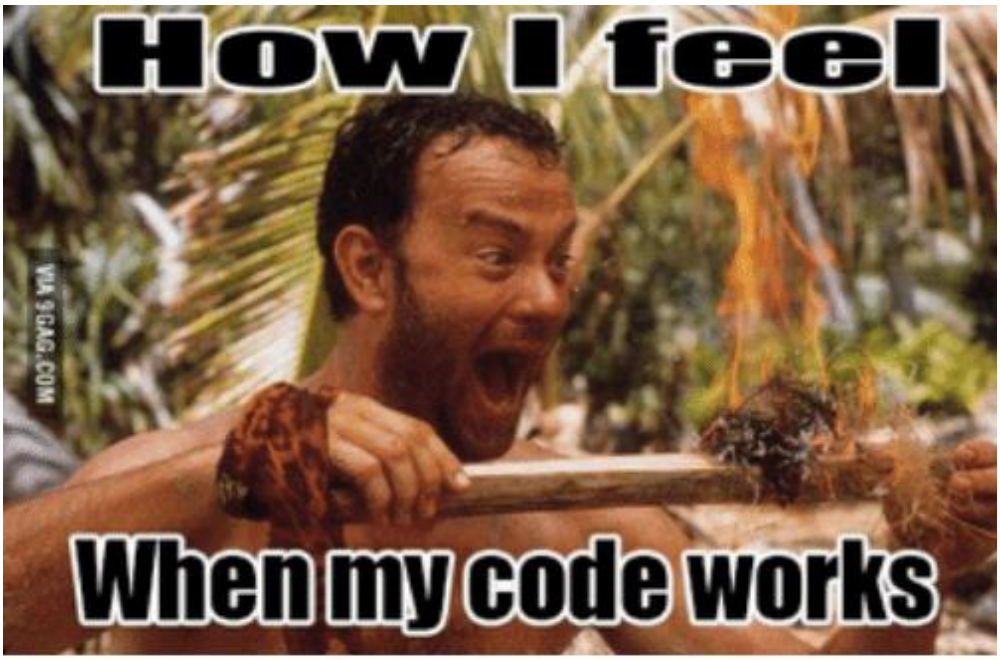
PROJECT II

EE409 DIGITAL IMAGE PROCESSING

DUE DATE: (DEC 03, 2017)



INSTRUCTIONS

- **Please read these instructions in full before proceeding to problems**

- **This project contains 3 problems. Rules of submission are same as for Project 1.**

- **For each problem, the relevant theory and instructions are placed in a separate document. (CCSeg.pdf, Gamma.pdf and PCA.pdf in Handouts folder). The required data is all contained in Data folder.**

- **Please note that you only have to implement selected sections from each of these documents. These sections and tasks are described in the instructions below.**

- **For some tasks, however, you might need to read the entire document. These documents have really good descriptions of connected components, gamma and PCA so it is advisable to read them in entirety.**

These exercises were originally developed by Charlie and Jordan at Purdue. All credit goes to them.

PROBLEM 1

This problem deals with connected components and image segmentation. You are supposed to write C code for thresholding and region growing algorithm, along with image segmentation. For this problem, refer to the accompanying document '**CCSeg.pdf**' for detailed instructions.

1. Implement the `ConnectedNeighbors` and `ConnectedSet` functions described in the CCSeg.pdf.

2. Apply the `ConnectedSet` function to extract pixels connected to (67,45) and T = 2, as explained on page 4 of CCSeg.pdf. Please note that you only have to show results for T = 2.

3. Generate a segmentation of the image as explained in Section 2 of CCSeg.pdf. You have to show the segmentation generated by T = 2, along with number of regions.

Report must include:

1. **Image showing connected set for T = 2**

2. **Randomly colored segmentation for T = 2**

3. **Number of regions in this segmentation and your C code**

PLEASE NOTE THAT YOU HAVE TO FOLLOW INSTRUCTIONS GIVEN IN CCSEG.PDF EXCEPT YOU ONLY HAVE TO SHOW RESULTS FOR T=2 FOR BOTH CONNECTED COMPONENTS AND SEGMENTATION.

PROBLEM 2

This problem is about gamma correction. We will calculate gamma for your display device so it is important that each student does his/her own exercise. For this problem, refer to the **Section 4 of** '**Gamma.pdf**' for detailed instructions.

1. Create an image of matching gray level as explained in Gamma.pdf

2. Derive an expression relating the matching gray level to the gamma of your monitor. Report the measured gray level and the measured gamma.

3. Gamma correct the image linear.tif. Display the uncorrected and corrected image along with how you did the correction.

Report must include:

1. **The matching gray level image**

2. **Derivation relating the matching gray level to the gamma of your monitor and the measured gray level and the measured gamma.**

3. **Uncorrected linear.tif and corrected linear.tif.**

PLEASE NOTE THAT YOU DO NOT HAVE TO IMPLEMENT TO IMPLEMENT THE HISTOGRAM PART OF THE GAMMA.PDF. YOU ONLY NEED TO IMPLEMENT ABOVE MENTIONED PARTS AND IGNORE THE REST.

PROBLEM 3

This problem is about eigenimages and Principal Component Analysis. The accompanying document PCA.pdf provides a nice, thorough explanation of the brief summary of PCA and eigenvectors we discussed in class. Please read through the whole document to solidify your understanding of eigenimages and PCA.

As for execution, you only have to implement Section 4 of the PCA.pdf (I mean if you insist, I could add more…)

1. Vectorize the training data as explained in Section 4.1

2. Compute the eigenvalues and eigenvectors of the image covariance

3. Reconstruct the first four letters using only *m* eigenvectors, as explained in section 4.

Report must include:

1. **First 12 eigenimages**

2. **Projection coefficients vs. eigenvector numbers**

3. **Resynthesized versions of the original image with different *m***

PLEASE NOTE THAT YOU ONLY NEED TO IMPLEMENT ABOVE MENTIONED PARTS AND IGNORE THE REST.