Yale CS 479/579 Problem Set 4 2014

Due Thursday, April 3, 2014, 1:00 pm
Submit by uploading to classesv2.

Below you will find the problems for Problem Set 4 Please note the following:

1. All problems should be uploaded to ClassesV2 together in a single zip folder.
2. Each problem should be in its own folder of the format "problem*n*". For example the solution to the first problem should be placed in a folder called "problem1a"
3. All assignments will be graded in the Zoo, which is a Linux environment. This means that all file and folder names are case-sensitive! It also means that you should test your solution on the Zoo computers.
4. You can start Matlab on the Zoo by opening a terminal and typing in *matlab*.
5. An automatic grader may be used, which may not be forgiving of formatting errors.
6. Check ClassesV2 from time to time, as we will post updates and correct any errors in the assignment there.
7. All Matlab functions should have its own m file with the same name as the function (as per Matlab's convention)

# **Problem 1** (C/C++, 479 and 579)

a.) Write C/C++ code to pull mattes from images. Use the method that uses images of the object in front of two different backgrounds. Assume that RGB images for two different backgrounds with and without the object are given i. Produce a matte – that is an image with the value of alpha between 0 (full background) and 255 (full foreground object) for each object.

You should create a program that uses your code to do matting on input files. The program `matt` should take 5 parameters: the 2 background images, the 2 composite images, and the output file in that order. The program should create an output tiff that includes both the foreground color and alpha map. You may assume that all input files are the same size.

You should include a make file for your program which includes any compiler options for external libraries you may use along with all files that we need to use the libraries without any additional downloads or configuration. However, you need not include the image files that we provided you. The following must work in the zoo after we unzip your code, change to the proper directory, and copy the image files into the directory (and without doing anything else):

```
make
./matt bach_mat/red_background.tif bach_mat/yellow_background.tif bach_mat/red_composite.tif bach_mat/yellow_composite.tif out.tif
```

You can display the alpha channel of an image on the zoo with the command:

```
display -alpha Extract filename.tiff
```

b.) Use your code to produce mattes for three objects – a glass, a teddy bear and a bust of Bach – from the tiff images in the directories for Problem 1. Submit the mattes, and produce examples of blending each object onto a new background. Discuss how well the method worked for each case. What, if anything, about each case made it particularly difficult?
You should place your images and answers in a **readme.pdf** file as well as **including all mattes and**

**blending, each in their own file, in an image format**.

**EXTRA CREDIT –** Take a set of your own pictures of an object with two backgrounds and produce a matte for the object.

# **Problem 2** (Matlab)

a.) (479 and 579) Write Matlab code to take images captured with full illumination and two different lighting patterns (the patterns are inverses of one another—i.e. pattern 1 is black where pattern 2 is white and vice versa) and produces two images – one with just direct illumination and one with just scattered light.

You should place your Matlab code in a function

```
[direct,scatterd] = illumination(full, pattern1, pattern2)
```

The inputs and the outputs should be uint8 arrays with values between 0-255. The function should work with both 2d and 3d arrays. You may assume that all inputs are the same size.

b)(479 and 579) Use your code to extract direct and scattered light from the same images given for a dish, a class of cloudy liquid and a sculpture given in the directory for Problem 2. Note that your results will have artifacts due to imperfections in the lighting patterns. Find the areas in each image that have the most scattered light, and explain why the area has more scattering than other areas.

c) (579 only) Write Matlab code to get rid of the artifacts that result from the imperfect patterns used.

Place your cod in a function:

```
[direct,scatterd] = illumination_fixed(full, pattern1, pattern2)
```

The inputs and the outputs should be uint8 arrays with values between 0-255. The function should, at the very least, work with 2d arrays. You may assume that all inputs are the same size.

**EXTRA CREDIT –** Take a set of your own pictures of an object with patterns to study the scattering on the object. You can (for example) project patterns by connecting your computer to the overhead projector in rooms 000, 200, 306, or 400.

> **NOTE:** These problems are based on methods by Smith And Blinn (1996) and Nayar et al. 2006. The papers are also available with the assignment if you would like to refer to them.