

For information about the CS department's computer labs and/or to setup a new UNIX account, go to <http://www.cs.utexas.edu/facilities/computing/>.

I. Using Matlab [60 points]

The goal of this problem set is to become familiar with basic Matlab commands, practice manipulating vectors and matrices, and try out basic image display and plotting functions. If you are unsure what a Matlab function does, check the reference manual (at the command line, type 'help' and then the command name).

1. Read over the provided Matlab introduction code and its comments:

<http://www.cs.utexas.edu/~grauman/courses/fall2009/psets/pset0/matlab.pdf>

Open an interactive session in Matlab and test the commands by typing them at the prompt. (Skip this step if you are already familiar with Matlab.)

2. Describe (in words where appropriate) the result of each of the following Matlab commands. Use the `help` command as needed, but try to determine the output without entering the commands into Matlab. Do **not** submit a screenshot of the result of typing these commands.

a. `>> x = randperm(5);`

b. `>> a = [1:10];`
`>> b = a([1:3:end]);`

c. `>> f = [1501:2000];`
`>> g = find(f > 1850);`
`>> h = f(g);`

d. `>> x = 22.*ones(1,10);`
`>> y = sum(x);`

e. `>> a = [1:1000];`
`>> b = a([end:-1:1]);`

3. Given a 100 x 100 uint8 matrix `A` representing a grayscale image, write a few lines of code to do each of the following. Try to avoid using loops.
 - a. Sort all the intensities in `A`, put the result in a single 10,000-dimensional vector `x`, and plot the values in `x`.
 - b. Display a figure showing a histogram of `A`'s intensities with 32 bins.

- c. Create and display a new binary image the same size as A , which is white wherever the intensity in A is greater than a threshold τ , and black everywhere else.
- d. Display the bottom right quadrant of A .
- e. Generate a new image (matrix), which is the same as A , but with A 's mean intensity value subtracted from each pixel. Set any negative values to 0.
- f. Use `rand` to write a function that returns the roll of a six-sided die.
- g. Let y be the vector: $y = [1:6]$. Use the `reshape` command to form a new matrix z whose first column is $[1, 2, 3]'$, and whose second column is $[4, 5, 6]'$.
- h. Use the `min` and `find` functions to set x to the single minimum value that occurs in A , and set r to the row it occurs in and c to the column it occurs in.
- i. Let v be the vector: $v = [1\ 8\ 8\ 2\ 1\ 3\ 9\ 8]$. Use the `unique` function to compute the total number of unique values that occur in v .

II. Programming problem: averaging images [40 points]

Write a program that will average a collection of images, compute the standard deviations at each pixel, and display the results.

The images below give some examples that were generated by averaging "100 unique commemorative photographs culled from the internet" by Jason Salavon. Your program will do something similar.



Download the images provided on the course website for this assignment. There are two sets, set1 and set2. Notice that they are all the same size within a single set. Then write code to do these things per set of images:

- Compute the average image in grayscale.
- Compute the average image in color, by averaging per RGB channel.
- Compute a matrix holding the grayscale images' standard deviation at each pixel (i.e., $X(i,j)$ holds the standard deviation across all the images' gray pixel intensities at row i , column j).
- Display each of the above.

Apply these steps to the two sets, separately. In your write-up, briefly explain the results – why do they look the way they do?

Matlab tips: Be sure to do the necessary typecasting (uint8 and double) when working with or displaying the images. Some useful functions: `title`, `subplot`, `imagesc`, `imshow`, `std`, `mean`, `imread`, `rgb2gray`, `dir`.

This code fragment can be used to loop through all the image files in one directory (here assuming your images are in a sub-directory named `images`, with `.jpg` extension):

```
filelist = dir('images/*.jpg');
for i=1:length(filelist)
    imname = ['images/' filelist(i).name];
    nextim = imread(imname);
    . . .
end
```

Submission instructions: what to hand in

Prepare these files:

- Single PDF file with your name at the top of it:
 - Responses and code snippets for Part I questions
 - Explanation and screenshot outputs for the image results in Part II
- Code files: Documented code for Part II, as `.m` file(s).

Submit the files electronically via the `turnin` program:

```
>> turnin --submit jaechul pset0 pset0.pdf part2code.m ...etc
```

Bring a hardcopy of the PDF file to class on Tuesday, Sept. 8. (The hardcopy must be identical to what you submit electronically by the deadline Monday night.)