Assignment #1

How to Deal with Images in MATLAB!

How TA evaluates your assignments:

Report: half of your score will be graded proportional to the quality of your report. You should provide a distinct section for each problem, include the desired outputs and explain what you've done. Don't forget to discuss your results as well. Note that in your reports, all your figures must have captions. It is not necessary to accommodate your source codes in your reports unless you want to refer to them. Compactness, expressiveness and neatness are of high importance.

Source Code: create an m-file for any problem and write all your codes there. If a problem consists of several sub-problems, separate them by comments in your code. Finally, name your m-files according to the number of the problems.

For evaluating your codes, TA creates two empty folders just beside your m-files, named as "input" and "output" (in the same directory). Then, he copies the input materials into the "input" folder and executes your m-files. Next, the output files will be checked in the "output" folder. Therefore, write your codes so as to load input files from an "input" folder and save the output files in an "output" folder. The exact name of the input and output files will be provided in the problem descriptions.

What to hand in:

You must submit your report (.pdf), source codes (m-files) and output files for each assignment. Zip all your files into an archive file and use the following template to name it:

HW1_XXXXX.zip

where XXXXX must be replaced with your student ID. Your file size must not be bigger than 20MB. If there is any question, don't hesitate to contact us through nasiri.hamid@gmail.com, s.izadi@live.com

The Due Date for This Assignment is: March 8Th

---

1. In this problem you are supposed to get familiar with image formation using basic commands in Matlab.

a. Create an array of zeroes 512 points on a side, and display it on axes that are evenly and equally spaced using the axis command. Include the results in your report.

b. Add circles of radius 16 and amplitude 1 to the array with centers at locations (200, 200), (232,200), (200,300), (248, 300), (200,400), and (264,400), and display. You will have to do some matlab programming to do this. Include the resultant figures in your report.

2. There are many useful built-in functions in MATLAB that should be used to increase both readability and efficiency of implementations. In this problem, you will get familiar with some of these functions. ***For this problem, include all resultant images in your report.***

Load the images 'rice.png' and 'cameraman.tif':

a. Use MATLAB command 'imresize' to resize one of the images to be the same size as other. What arguments the aforesaid command receives? You may find the command 'size' useful in finding the size of images. Provide a summary description on both commands.

b. Add images together using MATLAB addition operator '+' and 'imadd' function. What are the differences? Explain your observations.

d. Do the same for operator '-' and 'imsubtract' function.

c. Explain different datatypes in MATLAB, particularly the ones which are used for working with images. What are built-in MATLAB function for converting images into a particular datatype? Provide a summary description for each function.

Load image 'pepper.png':

d. Apply MATLAB function 'imcomplement' on the image. What effect do you see and to what aspect of traditional photography does the resulting image coloring relate?

e. Image inversion is also of particular use in medical imaging for highlighting different aspects of a given image. Apply the operator to the example images 'mir.tif', 'spine.tif' and cell image 'AT3 1m4 09.tif'. What resulting effects do you see in the transformed images which may be beneficial to a human viewer?

Load the images AT3_1m4_01.tif', 'AT3_1m4_02.tif' , ' AT3_1m4_09.tif' , ' AT3_1m4_10.tif':

f. Implement a function to blend a sequence of images into a single image using the following formula:

$$C = w_A A + w_B B$$

How could such a technique be used in a real application?

3. Use the image "balloon.bmp" and extract a patch of size 100x200 from it. After that, write a program to flip the extracted patch vertically and put it in the same coordinates of the original image. Repeat the same procedure for horizontal flip. Name the results as "balloon_vert.bmp" and "balloon_horiz.bmp" respectively and write them into output directory. The pipeline of the problem is depicted in Fig.5.

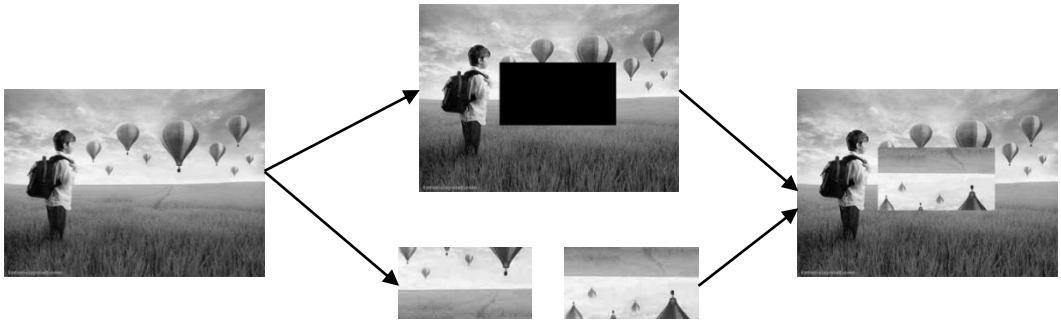

Figure 5 vertical flip

4. Load the image "bird.jpg". Each pixel of this image consists of 8 bits. As you know, each bit has a different significance proportional to its position in the 8-bit string. In this problem, you must represent the number of gray levels as a function of number of bits. To do so, use the bit-wise operators and remove the low significant bits one by one and show the results after each removal. Name the resulted image "P6_out1.bmp" to "P6_out7.bmp" respectively and save them in the output directory (P1_out1.bmp is the result with only one bit).

5. Write a function to resize any PxQ image to an NxM image without using MATLAB command 'imresize', Assume P>N and Q>M. Your fuction must get the image and its desired new size as input and gives the resized image as output. Verify your function with an arbitrary image.

6. Write a function to use a square image and a scalar value, say ***bsize***, and generate a coarse representation of the input image. In particular, the program must first partition the input image into *bsize-by-bsize* patches (blocks) and then the entries of each patch (pixels) must be replaced with the average value of that patch. Note that the size of the output image must not be altered.

a. Verify the performance of your program on "girl.jpg". To do so, set the value of bsize=4, 8, 16, 32

and name the results as "girl_2.bmp" to "girl_32.bmp", respectively. Write the output images in output directory. Your results should be similar to the ones presented in Fig.9. What can you infer from this exercise? Explain your observations.

Figure 9 coarse representation

7. Write a program to shear an image both in x- and y-directions and center the result. Your program should use an input image and two numbers, namely Xshear and Yshear, and generate the results. Use the following scheme as Forward Mapping:

$$\begin{pmatrix} x_{destinatio} \\ y_{destination} \end{pmatrix} = \begin{pmatrix} 1 & Xshear \\ Yshear & 1 \end{pmatrix} \left(\begin{pmatrix} x_{source} \\ y_{source} \end{pmatrix} - \begin{pmatrix} x_{center} \\ y_{center} \end{pmatrix} \right) + \begin{pmatrix} x_{center} \\ y_{center} \end{pmatrix}$$

As in the previous problem, it would be good idea to use Reverse Mapping due to the problem with holes in the destination image. Thus, for every pixel in the destination image, calculate the corresponding coordinates in the source image. To do so, you may use the following scheme:

$$\begin{pmatrix} x_{source} \\ y_{source} \end{pmatrix} = \begin{pmatrix} 1 & Xshear \\ Yshear & 1 \end{pmatrix}^{-1} \left(\begin{pmatrix} x_{destination} \\ y_{destination} \end{pmatrix} - \begin{pmatrix} x_{center} \\ y_{center} \end{pmatrix} \right) + \begin{pmatrix} x_{center} \\ y_{center} \end{pmatrix}$$

The resulting image should be the same size as the input image. If a source pixel lies outside the image, you should paint it black. The shear values Xshear and Yshear should be expressed as percentage of the image width and height respectively.

a. Verify your program on "cameraman.jpg" for different values of Xshear and Yshear. Name your resulting image "cameraman_shear.bmp" and save them in the output directory. Your results may be similar to the following Fig.11.

8. By using two images, we can obtain a composite image with a larger size, projected on the screen, as illustrated in Fig.12. However, due to the overlapping of the two projected sub-images, the composite

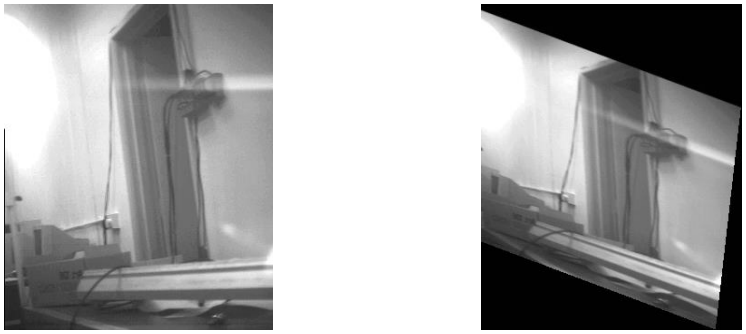

Figure 11 Image x- and y-shear

Figure 12 Image composite creation

image will have one region in the middle that has doubled gray scales. By finding maximum correlation between two overlapped regions, the coordinate of the overlapping can be determined. Write a program to find out the coordinate of the overlapping and modify the gray-scales of two input images—input1.jpg and input2.jpg—such that the composite image looks like being projected by one projector. Plot the correlation of two overlapped regions with respect to different overlapping positions. Name the result as "P11_out1.bmp" and write it into the output directory.

9. Please explain how an image is captured and stored in a digital camera. Provide your explanation in at least 10 and at most 20 sentences.

Good Luck,

Hamid Nasiri

Saeed Izadi