COMP 116
Jan 11, 2012

Assignment 1: Images as matrices of numbers, **Due Wed. 25th.Jan**

This assignment will give you practice in retrieving a file from a directory on your computer, using it in MATLAB, putting the results into a Word document, and formatting the results. It also shows an interesting data type: images that MATLAB can manipulate as matrices. It also introduces the concept of data types and converting between data types.

Take a few pictures of yourself or a couple of friends with your camera, or your phone, and move them to your computer. One way to do this from your phone is to simply email them to yourself. Name the pictures something like yourname.jpg, friend1.jpg, friend2.jpg. Put them in a directory where you can easily find them, probably in some directory on your C: or D: drive on your local computer, or on your H: drive if you are on a lab computer (your H: drive is accessible from all the labs on campus.) Use the pictures you take for the assignment, not the sample images I put up for you to play with.

Now, let's convert a color image to a black and white image! Do the following commands in MATLAB. Recall that % is a comment:

```
cd C:\           % "cd" means "Change Directory" so you are
                 % shifting attention to wherever it was you saved
                 % the pictures.
dir              % you should see yourname.jpg there
                 % if you can't see the file, you cannot load it
                 % with the next command.
x = imread('yourname.jpg');
                 % this "reads" your "im"age image file into a
                 % variable (a place to put stuff) labeled "x".
figure           % this command opens a new figure window
imagesc(x)       % a window with your distorted and perhaps
                 % discolored image should pop up.
axis equal;      % image should stop being distorted
xbw = rgb2gray(x);
                 % converts your color image to a BW image!
imagesc(xbw);    % let's display the B&W image
                 % yikes!  I will explain this later…
colormap(gray(256));
                 % image should now be in black and white
                 % Now use the "File -> Save As" command in the
                 % Figure window to save the new BW image!
```

The first two commands tell MATLAB to work with files on your C: drive, and lets you see the files. If you are working on a lab computer, you probably want to use the H: drive instead.

The 'figure' command simply opens a new figure. This is useful if you want to display multiple images. Without the 'figure' command 'imagesc' would display an image in its current figure window and would overwrite anything displayed previously.

The next command displays the image using a default image rectangle and current color scheme, which may not be appropriate, so the next command make the image have the right aspect ratio, and the last command makes the association between the numbers 0 – 255 and one of 256 levels of gray. (You could substitute flag(256) or hot(256) to see some other lists of colors, but make sure you come back to gray.)

We'll soon learn how to publish these figures, but for now, let's just copy into a Word document. In the figure window, select Edit >> Copy Figure.

Create a new Word document and Paste (control-V) to get the figure to appear. You can click on it to format it as you wish. (I often make it smaller or set the text to wrap around the image.) Go back to MATLAB, select the output from the command window and copy it (control-C), then paste it into Word.

Now, we can load another image as variable y and see what happens if you average the two faces. To allow fractions, we convert both images to real (double precision) numbers, rather than integers. Add a copy of this averaged face, and the names of the two people averaged, to your document.

The first command 'imread' reads in the image from disk as a matrix of UINT8 datatype values and assigns it to the variable name **x**. Each value in the matrix is an integer in the range 0 – 255. The output of the 'imread' command is fed as input to the 'double' conversion command. The second command 'double' converts each value in the matrix from an integer into a floating point real type. This conversion is necessary to give sufficient range for the addition and averaging operations. If you forget to use the double command, your resulting images may look unnaturally bright, white, and washed out. Which means you have just discovered that MATLAB uses saturation arithmetic, we will discuss this more later. *Note:* You'll see all the numbers in the image matrix listed if you leave off the semicolon (;).

```
y = imread('friend1.jpg');
% now go through the same process you went through above to
% convert the second picture to BWsc
x = double(xbw);
y = double(ybw);
figure
imagesc( (x+y)/2 )
axis equal
colormap(gray(256))
```

Load in a 3rd face and compute the average of all 3. Include the commands, names, and the image in your document. An incomplete sample solution is on Sakai already.

Add your name at the top of the document, save it, and submit the word document to Sakai. If you have any difficulty, Friday recitations are good times to seek help.