CICS-631-02 Computer Vision HW-06 See Dropbox for due date.

Segmentation

1. **Create a directory named HW06__LastName__FirstName**
   Do all your work in that directory. Then, when you submit your work, zip up the entire directory (from the directory name on down). This way when it is unzipped after downloading from MyCourses, we will see 38 directories, with unique names that are easy to understand.

   Use the LastName that matches the last name assigned to you in MyCourses, and the first name assigned to you in MyCourses.

2. **Changing your path:**
   When you read in an image, such as the `cameraman.tif` image, Matlab searches your **`path`**. The path is a list of executable directories. To make life easy for the grader, you are going to put your test images in a standard location, and change your **path** to include that directory.
   a. Put our images in a directory that at the same level as the directory HW06__LastName__FirstName, and name this directory `TEST_IMAGES`.
   b. In your programs, change the path to include this location using the command:
      `addpath( '../TEST_IMAGES' );`
   c. Do not re-submit the test images. The grader will have copies of them.

3. Write one function that runs your entire homework. Pause between each part of your homework, using `pause()`. This stops execution while the grader looks at your results. Then, when the grader hits return, the program continues.

   a. The function is to be in a file named `HWNN_Lastname_Firstname_MAIN.m`
      Where NN is the homework number. In this case 06.
   b. This function runs all the other parts of your homework:

```
HW06_LastName_FirstName_FIND_RASPBERRIES( 'TBK_RASPBERRIES__smr.jpg'  );
pause(3);
HW06_LastName_FirstName_FIND_GRAFFITI( 'Image_of_Graffiti.jpg' );
pause(3);
HW06_LastName_FirstName_FIND_AIRPLANE( 'TBK_Airplane.png' );
```

4. Remember to comment your code well to show that you understand it. Write your comments in your own words. Lack of comments will cost points. The professor may just look at your code, or might just look at your write-up.

5. **Discussion:**
   General discussion is good. We are looking for evidence that you understand the approach to what you are doing. In this case either matching a model of an object, or avoiding a model of the background.

   Some students have been doing the bare minimum, such as answering questions with "no" or "see code." Or they answer only the questions suggested in the homework. The grader is free to assign a minimum grade to such answers.

   Imagine that you are doing this to explain it to next semester's class. Feel free to exceed expectations.

6. Estimate how long it will take to complete this assignment (½ pt)

7. **Problem one – Raspberry Identification:** (3 pts)
   Create a function, in a separate file, called:
   `HW06_LastName_FirstName_FIND_RASPBERRIES( filename )` which reads in one image.
   Based on a model that you write for classifying pixels as raspberries (based on the color or surrounding colors), this classifies the pixels in the image as raspberry (1) or not-raspberry (0).

   You will develop this classifier by collecting many pixels on raspberries from the provided images.
   The test images will be very similar.

   Your program may change the color space of the image to any other color space you want to use.

   The program displays the input image, then pauses for 3 seconds. Then the program displays the resulting image, then it pauses for 5 seconds. The final result should be white where there are raspberries, and black everywhere else.

   In your write-up, describe your algorithm. This means: give details such as: Did you model the foreground, the background, or both? Describe what features you used? How did you decide on those features? How did you separate the foreground from the background? What distance metric did you use? What color space did you use?

   Write in complete paragraphs. Do not give sketchy answers. The professor wants to see a full write-up.

   A student from next semester, reading your write-up, should be able to understand what you did in your code. Pseudocode, algorithms, block diagrams, and flowcharts are helpful. In the write-up, show your input image, and your output image. Use imwrite( ) to write out your image and pull it into your write-up.

8. **Problem two – Graffiti Identification:** (4 pts)
   Create a function, in a separate file, called:
   `HW06_LastName_FirstName_FIND_GRAFFITI( filename )` which reads in one image.
   Based on a model that you write for classifying pixels as graffiti (based on the color or surrounding colors), this classifies the pixels in the image as graffiti (1) or not-graffiti (0).

   You will develop this classifier by collecting many pixels on the graffiti from the provided images.
   The test images will be very similar, but a different image.

   Your program may change the color space of the image to any other color space you want to use.

   The program displays the input image, then pauses for 3 seconds. Then the program displays the resulting image, then it pauses for 5 seconds.

   In your write-up, describe your algorithm. (As described in the last question.)

9. **Problem three – Generating an Airplane Silhouette:** (2 pts)
   Create a function, in a separate file, called:
   `HW06_LastName_FirstName_FIND_AIRPLANE( filename )` which reads in the image TBK_Airplane.png.

   The image contains fixed-pattern noise, which comes off the sensor. If you zoom in on the image, and magnify a section, you will see that there is a pattern to the noise. So, you need to find a way to minimize that noise.

   Once you remove the noise, build a model to classifying pixels as either airplane (1), or background (0). Your program should display a black and white image, where the image is white where there is airplane, and black where there is background.

   Your program may change the color space of the image to any other color space you want to use for classification. You may wish to do noise cleaning in one color space, and classification in a different color space.

   The program displays the input image, then pauses for 3 seconds. Then the program displays the resulting image, then it pauses for 5 seconds.

   In your write-up, describe your algorithm completely. (As described previously.)

10. Report how long it took you to complete the homework. This is for you to practice estimating, so be honest. (½ pt)