School Computing, Faculty of Engineering
Universiti Teknologi Malaysia

SECV 3213 FUNDAMENTAL OF IMAGE PROCESSING

Assignment 2
Image Restoration

Requirements

- This assignment is a group task. Form your group on your own comprises THREE (3) members.
- This assignment requires you as a group to restore THREE (3) images with noise, to become the same as the original or near original.
- You should apply the knowledge learned from the past topic about image enhancement in spatial domain such as image filtering, convolution, etc.
- Propose TWO (2) solutions for the image restoration. You will apply both solutions to the same images. Each solution should be implemented in a separate python script.
- Write a report about your work.
- Present your work as a group in a video.

Datasets and Tool

- This assignment comes with image datasets organized into two folders named **Noise** and **Original**. You can find the link for the dataset on eLearning.
- Your task is to restore the Noise images. The Original images are the expected restored images. You will be using them as references to compare with the restored images from your programs.
- Each group needs to choose only ONE (1) set consisting of THREE (3) images. Each group will have different set of images. For example, if you choose Set 1, that means your images are {ImgA1, ImgB1 and ImgC1}
- You are also provided with a python script named, **similarity.py**, which you will be using it to compare your restored images with the reference images.
  Please watch the attached video to learn how to use this program.

Code

a. Write two python programs, i.e., one program for each solution.

b. Each program should be able to do the followings:

   i. Read an input image file (a noised mage)

   ii. Perform the image enhancements to restore the image

   iii. Write the output image to a file

   iv. Display the input and the output images on the screen

c. Design each program such that it accepts the input and output image files as command-line parameters. For example, the program may be run from the command prompt as follows:

```
restore1.py  Noise/ImgA1.jpg   Restored/ImgA_solution1.jpg
```

where Noise and Restored are the folders for the input and output images, respectively.

Notes: you can find an example how to deal with command-line parameters from the program **similarity.py**.

Report

The report should include:

a. Problem analysis. Observe the images and identify the problems.

b. Proposed Solutions (two solutions). Each solution should include:

   i. A diagram showing the process flow or pipeline of the proposed solution.

   ii. Explanation about the proposed solution.

   Explain each process in the proposed solution with the parameters used. For example, you apply the average filter with size 3x3, etc.

   iii. The results of each proposed solution on each image.

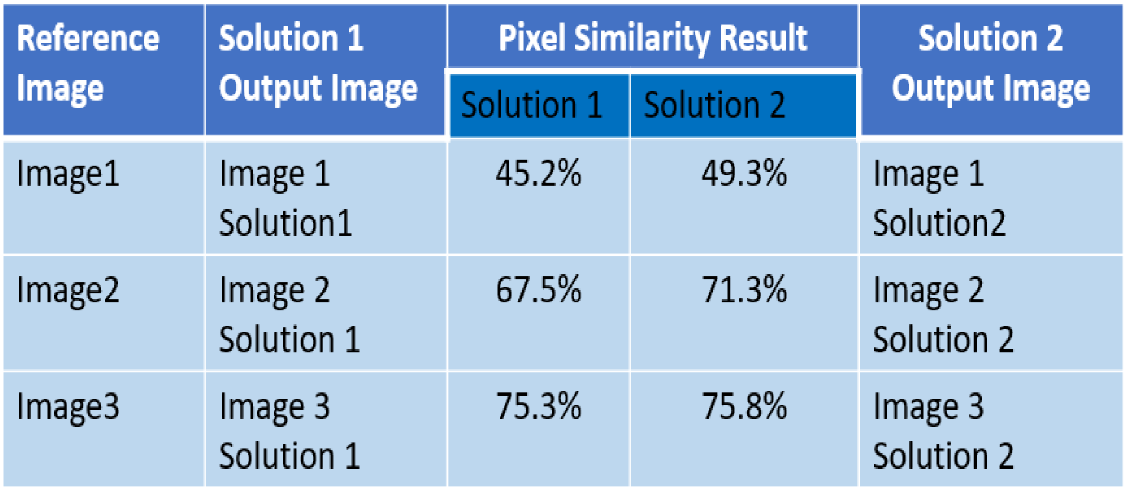
   That means, you will apply proposed solution 1 to image 1, image 2 and image 3, and repeats for proposed solution 2 to the same images.

   iv. Result Analysis.

   Explain your result. quantitatively and qualitatively.

c. Analysis of the proposed solutions:

i. Present the part in a table of comparison like below and explain your finding.



Reference Image	Solution 1 Output Image	Pixel Similarity Result		Solution 2 Output Image
		Solution 1	Solution 2	
Image1	Image 1 Solution1	45.2%	49.3%	Image 1 Solution2
Image2	Image 2 Solution 1	67.5%	71.3%	Image 2 Solution 2
Image3	Image 3 Solution 1	75.3%	75.8%	Image 3 Solution 2

d. Conclusion

Presentation Video

- Prepare a 15-minute presentation video for each group.
- Host the video on YouTube.
- Include the video link in your report.
- The video should show the followings:
  - Run of each python script. Use only one image to showcase this.
  - Presentation of each solution including the results.

Deliverables

In this assignment, you will deliver the following items:

- Python source code (2 programs)
- Report (in PDF)
- Video link (include in your report)

  Zip the pdf and source codes

Assessment

This exercise carries **14%** weightage for the final grade of this course. The breakdown weightage is as follows (out of 100):

- Overall (15)	
○ Completeness of the application	4
○ On-time submission	4
○ Follow best practice of programming (e.g., neat code, proper naming convention, modular, etc.)	7
- Code for Solution 1 (20):	
○ Input and Output. *Reading input images and writing output images, and show images on the screen*	5
○ Image Enhancements *Implementation of the proposed solution consisting of several image filtering methods.*	15
- Code for Solution 2 (20):	
○ Input and Output.	5
○ Image Enhancements	15
- Report (30)	
○ Problem analysis	5
○ Proposed Solutions	10
○ Analysis of Proposed Solutions	10
○ Conclusions	5
- Video Presentation (15)	
○ Run of each python code	5
○ Presentation of proposed each solution	10

Submission

- Deadline: 1 week time (Will be announced in class)
- Only one member from each pair needs to do the submission.
- Submission must be done on eLearning.
- Submit the zip file.

Plagiarism Warning

You may discuss with others and refer to any resources. However, any kind of plagiarism will lead to your submission being dismissed. No appeal will be entertained at all.