AE2IIP Coursework 2019-2020

This is the assessed coursework for IIP. It is worth **40%** of the final mark.

Deadline: 18 May 2020 16:00

Task Description and Guidelines

Special note: put **rng('default');** at the beginning of each of your main functions.

Task 1: Implement and compare spatial filters. (**22 marks**)

Requirements & Guidelines

- Load lena.jpg into matlab, and convert it to gray-level image.
- Add the following noise to the gray-scale image of lena, separately. You should provide the distorted image in the report.
  - Gaussian noise with $\mu=0$, $\sigma=20$.
  - Uniform noise in [-a, a], where a = 50.
  - Add 10% of salt & pepper noise.
- Implement the following spatial filters:
  - 3×3 mean filter.
  - 5×5 Gaussian filter with $\sigma=1$.
  - 3×3 median filter.
  - 3×3 anisotropic filter with the similarity function of $(D-d)/D$.
  - 5×5 bilateral filter with with $\sigma_s=1$ for space Gaussian and $\sigma_r=10$ for range Gaussian.
- Apply five implemented filters on the three distorted images you create above.
  - Provide the 3 × 5 = 15 filtered images.
  - SNR is calculated as follows:

$$SNR = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(f'(x,y))^2}{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(f'(x,y)-f(x,y))^2}$$

where $f'(x,y)$ is the filtered image and $f(x,y)$ is the clean reference image.

In this task, $f(x,y)$ is the clean gray-level image of lena, and $f'(x,y)$ the filtered image. In matlab, you may use the following function to calculate SNR:

SNR = snr(img_filtered,img_filtered - img_clean);

where img_filtered is the filtered image and img_clean is the clean image.

- ❖ Record down the SNRs for different filters on three distorted images in the following table:

Noise	Mean filter	Gaussian filter	Median filter	Anisotropic filter	Bilateral filter
Gaussian					
Uniform					
S&P					

- ❖ From the image visual quality and SNRs in the table above, discuss which filter is more effective to the Gaussian noise, uniform noise and salt & pepper noise, respectively. Justify your answer.

- Briefly explain the effect of filter parameters on the filtered image, e,g, the size of filter, σ in Gaussian filter, the similarity function in anisotropic filter, and σ_s, σ_r in bilateral filters. Use the experimental results to support your analysis, by twisting these filter parameters and applying mean, Gaussian, median, anisotropic and bilateral filters on the 3 distorted image. You do **NOT** need to try all the combinations on all the filters on all the distorted images. But your analysis and supportive experiments should be convincing.
- You should implement one main function that calls other functions to generate all the results, e.g. task1main().

Marking Criteria

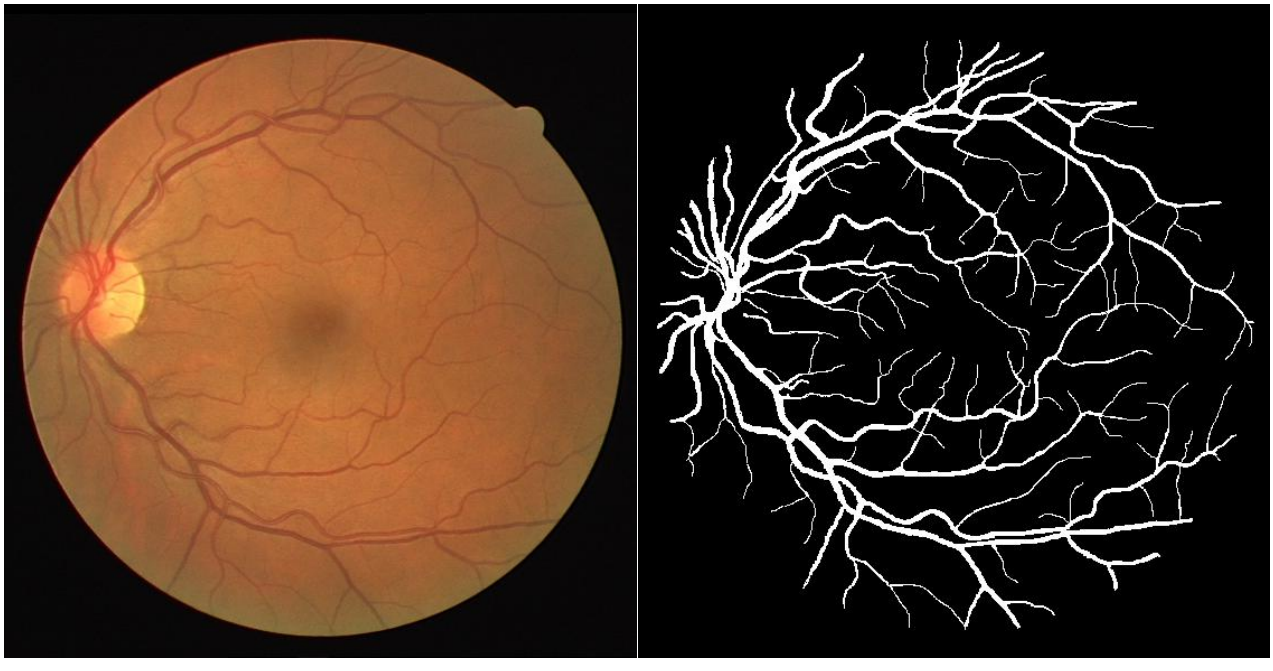
You should include the explanation of each filter you implement, the distorted images and the filtered images in your report. Marks have been allocated to each step. Marks will be given by considering:

- Whether you correctly implement the distorted image. (**3 marks**)
- Whether you correctly implement the filter. (**5 marks**)
- Whether you successfully obtain filtered images and derive SNR for each filtered image. (**3 marks**)
- Whether your arguments for the choice of filters are reasonable. (**3 marks**)
- Whether your analysis on the effect of filter parameters is reasonable. (**5 marks**)
- Whether your codes are well documented, including comments, and explanation of variables at the beginning of each function. (**3 marks**)

Task 2: Segment the retina blood vessel. (**18 marks**)

Retina blood vessel segmentation is challenging. Given the image on the left, it is very difficult to accurately segment the blood vessel as shown on the right, especially the thin ones. You are free to design your own algorithm to segment the blood vessel, down to per pixel level.

NOTE: You are not required to achieve the state-of-the-art results, but you should achieve satisfactory results.



Requirements & Guidelines

- 40 retina images are provided, together with the label images indicating where the blood vessel is, and the mask image masking out the background of retina image.
- You should **test your algorithm on "retina_images/1.tif" only**, and show the segmentation results for this image. You should **never** use "**label_images/1.tif**" when you design your algorithm. All other images being provided can be used as training images when you design your algorithm. (Although training may not be necessary.)
- When test your algorithm, you should compare the segmentation results of your algorithm with the ground-truth image (label_images/1.tif). You should report the following:
  - The percentage of **blood vessel** pixels that is being correctly classified as **blood vessel**. Denoted as P.
  - The percentage of **background** pixels (**only consider the region in the mask**.) that is being correctly classified as **background**. Denoted as N.
  - The percentages of pixels are being correctly classified. (**only consider the region in the mask**). Denoted as T.
  - For example, assume in the mask we have 100 pixels, 10 pixels are retina and 90 pixels are background. The segmentation result is that 12 pixels are retina (only 8 are true retina and 4 are false retina) and 88 pixels are background. (only 86 are true background and 2 are false background.) Then P = 8/10 = 80%; N = 86/90 = 95.56%; T = (8+86)/100=94%.
- You may consider the following filters. It is up to you to choose which one to use. You may use other filters if appropriate. **You can use FILTERS from internet resources, with proper acknowledgement**. You should **NOT** use the complete (or near complete) solutions you may find from internet resources.
  - Laplacian of Gaussian filter.
  - Difference of Gaussian filter.
  - Canny filter.
  - Match filter.

- ❖ Gabor filters.
- ❖ Other filters.

- You may directly process the filtered images to obtain the per-pixel classification results of the retina image, or you may try to combine the results from several filters.
- When discussing the pros and cons of your algorithm, you should compare it against one simple baseline algorithm.
- You should implement one main function that calls other functions to generate all the results, e.g. task2main().

Marking Criteria

You should include the explanation of your algorithm, the intermediate results and the final results in your report. Marks will be given by considering:

- The quality of your per-pixel level segmentation results for the first image. You should try to maximize both P and N. (**6 marks**)
- The explanation and justification of your algorithm. (**6 marks**)
- Discuss advantages and disadvantages of your algorithm. (**4 marks**)
- Whether your codes are well documented, including comments, and explanation of variables at the beginning of each function. (**2 marks**)

Reference and Plagiarism

You can only use the build-in functions of matlab, or the codes you implement by yourself, unless otherwise stated.

You are reminded of the School's Policy on Plagiarism.

Report

The page limit is **20** for the report. Font size 12. All images on the report should be large enough for visual inspection of the image quality.

Late submission

The standard late submission policy applies, i.e. 5% deduction of the total mark for every 24 hours. (including weekends, holiday.)

How to submit

Online submission via Moodle. You should zip all the matlab files, the PDF file of your report in **ONE** zipped file. (**You do not need to submit the resulting image files, but your program should show the resulting images with proper caption, e.g. using imshow(). In your report, you should include the resulting images and necessary images for the intermediate steps. You do not need to submit the dataset provided in Task 2**.)

You should name the zip file using your name and student ID, e.g. "DongChen_1234567.zip". Please note that every next submission overwrites the files in the previous one, so if you submit several times, make sure that your **last submission includes all the necessary files, can be unzipped and executable**.

You do **NOT** need to submit the hardcopy of your report to the faculty office.