Project I. Image Filtering and the Fourier Transform (Python)

Due Monday 02/11/2019

CMPE 685 Teaching Assistant: Nilesh Pandey np9207@rit.edu

Objectives
The objectives of this project are:
Understand and implement image filtering in the space domain.
Learn to compute and interpret the 2-D Fourier transform of an image.

Project Report:
A pdf version of the report (pdf and zipped overleaf project), images (zipped), and all code (zipped) should be uploaded on mycourses by midnight of the due date. The project report in paper form (without the code) is due at the beginning of the next class. There is no page limit, but the report should include the following sections: Introduction, a Section for each Topic that includes Methods and Results as subsections, Discussion, References (include URLs of images and other references as appropriate) and Appendix with code listing. Use color figures in the pdf version of the report and grayscale images in the figures of the printed version. Make sure you maintain the image aspect ratio when displaying images in the figures. The report will be graded based on methods used, results obtained, interpretation of results, answering of questions, clarity of presentation, discussion and English.

Report format: You are asked to use latex on www.overleaf.com for your report. You can use any single column template on overleaf. If you are not very familiar with latex, it may help you to use https://www.codecogs.com/latex/eqneditor.php for writing equations.

Images and Code

- You should write your own Python code for this project. Do not share any code with others and do not download any code from the web.
- The required libraries for this assignment is numpy, matplotlib, opencv, skimage, scipy. See the document `CMPE 685 Project 0 Python Getting Started` for more information on installation and tutorials.
  If you want to use the Computer Engineering Cluster, see the document `CMPE 685 Cluster Guidelines`

Computer Vision Datasets:
You may use your own images or any standard images used in Computer Vision. Select pictorial images with a main subject. Do not use images with text, cartoons or graphics. Many links to datasets are found below, and it is good to become familiar with some popular datasets.
http://www.imageprocessingplace.com/root_files_V3/image_databases.htm
http://www.cvpapers.com/datasets.html
http://riemenschneider.hayko.at/vision/dataset/

1. Filtering in the space domain

a) Process a grayscale image using a High Pass filter of your choice and display the image and result. Write your own code for filtering. Explain how you process pixels at the image boundary. Calculate the compute time for the filtering process.

b) Read a grayscale image and filter it using a Gaussian Low Pass filter (pick a filter size that is at least 5x5). Write your own code for filtering. Display the result. How does repeated application of the LP filter affect the resulting image?

c) Write a program that processes a grayscale image using the Sobel edge detector masks. Display the result for the horizontal and vertical edges (after taking abs and thresholding) and their sum (after taking abs).

2. Filtering a noisy image

a) Add salt and pepper noise to a grayscale image using the skimage/opencv library.
Add Gaussian noise to a grayscale image using the skimage/opencv library. The noise should be zero mean and have variance large enough for the noise to be visible in the noisy image.
Find the signal to noise ratio (SNR) in dB, using the definition $SNR_{dB} = 10\, log_{10}\left\{\frac{\sigma_i^2}{\sigma_n^2}\right\}$ where σ_i^2 is the image variance and σ_n^2 is the noise variance.

b) Filter the noisy images in 2.a) using a low pass filter and median filter (write your own implementation) and display the results. Choose the filter parameters to deal with the added noise while preserving the image structure as much as possible. What is the effect of low pass and median filtering on the two types of noisy images?
(Image filtering should be done in Gray Color Space, Write the code accordingly.)

c) Process the original and noisy images in 2.a) using the Sobel edge detector and display the results using the subplot command. What is the effect of noise on the output of the Sobel edge detector? Propose a method for reducing the effects of noise when detecting edges.

3. The 2-D Fourier Transform

a) Read a grayscale image and compute its 2-D FFT using the np.fft.fft2 command.
Display the magnitude of the 2-D FFT. Where is the origin located?
Display the magnitude of the 2-D FFT after using the np.fft.fftshift command. What is the effect of np.fft.fftshift on the location of the origin?
Display the 2-D DFT magnitude after taking its log. Which method of display do you prefer (with or without log) and why?

b) Take the inverse Fourier Transform using np.fft.ifft2 and compute the Mean Square Error (MSE), between the original and the reconstructed image, where

$$MSE = \frac{1}{MN}\sum_{x,y}\{(f(x,y) - g(x,y))^2.$$

c) Repeat step b) after zeroing out all frequencies outside a radius of N/3.
Repeat step b) after zeroing out all frequencies outside a radius of N/4.
Repeat step b) after zeroing out all frequencies outside a radius of N/8.
Repeat step b) after zeroing out all frequencies outside a radius of N/16.
Use the subplot command to display the reconstructed images and corresponding DFT magnitude. Tabulate the MSE for all cases and comment on the visual appearance of the reconstructed images.

4. The Magnitude and Phase of the 2-D DFT

a) Read two grayscale images (use faces, animals or objects that are well separated from the background). Display the magnitude and phase of their DFT.

b) Reconstruct each image by using the magnitude of the other image and its own phase. Comment on the visual quality of each reconstruction and compute the MSE in each case.