MT Assignment#2 Report

Image Convolution

資工三 406410035 秦紫頤

Setup

Components

There are:

- 2 folders: result, src
- 3 files: hw2.m, hw2.mlx, MT-hw2_report

in this assignment folder.

- **result:** This folder contains all the results generated from the program. There are 2 folders and 3 files in this folder.
    - **hsize: 3** output images after Gaussian blur with different hsize
    - **sigma: 3** output images after Gaussian blur with different sigma
    - **edge_detection.jpg:** output image after convolution using edge detection filter
    - **psnr.csv:** peak signal-to-noise ratio value of original value and different convolution condition
    - **unsharp.jpg:** output image after convolution using the unsharp filter
- **src:** The source image to do convolution. Inside this folder only contains one image **"shiba.jpg"**
- **hw2.mlx, hw2.m:** The program for this assignment
- **MT-hw2_report:** the report of this assignment

Prerequisite

1. MATLAB: R2019b (recommended)

Execution

1. Open **MATLAB**
2. Open this assignment folder from MATLAB
3. You can either execute **hw2.m or hw2.mlx** to get the result
4. All the result will be in **"./result"**

Method Description

I split this assignment into 3 parts. The detail is as follow:

Part1: Different hsize Gaussian filter

1. Read the image and do **zero-padding**

a. Open the folder that the image is in (**“./src”**)
b. Read the image (**“shiba.jpg”**)
c. Convert the data type to double
d. Do zero-padding with **padarray()**

2. Gaussian filter with **hszie=3x3** and sigma=1
    a. Create a Gaussian filter with **fspecial()**
    b. Convolution **(sum of element-wise matrix multiplication**) using for loop sweeping the whole image matrix with a layer of padding as margin
    c. Output the resulting convolution image
    d. Calculate the PSNR between the original image and the image after the convolution using **psnr()**
    e. Write the PSNR result to an array
3. Gaussian filter with **hsize=5x5** and sigma=1
    a. Add one more layer of zero-padding as the margin around the image
    b. The rests are the same as **Step2**
4. Gaussian filter with **hsize=7x7** and sigma=1: Same as **Step3**

Part2: Different sigma Gaussian filter

1. Gaussian filter with hsize=3x3 and **sigma=1**
    a. Using the image with only one layer of zero-padding as the margin
    b. The rests are the same as **Step2 in Part1**
2. Gaussian filter with hsize=3x3 and **sigma=5**: Same as **Step1** but remember to change the sigma value in **fspecial()**
3. Gaussian filter with hsize=3x3 and **sigma=10**: Same as **Step2**
4. Write the whole PSNR table to CSV file
    a. Transform the PSNR array to a table using **array2table()**
    b. Name the variable of the table
    c. Write the table to a CSV file (**“./result/psnr.csv”**)

Part3: Apply unsharp and edge detection mask

1. Unsharp mask
    a. Use the unsharp mask from the assignment SPEC $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$
    b. Doing convolution
    c. Output the resulting image
2. Edge detection
    a. Use the edge detection mask from the lecture slide **Unit3 p.61**
    b. Doing convolution
    c. Output the resulting image

Results

1. The results of (a) are in the folder **"./result/hsize"**. There are 3 images corresponding to three conditions hsize=3x3, 5x5 and 7x7. Show the three output image as follow Figure (a), Figure (b) and Figure (c)

Figure (a)
hsize_3x3.jpg
psnr=33.8678

Figure (b)
hsize_5x5.jpg
psnr=32.5891

Figure (c)
hsize_5x5.jpg
psnr=32.4933

2. The results (b) in the folder **"./result/sigma"**. There are 3 images corresponding to three conditions sigma=1,5 and 10. Show the three output image as follow Figure (d), Figure (e), Figure (f)

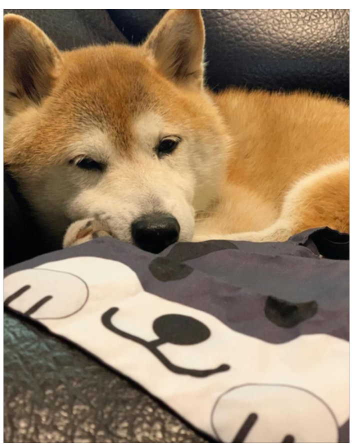

Figure (d)
Sigma_1.jpg
psnr=33.8678

Figure (e)
sigma_5.jpg
psnr=32.52

Figure (f)
sigma_10.jpg
psnr=32.4847

3. Complete psnr result can be found in **"./result/psnr.csv"** and is shown in Table (a) below as well

TEST_ITEM	hsize=3x3, sigma=1	hsize=5x5, sigma=1	hsize=7x7, sigma=1	hsize=3x3, sigma=1	hsize=3x3, sigma=5	hsize=3x3, sigma=10
PSNR_VAL	33.8678	32.5891	32.4933	33.8678	32.52	32.4847

Table (a)

4. Compare and discuss the result above
   a. PSNR value: PSNR is the abbreviation of peak signal-to-noise ratio. It is in decibel and between two images. PSNR is used as the quality measurement between the original image and the compressed image. Higher PSNR means the compressed image is in better quality, the reconstruction is better.
   b. Compare the result of different hsize. From the output images, I can't see much difference. But I can tell a huge difference from the image size. Table (b) below shows the original image and 3 compressed images image size. The size reduces more when the hsize is bigger which means image compresses more. Gaussian filter is a smoothing technique for image and also if the convolution mask is larger the resulting image will be smoother which indicates the image size will be smaller. The result makes sense. Also while hsize becomes large PSNR value is smaller, this means if the image is more compressed PSNR value is smaller.

IMAGE	shiba.jpg (original image)	hsize_3x3.jpg	hsize_5x5.jpg	hsize_7x7.jpg
SIZE	196.9kB	80.0kB	73.1kB	72.6kB

Table (b)

   c. Compare the result of different sigma. Same as b I can't see much difference from the output image so I look at the image size of the three output image. The size is shown in Table (c) below. The image size did reduce. Sigma value is the standard deviation of the gaussian filter. Standard deviation is basically equivalent to radius so larger standard deviation equivalent to the larger radius and can think of as larger filter area which yields a smoother resulting image. Same as b if the sigma is larger than PSNR value will be smaller.

IMAGE	shiba.jpg	sigma_1.jpg	sigma_5.jpg	sigma_10.jpg
SIZE	196.9kB	80.0kB	76.8kB	76.7kB

Table (c)

   d. But if comparing hsize and sigma, I found out that changing hsize compressed the image more dramatic than changing sigma. Which I don't quite know the reason why. To conclude the result, PSNR value measures the quality of the compressed image. If the hsize and sigma are larger than PSNR value will be smaller cause the image quality won't be that great compared to the original image but the image size will be smaller. This is kind of the compromise between image size and image quality.
5. The resulting image after applying **unsharp filter and edge detection filter** are in **"./result/unsharp.jpg"** and **"./result/edge_detection.jpg"** also shown in Figure (g) and Figure (h) below.

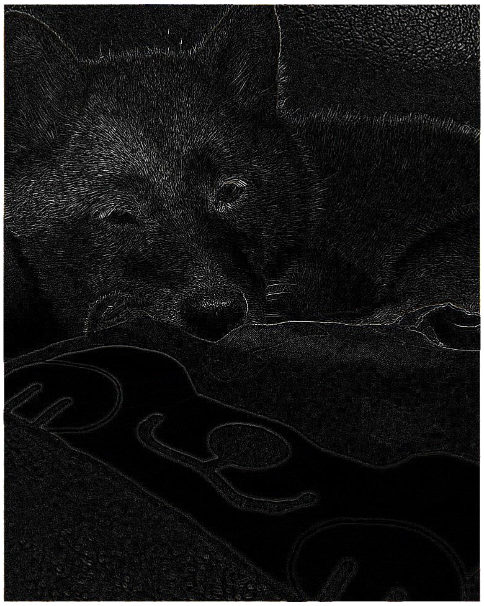

Figure (g) unsharp.jpg

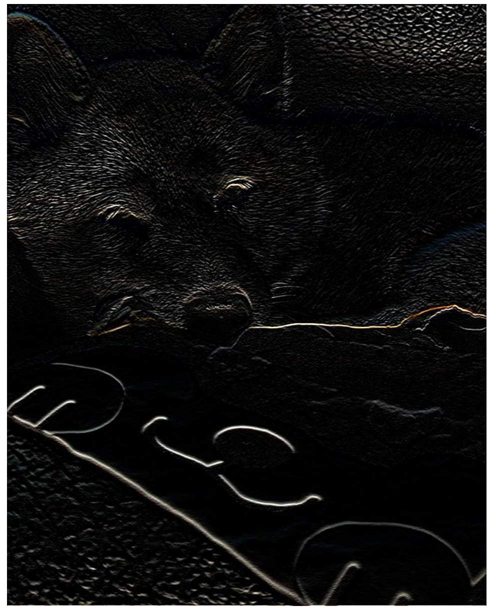

Figure (h) edge_detection.jpg

Discussion

- Actually, I'm quite confused about the result of this experiment. Gaussian blur is a technique of smoothing the image but I can't feel the smoothing (blur) effect from the resulting image, so I don't know if it is usual. But I did check the pixel value of the original image and the images after convolution, they are really not the same and also PSNR and the image size proved the images are actually compressed.
- I think if really want to make the image be more blurry then there are two ways:
  - Apply convolution more times
  - Make the hsize and sigma larger at the same time
- For the unsharp and edge detection part, I can tell big difference from the output image. They are both black and white because the coefficients in both filters are either very big or very small to highlight the characteristic of the image. Output image of unsharp filter looks more detail. You can tell it from the carpet and the dog's furs. Output image of edge detection filter did highlight the edges can tell from the carpet.

Problems and Difficulties

I think I didn't encounter any problem or difficulty while doing this assignment. The instruction is clear and concise and pretty easy to understand. But not using some easy use MATLAB built-in function did make the code in the program a bit longer than it should be (many for loops). But this gives me a clearer picture of how convolution actually works.