Omer Dogan
150130208
https://github.com/omerdogan3

Assignment 1

In this file, you should enter the name of the image, then the size of the filter. When you hit the enter, it manipulates the input image then shows you corresponding output. If you enter a non-existing directory name or invalid filter size, program will not run properly. In this assignment, i have converted all images to double. I have used “im2double” built-in function to do it. I added clc; and clear; commands at the beginning of the all files to avoid from any kind of misunderstood.

Q1.m

After you entered the name of the image and the size of the filter, a variable named tmp_image will be created as an extended array. Extention size changes according to user input.

```
tmp_image = zeros(size(original_image,1) + fix(filter_size/2),
size(original_image,2) + fix(filter_size/2));
%if you enter 7 as the filter size, it extends the original image 2
pixels from each sides.
```

Then i have implemented the convolution filtering to the tmp_image with nested for loops. I divided the temporary total value with the square of the filter_size value, as a result it multiplies every pixel with 1/9 for 3, 1/25 for 5, 1/49 for 7 and so on.

As an output for this implementation, we should get more blurred image if we increase the size of the filter. And as we can see above, output of the operation as we expected.

User Input & Output

```
Name of the image: Cameraman.bmp
Please enter size of the filter: 11
```

Omer Dogan
150130208
https://github.com/omerdogan3

Q2.m

To apply subtaction operation, i have coppied and pasted source code from Q1.m file. To apply subtraction operation, it is required to have same sized matrices. For this reason, i have resized the original_image array.

As an alternative way, minus() built in function should be used for this question. However, i have created an independent array that stores result of the subtraction operation, then i have implemented nested loops and subtracted each pixel from another.

If we increase the size of the filter, the image will be more blurred, for this reason our output of the subtraction operation will be less detailed. In the other words, it will detect less edges.

User Input & Output

Name of the image: >> Cameraman.bmp
Please enter size of the filter: 3


Name of the image: >> Cameraman.bmp
Please enter size of the filter: 11


Omer Dogan
150130208
https://github.com/omerdogan3

Q3.m

To apply shifting operation, we have to resize our filter according to our shifting direction and the magnitude of the shifting. To run this part of the assignment, you should only enter the name of the image, you do not need the enter size of the filter.

- To shift the image up by 2 pixel using a filter, i have used [0,1,0;0,0,0;0,0,0] array as a filter.
- To shift the image right by 3 pixels using a filter, i have used [0,0,0,0,0,0,0,0;0,0,0,1,0,0,0;0,0,0,0,0,0,0,0] array as a filter.

User Input & Output

Name of the image: >> Cameraman.bmp

Name of the image: >> Lena.bmp

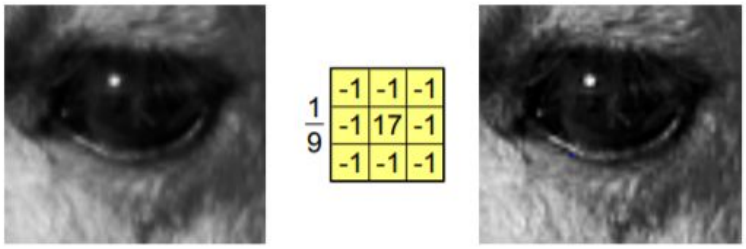

Omer Dogan
150130208
https://github.com/omerdogan3

Q4.m

To sharpen the image it is required to follow these steps:

- blur image with the mean filter
- subtract the blured image from the original image. As a result, we will have only details in the image
- add subtracted image to the original image.

However, to have better performance, we can handle these operations in one step with using mutability feature of the convolution operation. Basically, i have implemented the operation below.



Sharpen (identity minus mean filter)

In this part of the assignment, it generates an identity matrix according to your filter size. Then it has been subtracted from the mean filter than then applies convolution to the image. At the end, the output is as we expected.

User Input & Output

Name of the image: Cameraman.bmp
Please enter size of the filter: 3

Omer Dogan
150130208
https://github.com/omerdogan3

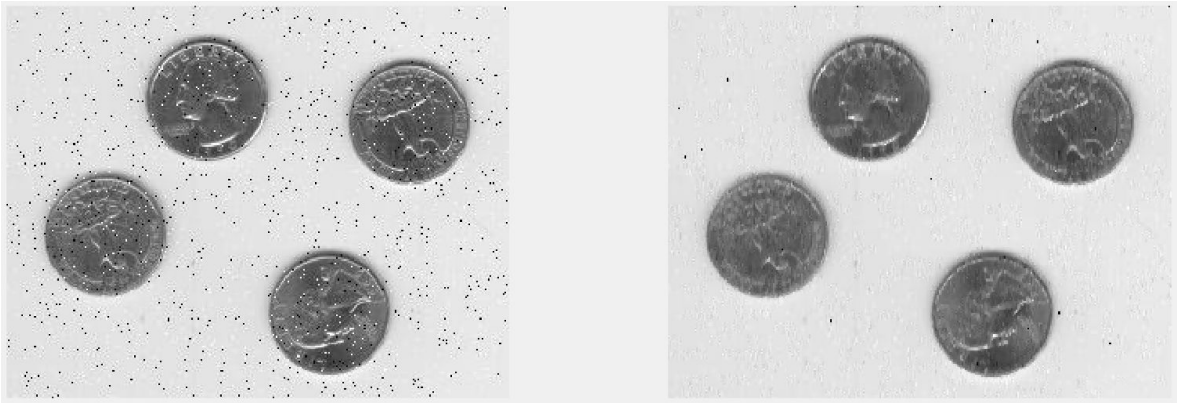
Q5.m

To implement medium filtering operation, i have used a temporary window to store all of the pixels in the current iteration of the convolution operation. In this step, basically, i have used same nested loop implementation. However, instead of addition and division operation, i took the medium element in the temporary window and assigned it to the result array.

In this part of the assignment, you are required to enter not only the name of the image, but also the size of the filter. As we can see above, if we use bigger filter sizes, the clearness of the picture is disappearing as we expected.

User Input & Output

Name of the image: salt_pepper.jpg
Please enter size of the filter: 3



Name of the image: salt_pepper.jpg
Please enter size of the filter: 11