Labs 2 - Image filtering and histogram

Submitted by **Meldrick REIMMER and Selma BOUDISSA**

Contents

1 Problem 1

In this exercice on the image of the Anthe Galaxy. We had to test different algorithm on it, here after the code with the algorithms.

```
%% Problem 1

% Read and display the original image
  figure()
  subplot( 2,2,1)%display the image in the same figure
  Image = imread ('galaxy.png'); %Load the image
  imshow (Image)%display the image
  title('Original Image')

% Test Robert Gradient
  subplot(2,2,2)
  rob= edge(Image, 'Roberts');
  imshow(rob);
  title('Robert Gradient')

 % Test Sobel
  subplot(2,2,3)
  SobelTest = edge(Image,'Sobel');
  imshow(SobelTest);
  title('Sobel Gradient')

% % Test Laplacian
  subplot(2,2,4)
  Laplacian = edge(Image,'log') % Apply Laplacian filter
  imshow(Laplacian);
  title('Laplacian')
```

prob1.m

In the figure above the result of the comparaison of all the algorithms.

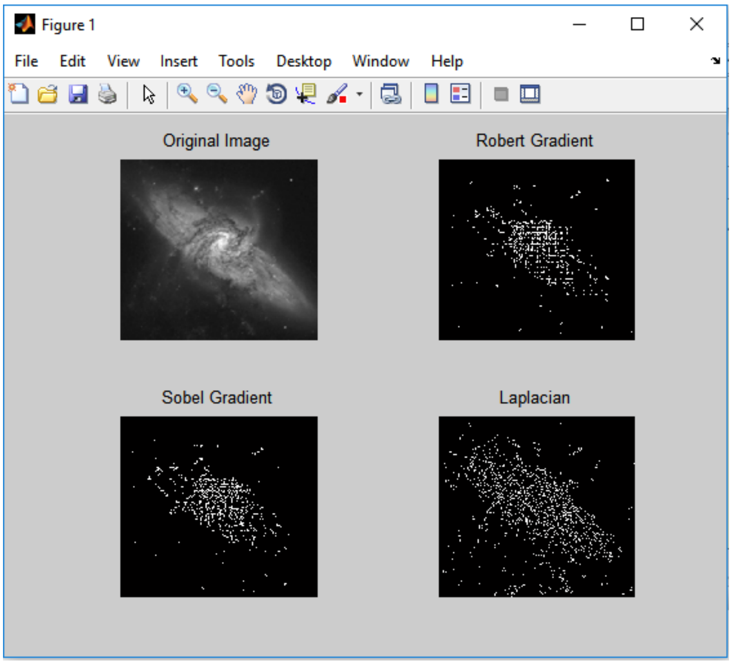

Figure 1: Comparaison of the effect of the different algorithms

Let's analyze the influence of each gradient methods on the original image of the Anthe Galaxy. These methods are used to find the edges of an object in images using any methods (Roberts, Sobel or Laplacian). For the method parameter the Edges Detection block finds the edges in an input image by approximating the gradient magnitude of the image.

1.1 Robert Gradient

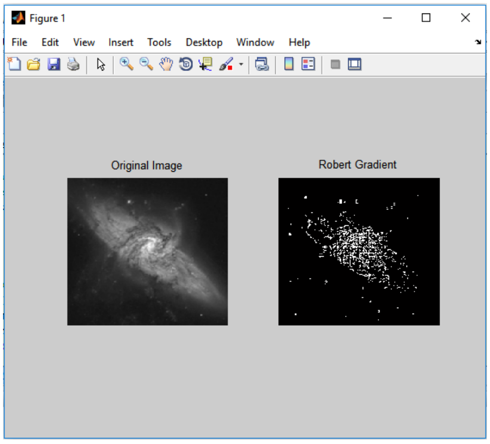

Figure 2: Result of Robert Gradient algorithms

Analyze: As observed in figure 2, the Robert method detect the edges of the object on the image and return those point where the gradient intensity is maximum.

1.2 Sobel Gradient

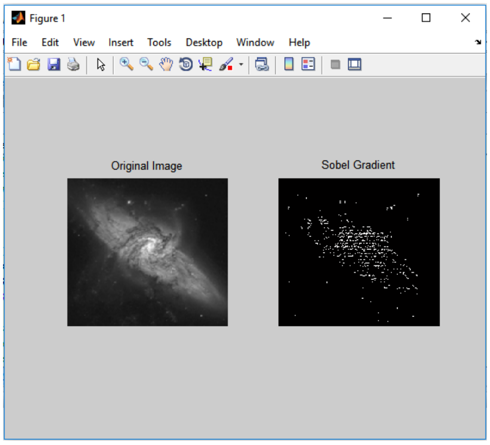

Figure 3: Result of Sobel Gradient algorithms

Analyze: As observed in figure 3, the Sobel method detect the edges of the object on the image also same as the Roberts methods and return those point where the gradient intensity is maximum. We can analyze that the image turn black and only those white part are showing.

1.3 Laplacian Gradient

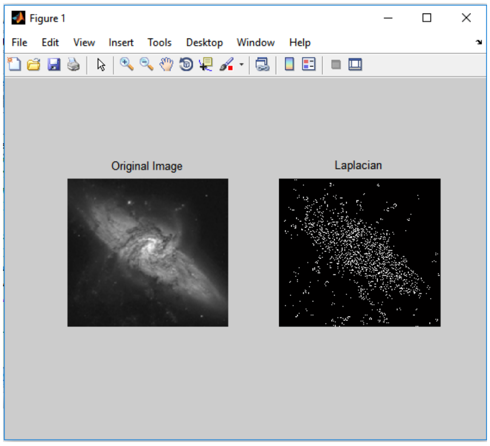

Figure 4: Result of Laplacian Gradient algorithms

Analyze: As observed in figure 4, the Laplacian method detect the edges of the object on the image just as the Roberts and Sobel methods and return those point where the gradient intensity is maximum. We can analyze that the image turn into black and only those white part are showing. We can observe that for the Laplacian method the white noise is more important than for the Roberts and Sobel method.

2 Problem 2

For the first part we are to add a gaussian white noise on the original image of Anthe Galaxy. See the code below.

```
%% Problem 2
% Add a gaussian white noise on the image
Addnoise = imnoise(Image,'gaussian') ;
imshow (Addnoise)
title ('Image with gaussian white noise')
```

prob2.m

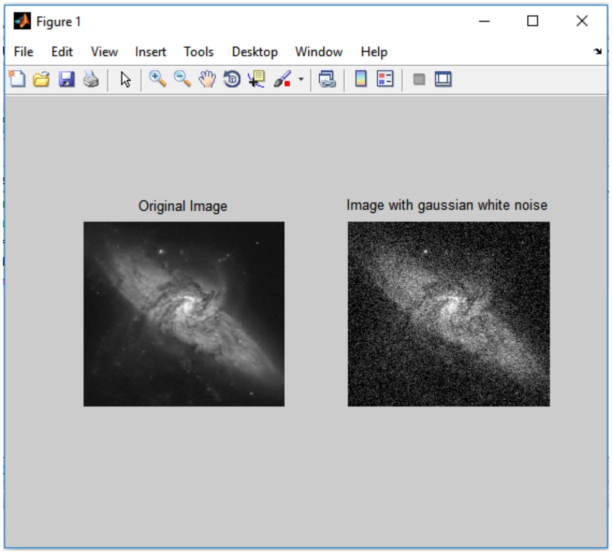

Figure 5: Result of the gaussian white noise

For the second part we are to use a spatial low-pass filter to eleminate the artefacts. Here after the code for the spatial low-pass filter.

```
% % Apply a spatial low-pass filter to eliminate the artefacts
  h = fspecial('gaussian', [7 7], 0.9);
  filtered = imfilter(Image, h);
  imshow(filtered)
  title(' spatial low-pass filter');
```

prob21.m

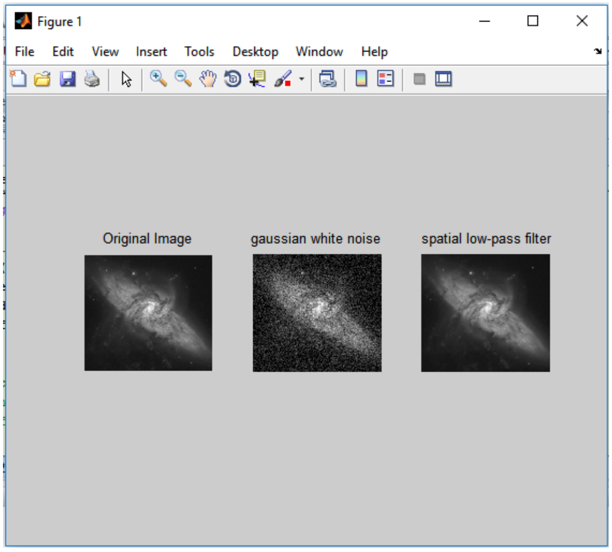

Figure 6: Result of the spatial low-pass filter

For the third part we have to give the measure of the Peak Signal to Noise Ratio (PSNR) of the image with noise and the filtered image. Here after the code.

```
% % Calculate the PSNR of the image with noise
 [peaksnr, snr] = psnr(Addnoise, Image);
 fprintf('\n The PSNR value of the image zith noise is %0.4f', peaksnr);

% % Calculate the PSNR of the filtered image
 [peaksnr, snr] = psnr(filtered, Image);
 fprintf('\n The PSNR value of the filtered image is %0.4f', peaksnr);
```

psnr.m

The value of the PSNR for the noise image is 20.5867 The value of the PSNR for the filtered image is 43.9606

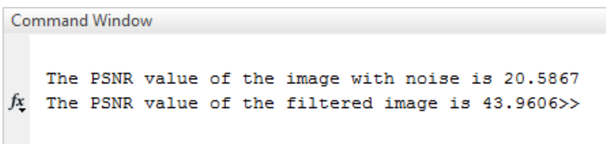

Figure 7: Value of psnr from command window

For the last part we are to add a spatial high-pass filter on the original image to enhanced the transitions.Below the matlab code.

```
% Apply a spatial high-pass filter

% First Kernel
kernel1 = -1 * ones(3)/9;
kernel1(2,2) = 8/9;

% Second Kernel
kernel2 = [-1 -2 -1; 0 0 0; 1 2 1];

% Third Kernel
kernel3 = [-1 0 1; -2 0 2; -1 0 1];

% Appling filters
filteredImage1 = imfilter(single(Image), kernel1);
filteredImage2 = imfilter(single(Image), kernel2);
filteredImage3 = imfilter(single(Image), kernel3);

%SHowing

subplot(2,2,1);
imshow(Image);
title('Original');
subplot(2,2,2);
imshow(filteredImage1, []);
title('Kernel 1');
subplot(2,2,3);
imshow(filteredImage2, []);
title('Kernel 2');
subplot(2,2,4);
imshow(filteredImage3, []);
title('Kernel 3');
```

highpass.m

In the figure below is the result of different type of high-pass filters on the original image.

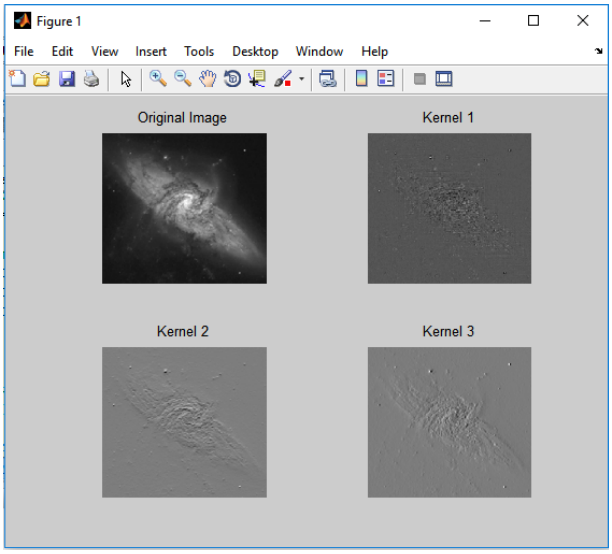

Figure 8: Result of the spatial high-pass filters

3 Problem 3

For this part we are to enhance a MRI scan image.We will be using various filters to try to see the best fit for the scan image. Here after the application of the filters matlab.

```
%% Problem 3

img = imread('./images/mri.png');
res_imadjust = imadjust(img);
res_histeq = histeq(img);
res_adapthisteq = adapthisteq(img);

%Display the plot
figure
subplot(2,2,1);
imshow(img);
title('Original');

subplot(2,2,2);
imshow(res_imadjust);
title('IMADJUST');

subplot(2,2,3);
imshow(res_histeq);
title('HISTEQ');

subplot(2,2,4);
imshow(res_adapthisteq);
title('ADAPTHISTEQ');
```

scan.m

Analyze: In the figure below is the result of the applied filters on the scan image.Each filter had an impact on the original image. Some specific details of the image were seen, some were not. In conclusion we will say the best image is obtain with all the filters depending of what specific detail we want.

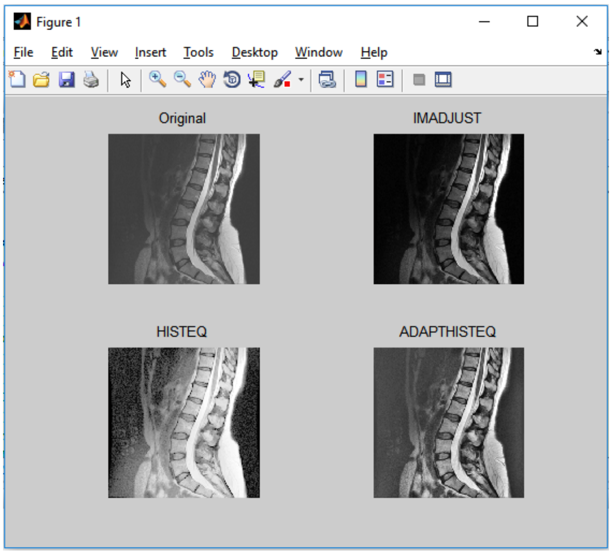

Figure 9: Result of the applied filters on the scan image

4 APPENDIX

4.1 Code of LAB 2

```
%% Selma Boudissa and Meldrick Reimmer
% 21/03/17
% Image Processing
% Lab2 - Image filtering and histogram
clc
close all
clear all


 %% Problem 1

 % Read and display the image
   figure(1)
   subplot( 2,2,1)
   Image = imread ('galaxy.png'); %Load the image
   imshow (Image)
   title('Original Image')

% % Test Robert Gradient
  subplot(2,2,2)
  rob= edge(Image, 'Roberts');
  imshow(rob);
  title('Robert Gradient')

 % Test Sobel
  subplot(2,2,3)
  SobelTest = edge(Image,'Sobel');
   imshow(SobelTest);
   title('Sobel Gradient')

 % Test Laplacian
  subplot(2,2,4)
  Laplacian = edge(Image,'log') % Apply Laplacian filter
  imshow(Laplacian);
  title('Laplacian')

%% Problem 2

 % Read and display the image
   figure(2)
   subplot( 1,3,1)
   Image = imread ('galaxy.png'); %Load the image
   imshow (Image)
   title('Original Image')

% Add a gaussian white noise on the image
  subplot(1,3,2)
  Addnoise = imnoise(Image,'gaussian') ;
  imshow (Addnoise)
  title ('gaussian white noise')

 % Apply a spatial low-pass filter to eliminate the artefacts
   subplot(1,3,3)
   h = fspecial('gaussian', [7 7], 0.9);
   filtered = imfilter(Image, h);
   imshow(filtered)
   title(' spatial low-pass filter');

 % Calculate the PSNR of the image with noise
```

```
  [peaksnr, snr] = psnr(Addnoise, Image);
  fprintf('\n The PSNR value of the image with noise is %0.4f', peaksnr);

% Calculate the PSNR of the filtered image
  [peaksnr, snr] = psnr(filtered, Image);
  fprintf('\n The PSNR value of the filtered image is %0.4f', peaksnr);


% Applying a spatial high-pass filter
% Spatial High pass filter

% First Kernel
 kernel1 = -1 * ones(3)/9;
 kernel1(2,2) = 8/9;

% Second Kernel
kernel2 = [-1 -2 -1; 0 0 0; 1 2 1];

% Third Kernel
kernel3 = [-1 0 1; -2 0 2; -1 0 1];

% Appling filters
filteredImage1 = imfilter(single(Image), kernel1);
filteredImage2 = imfilter(single(Image), kernel2);
filteredImage3 = imfilter(single(Image), kernel3);


%Plotting the Results.

figure(3)
subplot( 2,2,1)
Image = imread ('galaxy.png'); %Load the image
imshow (Image)
title('Original Image')

subplot(2,2,2);
imshow(filteredImage1, []);
title('Kernel 1');

subplot(2,2,3);
imshow(filteredImage2, []);
title('Kernel 2');

subplot(2,2,4);
imshow(filteredImage3, []);
title('Kernel 3');


%% Problem 3
% % Read and display the image
  Img = imread ('./images/mri.png'); %Load the image
  figure (4);
  title ('Original image')
  imshow (Img)


%  Enhance the contrast on the scan image
 H= imadjust (Img)
 figure (5) , title ('Image after contrast enhanced')
 imshowpair (Img, H , 'montage')
```

LAB.m