CPS 843 Assignment 1

Xiaoxin Zhou
Computer Science
Ryerson university
Toronto, CA
Xiaoxin.zhou@ryerson.ca

Abstract: This work is for Assignment 2.

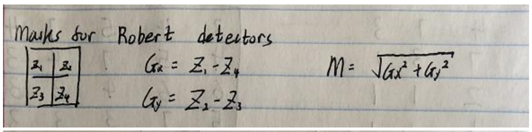
Introduction:

This work we are going to write down the masks for Robert, Prewitt, and Sobel edge detectors and using MATLAB to present results and perform a reasonable analysis. Write down the first and second order derivatives and do the compute. Write down the image sharpening process based on unsharp masking and high-boost filtering, and making a demo on MATLAB. Test the average and gaussian filter.

Part 1

Problem 1:

Write down the masks for Robert, Prewitt, and Sobel edge detectors:



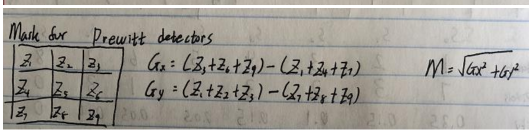
Masks for Robert detectors

z_1	z_2
z_3	z_4

$G_x = z_1 - z_4$

$G_y = z_2 - z_3$

$M = \sqrt{G_x^2 + G_y^2}$



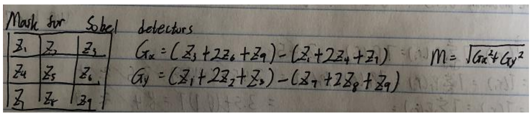
Mask for Prewitt detectors

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$G_x = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$

$G_y = (z_1 + z_2 + z_3) - (z_7 + z_8 + z_9)$

$M = \sqrt{G_x^2 + G_y^2}$



Mask for Sobel detectors

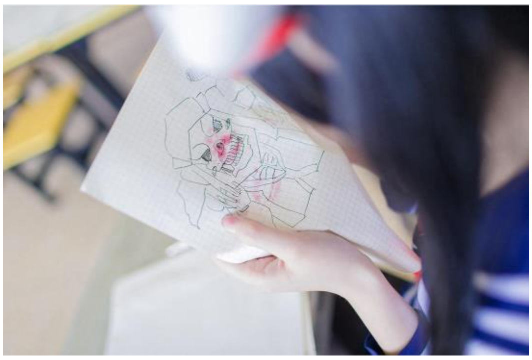
z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$G_x = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$

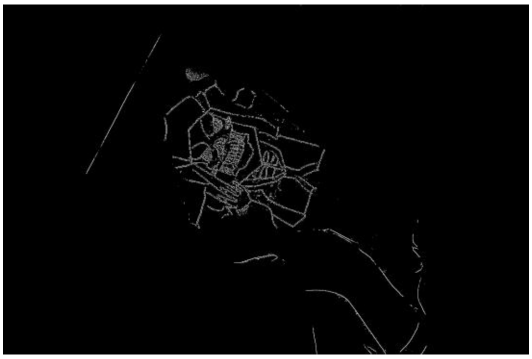
$G_y = (z_1 + 2z_2 + z_3) - (z_7 + 2z_8 + z_9)$

$M = \sqrt{G_x^2 + G_y^2}$

Original Image:



Roberts Image:



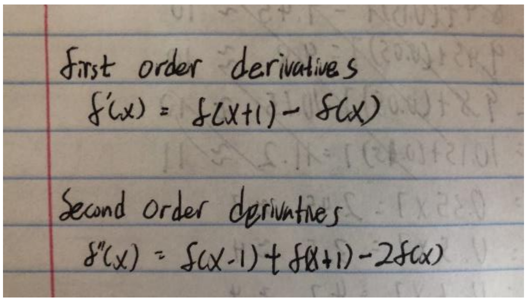
Prewitt Image:

Sobel Image:

reasonable analysis:

The Roberts operator consist of a pair of two-dimensional (2x2) convolution mask, Prewitt and Sobel are consist of a pair three-dimensional (3x3) convolution mask. From the experiment, Roberts missing detail of the image and gives more noise. Prewitt and Sobel show the same performance. However, Sobel provides less noise than Prewitt.

Problem2:

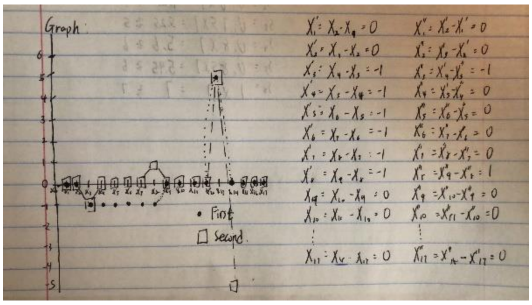


first order derivatives

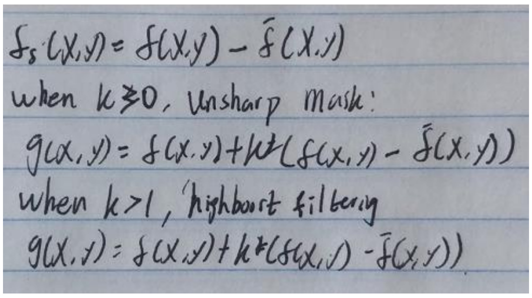
$f'(x) = f(x+1) - f(x)$

Second order derivatives

$f''(x) = f(x-1) + f(x+1) - 2f(x)$



Problem3:



$f_s(x,y) = f(x,y) - \bar{f}(x,y)$

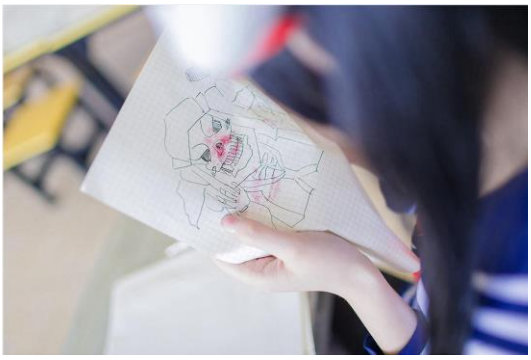
when $k \geq 0$, Unsharp mask:

$g(x,y) = f(x,y) + k*(f(x,y) - \bar{f}(x,y))$

when $k > 1$, highboost filtering

$g(x,y) = f(x,y) + k*(f(x,y) - \bar{f}(x,y))$

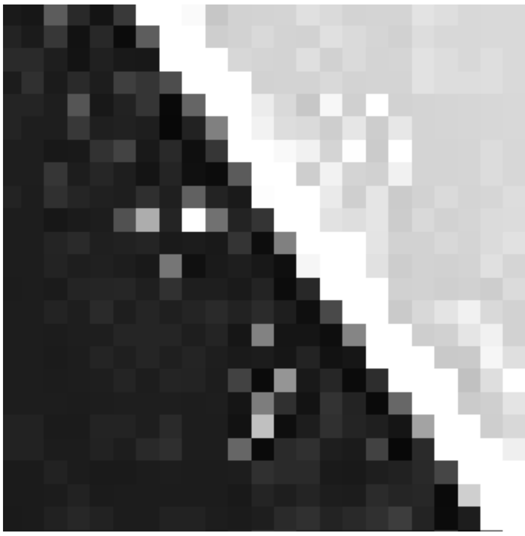
Original Image:



Unsharp masking: K = 1

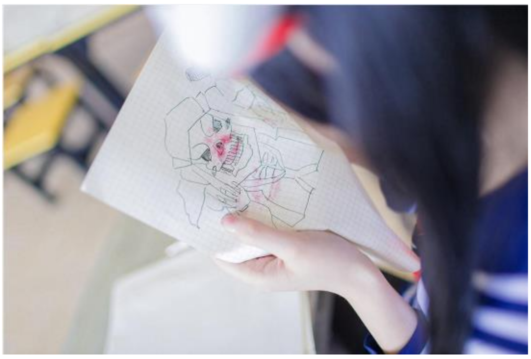
Highboost Image: K = 5

Increasing the K value can provide a stronger edge contrast; however, this process will damage the image and generated noise.

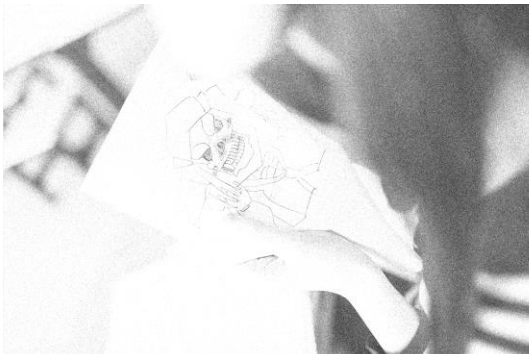


Problem 4:

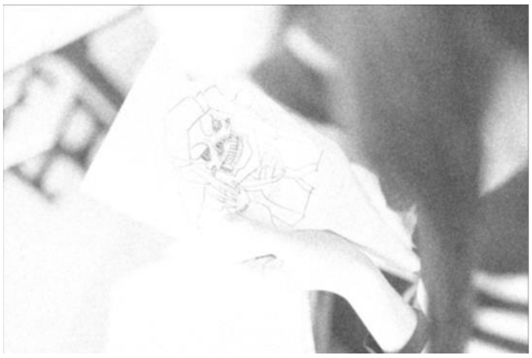
Original Image:



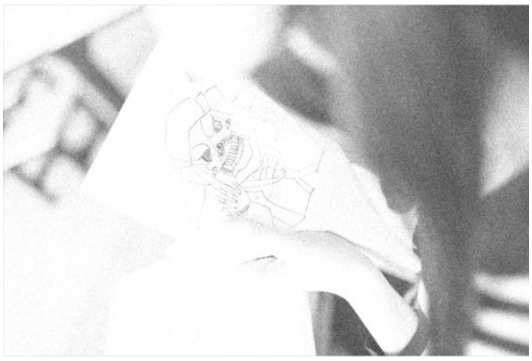
After gaussian noise with 0.3 variance.



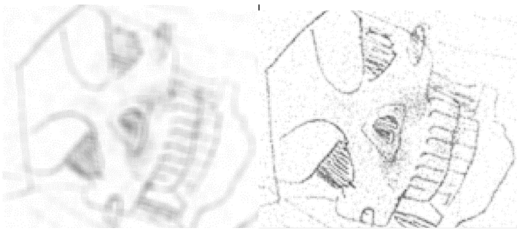
Average Filter with 5



Gaussian Filter with 5



My Gaussian noise level is 0.3, and all the filter level is 5. From the experiment



The left image is average, and the right image is the gaussian filter. When the same level sets both filters, gaussian give a higher contrast and intensity of the image. When we change the noise level, the result of the conclusion is the same, and the gaussian gives a better result.

Problem5:

1)

first order derivative

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y)$$

$$\frac{\partial f}{\partial y} = f(x, y+1) - f(x, y)$$

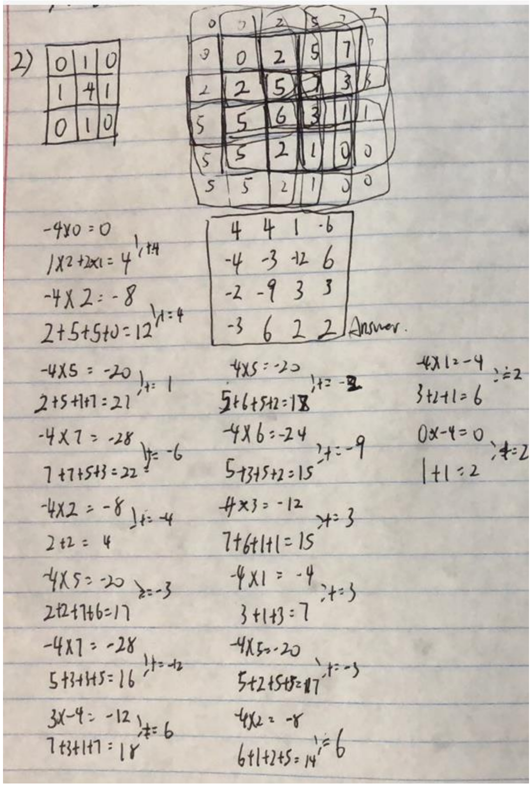
$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

2)

$$\begin{matrix} 0 & 2 & 5 & 7 \\ 2 & 5 & 7 & 3 \\ 5 & 6 & 3 & 1 \\ 5 & 2 & 1 & 0 \end{matrix}$$

$$Gx = \cancel{0} \quad \begin{matrix} 3 & 2 \\ 2 & -4 \\ -3 & -2 \\ -1 & -1 \end{matrix} \qquad Gy = \begin{matrix} 3 & 1 & -4 & -2 \\ 0 & -4 & -2 & -1 \end{matrix}$$

3)



4)

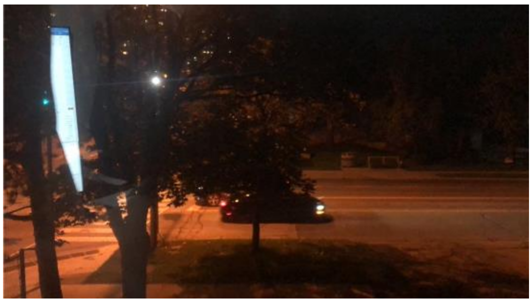
Part2:

Improve the visibility of an image can increase the computer vision algorithm's robustness in low-light conditions. The "Low-Light Image Enhancement" provided a haze removal technique to enhance low-light images. The most important part is to invert the image for the low-light piece. This means the low light part becomes a high light image, but it is hard to see, and the original high light part is missing; the inverted image is also hazy. So, we need to reduce the haze to improve the image intensity and contrast. In the last, we can reverse the picture again, which is back to the original, this action can give the image original high light part to previous, and the low light part is improved by reducing the hazy steps.

Coding:

```
A = imread('2.jpg');
imshow(A);
```

To read the original image:



```
AInv = imcomplement(A);
imshow(AInv);
```

Invert the low-light image

```
BInv = imreducehaze(AInv);
imshow(BInv)
```

Remove the hazy from the inverted low light part.

```
B = imcomplement(BInv);
```

Invert back the image.

We can see the trees chair on the other side and more after using the low-light image enhancement algorithm. From this experiment, I understand if we need to increase the feature extraction, we can use this algorithm to improve our robustness.

Resource:

[1] MATLAB. Low-Light Image Enhancement. 2021, from https://www.mathworks.com/help/images/low-light-image-enhancement.html