ECSE 4540 Image Processing

HW 3

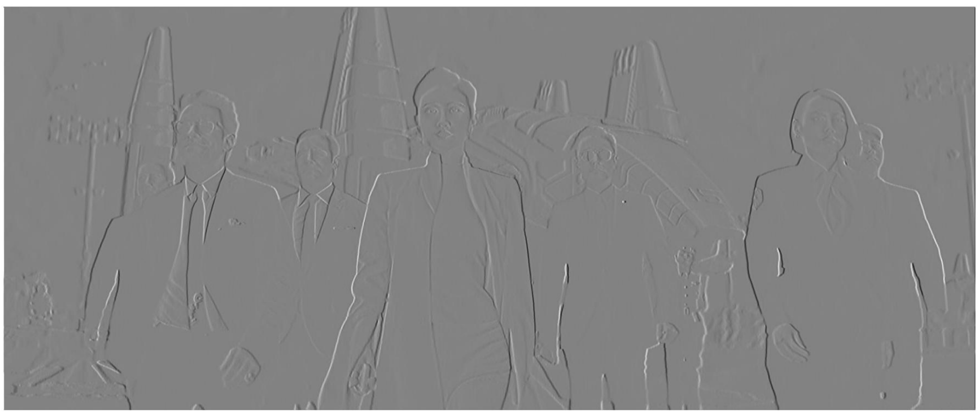
Dec 11, 2017
Daniel Southwick
661542908

```
%% ===================================3.1===================================
```

Main Function

```
%% 1a
walk = imread('walk.png');
walk = im2double(walk);
walk1 = filter2([-1 0 1; -2 0 2; -1 0 1], walk);
imshow(walk1,[]);
```

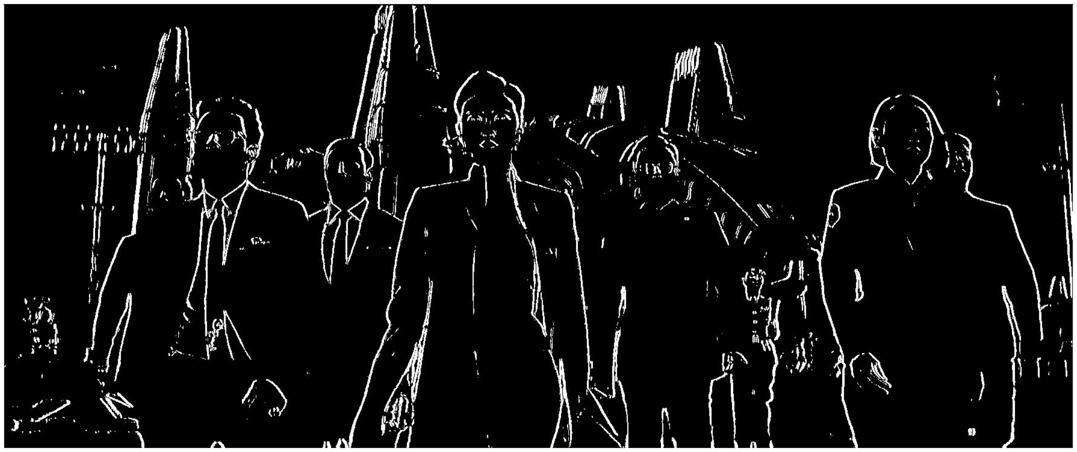
Matlab Result



Main Function

```
%% 1b
walk2 = abs(255*walk1) > 200;
imshow(walk2,[]);
walk2 = abs(255*walk1) > 75;
imshow(walk2,[]);
```

Matlab Result

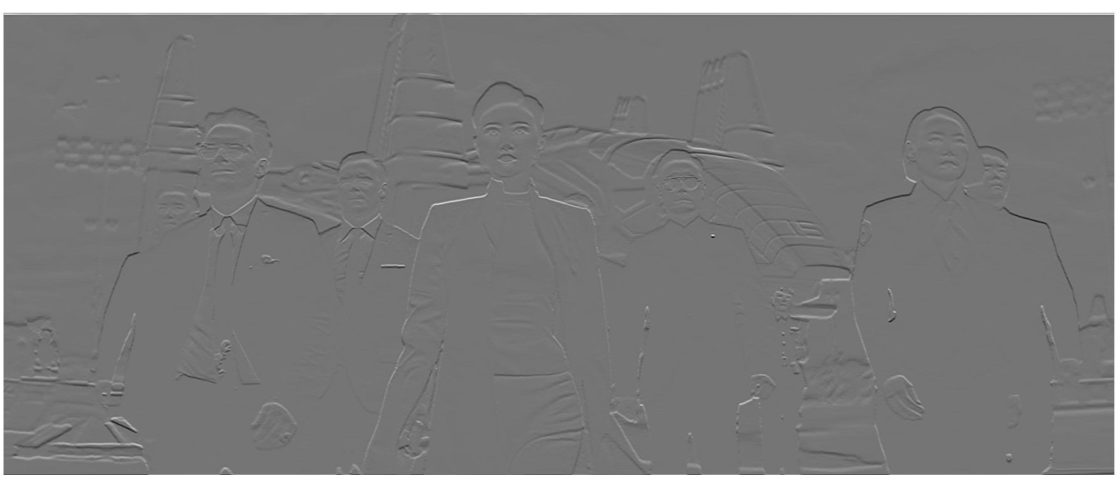


From the first Sobel filter we see this:This results in a fully grayscale image that mostly highlights the vertical edges of the objects present. Once the image is passed through the absolute thresholding of 200, it is clear that only the vertical lines of the image remain. Thus, as indicated by the 3 vertical 0's in the matrix, this Sobel operator attempts to edge detect for vertical lines specifically. In the third image we begin to see the vertical lines of the plane, this occurs when the thresholding is lowered to about 75. However, as a result of lowering the threshold many more lines that are non-vertical also begin to appear, such as the diagonal stripes on the women's shirt.

Main Function

```
%% 1c
walk3 = filter2([-1 -2 1; 0 0 0; 1 2 1], walk);
imshow(walk3,[]);
walk4 = abs(255*walk3) > 200;
imshow(walk4,[]);
```

Matlab Result



Above we see the result of the second Sobel operator. The difference between this grayscale image and the first one is that this one does not enunciate the vertical edges but rather detects the edges of the horizontal lines. Once this is thresholded to 200, we receive the second picture. Very clearly these tries to detect the horizontal edges of the original picture.

```
%% ===================================3.2====================================
```

Main Function

```
%% 2a
cat = imread('fuzzycat.png');
cat1 = imfilter(cat, ones(5)/25);
imshow(cat1,[]);
```

Matlab Result

We first filter the image with the low-pass filter ones(5)/25 to arrive at the image below. This image is clearly a more blurry version of the original picture. However, the purpose of this blurry image to enunciate the edges for the next part of the problem by subtracting the blurry image from the original image.

Main Function

```
%% 2b
cat2 = cat + (cat - cat1);
cat2(cat2>255)=255;
imshow(cat2,[]);
cat3 = cat + 2*(cat - cat1);
cat3(cat3>255)=255;
imshow(cat3,[]);
cat4 = cat + 3*(cat - cat1);
cat4(cat4>255)=255;
imshow(cat4,[]);
cat5 = cat + 4*(cat - cat1);
cat5(cat5>255)=255;
imshow(cat5,[]);
cat6 = cat + 5*(cat - cat1);
cat6(cat6>255)=255;
imshow(cat6,[]);
```

Matlab Result

k=1

k=2

k=3

```
k=4
```

```
k=5
```

Above are the images with the unsharp masking of different k-values being applied.. The higher the value of k that is applied the sharper this image become and the more the edges are enunciated. Below are the 5 different k values applied. Overall these pictures are much better than the low-pass filter applied in the first part, and are increasingly better the higher the k-value used.

```
%% ====================================3.3====================================
```

Main Function

```
%% 3a
boat = imread('boat.png');
boat1 = imfilter(boat, ones(5)/25);
imshow(boat1,[]);
```

Matlab Result

Main Function

```
%% 3b
boat2 = medfilt2(boat1,[5 5])
imshow(boat2,[]);
```

Matlab Result

3a is the image with the low-pass filter ones(5)/25 applied onto it. Overall this picture is much blurrier than the original image, but it removes the grain effect that was present in the original image. 3b is the image with the 5x5 median filter applied onto it. While this image is also still very blurry when compared to the original image, it is better at preserving the sharp boundaries across the image. So, while some of the grain effect is still present, the edges around the men's clothes, facial features, and body are more defined. Perceptually this image seems better.

3.4

a)
$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} a & b & c & d & e \\ f & g & h & i & j \\ k & l & m & n & o \\ p & q & r & s & t \\ u & v & w & x & y \end{bmatrix}$$

$a = e = u = y = 0$

$b = c = d = f = j = k = o = p = t =$

$v = w = x = \frac{1}{9}$

$g = -4 \times \frac{1}{9} + \frac{1}{9} + \frac{1}{9} = \frac{-2}{9}$

$h = \frac{1}{9} + \frac{1}{9} + \frac{1}{9} - \frac{4}{9} = -\frac{1}{9}$

$i = \frac{1}{9} + \frac{1}{9} + \frac{-4}{9} = \frac{-2}{9}$

$l = \frac{1}{9} + \frac{1}{9} + \frac{1}{9} - \frac{4}{9} = \frac{-1}{9}$

$m = 0$

$n = l = -\frac{1}{9}$

$q = \frac{1}{9} + \frac{1}{9} - \frac{4}{9} = \frac{-2}{9}$

$r = \frac{1}{9} + \frac{1}{9} + \frac{1}{9} - \frac{4}{9} = \frac{-1}{9}$

$s = \frac{1}{9} + \frac{1}{9} - \frac{4}{9} = \frac{-2}{9}$

$$\begin{bmatrix} 0 & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & 0 \\ \frac{1}{9} & \frac{-2}{9} & \frac{-1}{9} & \frac{-2}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{-1}{9} & 0 & \frac{-1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{-2}{9} & \frac{-1}{9} & \frac{-2}{9} & \frac{1}{9} \\ 0 & \frac{1}{9} & \frac{1}{9} & \frac{1}{9} & 0 \end{bmatrix}$$

b) Result will be Same

Since convolution does commute

Scanned by CamScanner

```
%% ===================================3.5=====================================
```

Main Function

```
%% 5a
board = imread('board.png')
h=fspecial('gaussian',[19 19],3)
board2 = imfilter(board,h);
imshow(board2);
```

Matlab Result

The above image is passed through a 2D Gaussian low pass filter with standard deviation of 3, the image appears to be slightly blurred and almost out of focus. 2D Gaussian low pass applied the filter both to the columns and the rows.

GAUSSV

```
%% 5b
function H=gaussv(N,sigma)
h=(N+1)/2;
for i=1:N
    H(i)=exp(-(i-h)*(i-h)/(2*sigma.^2))/(sigma*sqrt(2*pi))
end
H=(1/sum(H))*H
H=H'
end
```

Main Function

```
%% 5c
board3 = imfilter(board,gaussv(19,3));
imshow(board3,[]);
```

Matlab Result

When 1-D Gaussian low pass filter is applied, only to the columns, the picture became much clearer compare to part a). This is caused by the way that original picture was taken, only out of focus on the vertical; this is why when the image passed through 2-D Gaussian is blurrier than the 1-D column Gaussian with the same length and standard deviation.

Main Function

```
%% 5d
board4 = imfilter(board,gaussv(31,5));
imshow(board4,[]);
```

Matlab Result

Comparing to part c), the image became blurry again, though both part c) and d) was made by passing through 1-D column low pass Gaussian, the standard deviation in part d) was raised from 3 to 5, thus the image became more filtered and at this point over filtered, so the picture went back to blurry.

Main Function

```
%% 5e
board5 = imfilter(board,gaussv(75,5));
imshow(board5,[]);
```

Matlab Result

The picture from part e) does look almost exactly the same as part d), the only thing changed in this part was the length of the filter, however, the length does not affect the picture much due to the Gaussian filter formula, since the standard deviation was kept at 5, the picture looked almost the same as part d).