Detection of Critical Lunar Crescent by Image Processing Techniques

Mohsen Lesani
Computer Science Department
Shahid Bahonar University of Kerman

Seyyed Jalil Fatemi
Physics Department
Shahid Bahonar University of Kerman

Abstract

Vision of new moon is important especially for Islamic calendar. Human eye even employing a telescope has limited ability to recognize a critical moon crescent. Images can be taken by CCD cameras from the sky where a critical moon crescent should be seen. The crescent is not sometimes evident in these images also. This paper presents applying image processing techniques to these images to automatically highlight the crescent. Highlighting the crescent helps to recognize the crescent when it can not be seen by advanced observational instruments.

1 Spatial Image Filtering

Image filtering is done generally for enhancing image quality. Two main approaches in image filtering are spatial filtering and frequency domain filtering. The following lines describe the main issues in spatial image filtering and especially sharpening spatial filters.

Obviously a gray scale image can be viewed as a function *image(x,y)* with variables *x* and *y* ranging on the width and height of the image. Allocating a byte for each pixel storage makes *image(x,y)* to range from 0 to 255.

Spatial filtering is based on concept of filter also called mask or window. The mask is fundamentally a table of numeric values that are premeditated for a definite filtering. The mask is applied to every pixel of the original image yielding a result value for the corresponding pixel in the result image. To be more precise the mask dimensions can be m=2a+1 and n=2b+1 where *a* and *b* are positive integers. This is to only allow masks of odd sizes. Oddness of mask size provides mask symmetry. The mask can therefore be viewed as a two variable function *weight(s ,t)* with variables *s* and *t* ranging from *–a* to *a* and *–b* to *b*. Applying the filter to a pixel at *(x, y)* location on the original image yields:

$$filterResultedImage(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} maskWeight(s, t)\, image(x+s, y+t)$$

The effect is that the corresponding result value for an original image pixel is determined not only by the pixel value itself but also based on the neighbor pixel values. The extent of participation of a neighbor pixel is defined by its corresponding mask cell value which is also called coefficient or weight of the mask cell.

The process of filtering can simply be visualized as a mask sliding through the original image pixel by pixel from left to right and from top to bottom. The locations where all the mask cells are not on the original image, can be ignored resulting in a result image slightly smaller than the original one. If the resulting image is desired to be the same size as of the original one, just the cells of the mask that are on the image can be applied in locations where some cells of the mask are located outside the image. Other solutions include padding the original image with zero valued pixels or with duplicate columns and rows.

2 Sharpening Spatial Filters

The sharpening filters are aimed to highlight details of the image. Some edges in the image may be invisible as a result of limitations of image acquisition conditions such as too much or too little light. Discontinuities of adjacent pixel gray level values of an image depict borders of the image objects. The sharpening filters generally highlight discontinuity in the filtered image. From a mathematical point of view, two popular differentiation operators are first and second-order derivatives.

The first order derivative is defined as:

$$\frac{df}{dx} = f(x+1) - f(x)$$

and the second derivative is therefore defined as:

$$\frac{d^2 f}{d^2 x} = f(x+1) + f(x-1) - 2f(x)$$

To be able to benefit from the mentioned mathematical operations in spatial filtering, they should be translated to masks.

2.1 First-order Derivative Filters

The first-order derivative can not only be defined on the x axis, but also on y, +45 and -45 diagonal axises. So the offered filters for the first order derivate would be:

Roberts Filters:

-1
1

Horizontal

-1	0
0	1

+45^o

-1	1

Vertical

0	1
-1	0

-45^o

Prewitt Filters:

-1	-1	-1
0	0	0
1	1	1

Horizontal

-1	-1	0
-1	0	1
0	1	1

+45^o

-1	0	1
-1	0	1
-1	0	1

Vertical

0	1	1
-1	0	1
-1	-1	0

-45^o

Sobel Filters:

-1	-2	-1
0	0	0
1	2	1

Horizontal

-2	-1	0
-1	0	1
0	1	2

+45^o

-1	0	1
-2	0	2
-1	0	1

Vertical

0	1	2
-1	0	1
-2	-1	0

-45^o

2.2 Second-order Derivative Filters

The second-order derivative filters are most employed for line detection are:

-1	-1	-1	-1	-1	2	-1	2	-1	2	-1	-1
2	2	2	-1	2	-1	-1	2	-1	-1	2	-1
-1	-1	-1	2	-1	-1	-1	2	-1	-1	-1	2
Horizontal			$+45^{o}$			Vertical			-45^{o}		

The Laplacian filters that apply the second-order derivate on both x and y axises and also diagonal axises simultaneously are:

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

2.3 Constructing the Sharpened Image

The mentioned filters when applied yield images that highlight the discontinuities called edge images. The edge image is added to the original one to make it sharper. Thus the final sharpened image is:

$$sharpenedImage(x, y) = image(x, y) + c\ filterResultedImage(x, y)$$

where c is a positive real number to boost the effect of the discontinuities image in the final image. The sharped image obtained this way may need scaling in order to get all its pixel values in the [0, 255] gray scale range.

3 Results

Figure 1 shows the the original image taken by a Nikon Digital Camera COOLPIX-S10 taken in Kerman, Iran on 21st of December 2006. Figure 2 depicts the moon crescent that is taken from Monzor moon location calculator software for the same location and time; so it is expected to have been seen in the sky that the original image is taken from. Figures 3 and 5 are the sharpened images obtained by applying +45 diagonal and horizontal Sobel filters to the original image respectively. The edge boosting constant c is taken 5 for both of the sharpened images. The sharpened images are superimposed on the crescent image that is expected to have been seen as shown is Figures 4 and 6.

Edge Images obtained by applying vertical and +45 diagonal Sobel masks are shown in Figures 7 and 8. To highlight the edges more clearly, the edge images are first scaled to the [0, 255] range and then down thresholded by gray level value threshold of 25 for noise reduction and finally scaled to the [0, 255] range again. They are also inverted to show the edges in dark. The two edge images are superimposed to obtain Figure 9 that shows the edges that are detected in either of the two edge images. Figure 10 shows the border of the target crescent image. The border of the target crescent is superimposed on the edge image of Figure 10 to obtain Figure 11.

All the resulting images in this section are the outcome of a standalone image processing application developed in Java.

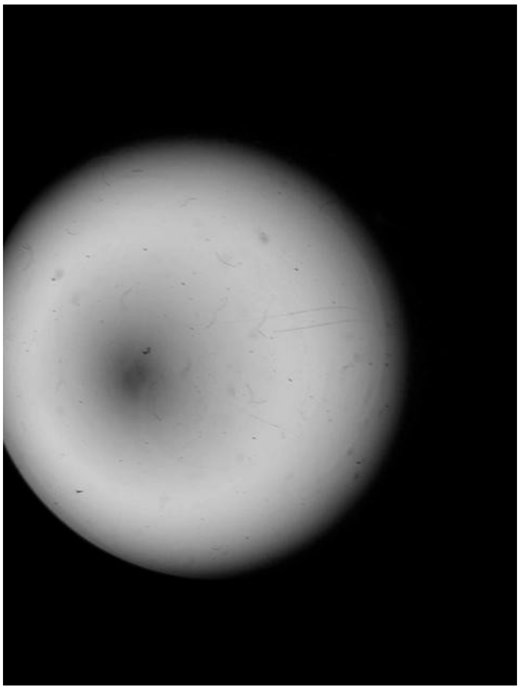

Figure 1: The Original Image

Figure 2: The Target Image

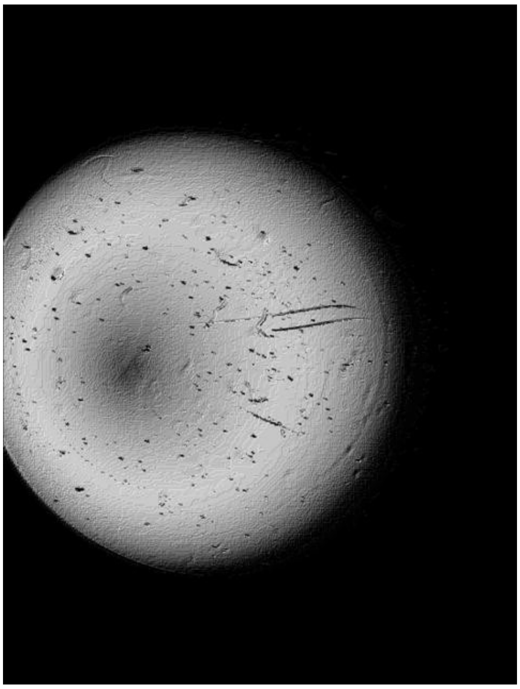

Figure 3: Sharpened Image with +45 Diagonal Sobel

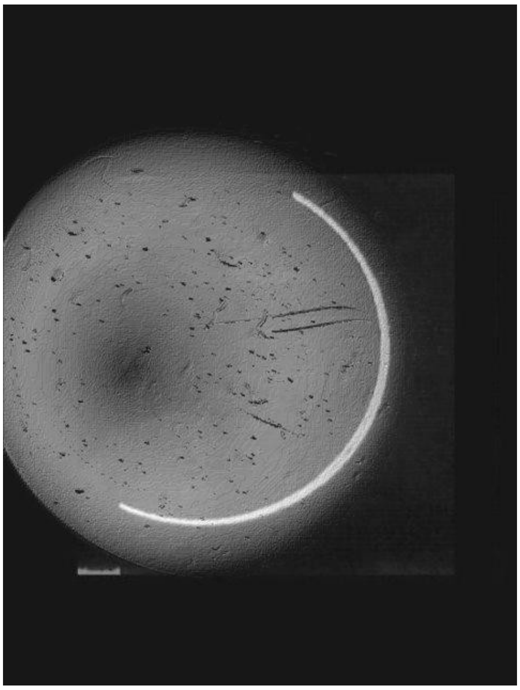

Figure 4: Figure 2 Superimposed on Figure 3

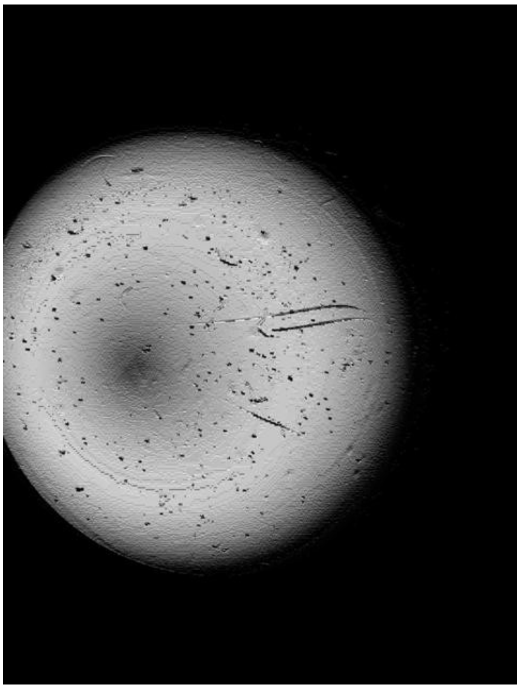

Figure 5: Sharpened Image with Horizontal Sobel

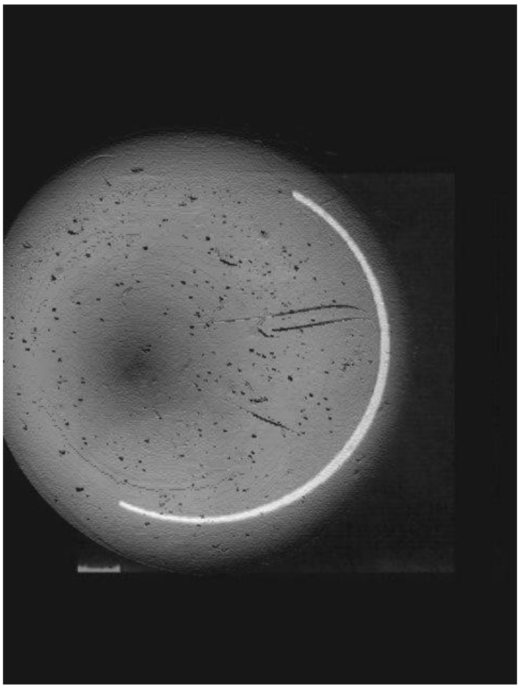

Figure 6: Figure 2 Superimposed on Figure 5

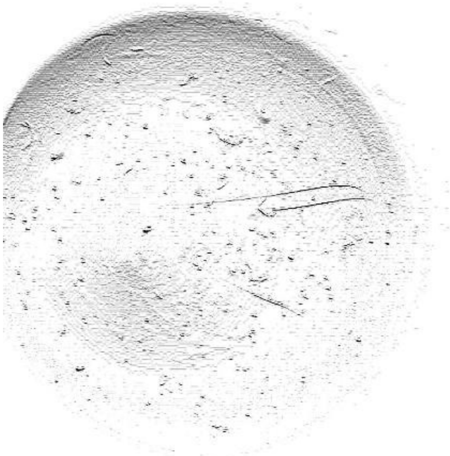

Figure 7: Edge Image Obtained by Vertical Sobel

Figure 8: Edge Image Obtained by +45 Diagonal Sobel

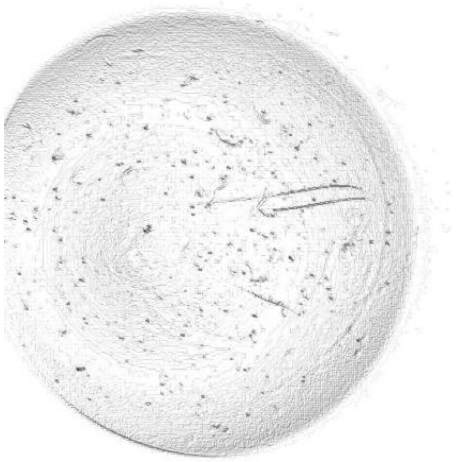

Figure 9: Superimposed of Figures 8 and 9

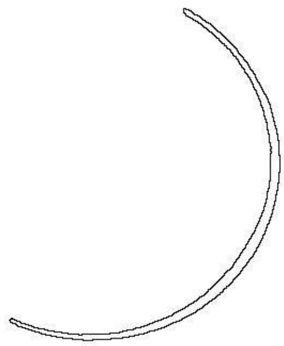

Figure 10: Border of Figure 2

Figure 11: Superimposed of Figures 9 and 10

4 Conclusion and Future Works

The crescent can hardly be recognized on the original image of Figure 2 but is highlighted in the sharpened images. The superimposed images in Figures 4, 6 and 11 show that the techniques employed were successful in highlighting the correct crescent. The shape and direction of the detected crescent is the same as of the crescent that should have been seen; therefore the two crescents could be completely superimposed. This paper presented an inter disciplinary research where knowledge from both physics and image processing from artificial intelligence came together.

Although the results seem satisfactory, some future works are also being pursued. Experimenting on more critical crescent photos and also applying more filters are two future works. The techniques presented help in recognizing the crescent semi automatically with a post review of a human. It is more desirable if the process of detection could be completely automated that is to propose mechanisms that autonomously highlight and recognize the crescent if it is present in the image.

5 References

1. Gonzalez, Woods. Digital Image Processing, 2nd Ed., Prentice Hall 2002.
2. Jane, Kasturi, et all. Machine Vison. 1995.
3. Monzor moon location calculator software
4. Fatemi, et all. Visibility of very critical lunar crescent by using CCD, Emirates Astronomical Conference, 13 December 2006.