

Modified Bit-Planes Sobel Operator: A New Approach to Edge Detection

Rashi Agarwal, Ph.D
Reader, IT Department
CSJMU Kanpur-208024

ABSTRACT

The detection of edges in images is a vital operation with applications in various fields. There are a number of methods developed already for the same. We have developed a 'global method' for extraction of edges which is a modification of the existing Sobel operator. We have first extracted the bit planes of each image and have applied the Sobel operator on each bit plane for enhanced results. After this we have recreated the image by adding up the edges of all the bit-planes in their order of importance. This is a fairly simple global method which yields very good results. The computations are simpler and faster as well. Pratt's figure of merit (FOM) has been used to quantify the measure of edges. The values of Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) have been calculated to assess the performance of the new algorithm in comparison to the previous existent one in presence of additive Gaussian noise. The results favor our new algorithm clearly.

General Terms

Digital Image Processing.

Keywords

Edge Detection, Filtering, Segmentation, Sobel Operator.

1. INTRODUCTION

Edge detection is a vital operation in image processing with numerous applications in scene analysis and object recognition. Various algorithms with different complexities and tradeoffs exist for detection of edges. Numerous studies have been done in the past to assess these algorithms. [1]. Detection of edges refers to the process of identifying and locating sharp discontinuities in an image. The discontinuities are abrupt changes in pixel intensity which characterize boundaries of objects in a scene. Classical methods of edge detection involve convolving the image with an operator (a 2-D filter), which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. There are two main approaches to edge detection:

Gradient: The gradient method detects the edges by looking for the maximum and minimum in the first derivative of the image.

Laplacian: The Laplacian method searches for zero crossings in the second derivative of the image to find edges. An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location.

In the former method, the discontinuities are enhanced by neighborhood operators. The Robert's, Prewitt and Sobel operators are examples of this method. A quantitative assessment of these is done in [2]. All the gradient-based algorithms have kernel operators that calculate the strength of the slope in directions which are orthogonal to each other, commonly vertical and horizontal. Later, the contributions of the different components of the slopes are combined to give the total value of the edge strength.

Most edge detection methods work on the assumption that the edge occurs where there is a discontinuity in the intensity function or a very steep intensity gradient in the image. Using this assumption, if one takes the derivative of the intensity value across the image and finds the points where the edges could be located. The gradient is a vector, whose components measure how rapid pixel value are changing with distance in the x and y direction.

Edge detection in noisy environment can be treated as an optimal linear filter design problem. Canny formulated edge detection as an optimization problem and defined an optimal filter, which can be efficiently approximated by the first derivative of Gaussian function in the one-dimensional case. Canny's filter was further extended to recursive filters, which provide a more efficient way for image noise filtering and edge detection. [3].

Other edge detection methods include differentiation based edge detection using logarithmic image processing (LIP) models [4], contrast-based methods [5], relaxation labeling techniques [6] and anisotropic diffusion [7]. In fact, these methods can be combined to achieve better performance.

2. METHODOLOGY

In our study we have used a modified version of the simple Sobel filter to find out edges. We have applied the filter to various bit planes of an image as described further and combined the obtained masks in order of their significance to get the final edge image effectively. Appropriate threshold is applied to determine the best edge extracted according to the Pratt Figure of Merit which is discussed later.

2.1 Bit Plane Slicing

Given an image (X-bit per pixel), slicing the image at different planes (bit-planes) plays an important role in image processing. We can extract X number (0 to X-1) of different bit-planes for the image where each bit-plane represents the corresponding bits of each pixel. In general, 8-bit per pixel images are processed. We can slice an image into 8 bit-planes. Zero is the least significant bit (LSB) and 7 is the most significant bit (MSB) [8]:

Shown below is an 8-bit per pixel image and how each plane represents information on that plane.

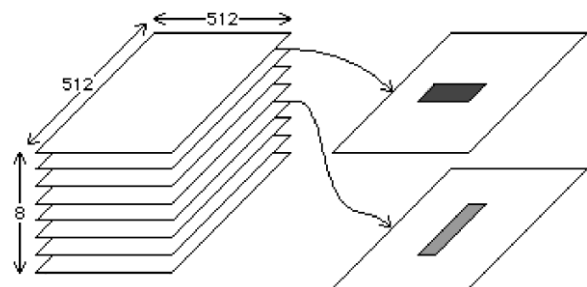


Fig. 1. Bit planes of an 8 bit image (512 X 512 size)

These bit planes can be combined in their order, back again to get the original image. The LSB bit-plane contains least important information related to the image and MSB bit-plane contains the most important information related to the image.

2.2 Sobel operator

The Sobel operator performs a 2-D spatial gradient measurement on an image. Typically it is used to find the approximate absolute gradient magnitude at each point of an input grayscale image. The Sobel edge detector uses a pair of 3 x 3 convolution masks, one estimating gradient in the x-direction and the other estimating gradient in y-direction. In the Sobel mask operator consists of a pair of 3x3 convolution kernels as shown in Figure 1. One kernel is simply the other rotated by 90°. [9,10] .

-1	0	+1	+1	+2	+1
-2	0	+2	0	0	0
-1	0	+1	-1	-2	-1
G _x			G _y		

Fig. 2. Kernels G_x and G_y

2.3 Modified Biplane Sobel Operator (MBSO)

The process of extraction of edges was carried out on each bit-plane of the image and these edges were then recombined to give the final result. We have named our method the Modified Biplane Sobel Operator (MBSO). Below are shown images of “Lena” in figure 3, which are most popular in the image processing fraternity. The output from the Sobel operator is shown in figure 4 whereas the output from MBSO is shown in figure 5.



Fig. 3. Original “Lena” image



Fig. 4. Sobel operator applied on “Lena”



Fig. 5. MBSO applied on “Lena”

The various bit-planes extracted and their corresponding edges are shown as follows.



Fig. 6. Bit-plane 7 of original image

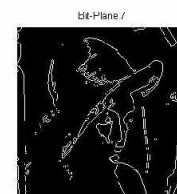


Fig. 7. Bit-plane 7 edge



Fig. 8. Bit-plane 6 of original image



Fig. 9. Bit-plane 6 edge



Fig. 10. Bit-plane 5 of original image



Fig. 11. Bit-plane 5 edge

The “cameraman” image gave the following results:



Fig. 12. The “Cameraman” image



Fig. 13. Edge extracted from Sobel operator



Fig. 14. Edge from MBSO

The “bam” image gives following results:



Fig. 15. The “Barn” image

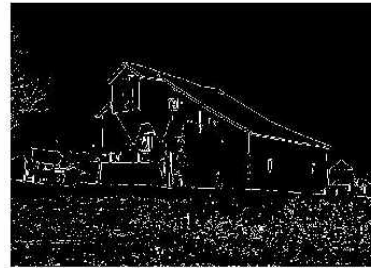


Fig. 16. “Sobel” operator applied



Fig. 17. “MBSO” applied

The ‘circuit’ image gave results as shown:

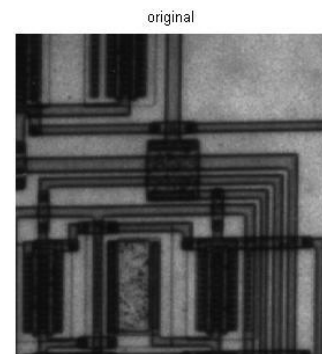


Fig. 18. The original “Circuit” image

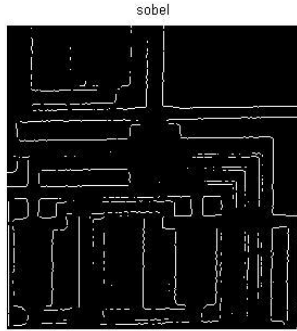


Fig. 19. “Sobel” operator applied

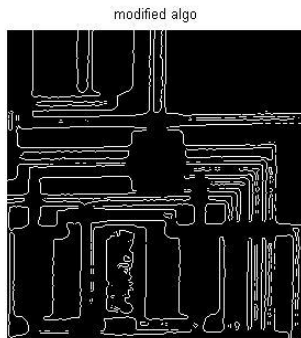


Fig. 20. “MBSO” applied

It is visually clear that a lot of edges which were lost when the filter was applied on the whole image are now visible when we process the images on their bit plane level. The bit-plane 0-5 may be omitted as they tend to introduce noise where as bit-plane 6 and 7 give a much better output.

3. RESULTS AND DISCUSSION

Several quantifiable comparison methodologies exist for edge detection [11,12,13]. Heath et al. propose an edge detection comparison technique based on visual assessments by a group of people trained to assess edge quality [11,12]. This paradigm is based on experimental psychology and statistics and assesses whether there is a statistically significant difference in edge detector outputs as perceived by humans when considering an object recognition task. Although this technique can be used for evaluating color edge detectors, it was not used due to its complexity and inability to easily factor out extraneous edges. We have instead used the well-known Pratt’s Figure of Merit [14] to compare edge detector output. This measure is well understood and it is possible to control largely what is being tested.

Pratt’s Figure of Merit (FOM) attempts to balance three types of errors that can produce erroneous edge maps: missing valid edge points, failure to localize edge points and classification of noise fluctuations as edge points. The Figure of Merit is defined as [44]

$$R = \frac{1}{I_N} \sum_{i=1}^{I_A} \frac{1}{1 + ad^2}$$

In this equation, I_N is the maximum of I_A and I_I . I_A represents the total number of actual edge pixels; i.e., those edge pixels

that were found. I_I represents the total number of ideal pixels in the image; i.e., the number of edge pixels in the reference image (e.g. Figure 27 see for an ideal edge map of Image 26). The parameter a is a scaling constant while d is the distance from an actual edge point to the nearest ideal edge point (in this thesis $a = 0.9$). The scaling factor is used to penalize edges that are localized, but offset from the true position (in this thesis a scaling value of 0.9 is used). The rating factor is normalized so that a value of one means that the edge has been detected perfectly. The Figure of Merit is normalized with the maximum of the actual and ideal number of edge pixels in order to ensure a penalty for smeared (i.e., $I_I < I_A$) or fragmented edges (i.e., $I_I > I_A$).

In this work, Pratt’s Figure of Merit was used to determine an ideal edge threshold. This was done by choosing the threshold value corresponding to the highest Figure of Merit score for each algorithm. In this way, all results are compared to each other when they are closest (from a Figure of Merit point of view) to the ideal edge map. This ensures that the results being compared are relatively the best possible for that edge detection algorithm.

The typical training run to obtain an edge map was as follows:

- Convert the RGB image into grayscale.
- Perform edge detection operation on grayscale image.
- Normalize edge values to the 0-255 range.
- Apply Pratt’s Figure of Merit at all threshold levels (i.e., from 0 to 255).
- Choose the image corresponding to the highest Figure of Merit value as an illustration of a good edge detection with the edge detection operator being tested.
- Perform the bitplane slicing of the image .
- Apply the edge detection on each bitplane.
- Combine the bitplanes in order of their significance.
- Apply Pratt’s Figure of Merit at all threshold levels (i.e., from 0 to 255)
- Choose the image corresponding to the highest Figure of Merit value as an illustration of a good edge detection with the edge detection operator being tested

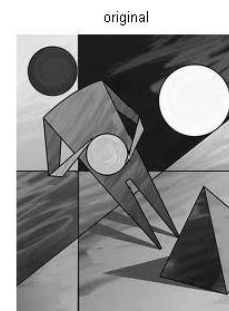


Fig. 21. “Figure.jpg” taken to assess Pratt FOM

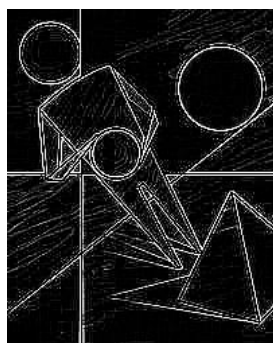


Fig. 22. Ideal Edge Map

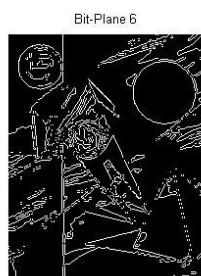


Fig. 23. Sobel applied on Bitplane 6

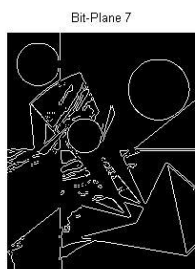


Fig. 24. Sobel applied on Bitplane 7

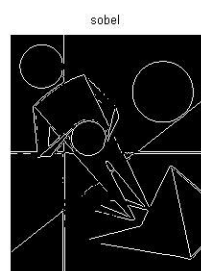


Fig. 25. Edge detected from "Sobel"



Pixel info: (X, Y) Intensity

Fig. 26. Edge detected from "MBSO"

The FOM obtained by taking the ideal edge as figure 22 for the original image shown in figure 21. The figure 24 represents the edge extracted using Sobel operator where as the figure 25 represents the edge extracted from MBSO. The edges applied on bit-planes 6 and 7 are shown respectively in the figures 23 and 24. The quantitative values are given in Table 1. These were derived by finding the difference between the ideal edge map and images got from our algorithms.

Table 1. FOM for both the algorithms on image "Figure.jpg"

Algorithm	FOM
Sobel	0.66103
MBSO	0.78210

Similarly another image in which edges are clearly visible was taken to be sure of the ideal edge map.

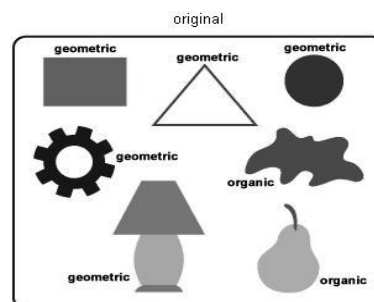


Fig. 27. "Shapes.jpg" image

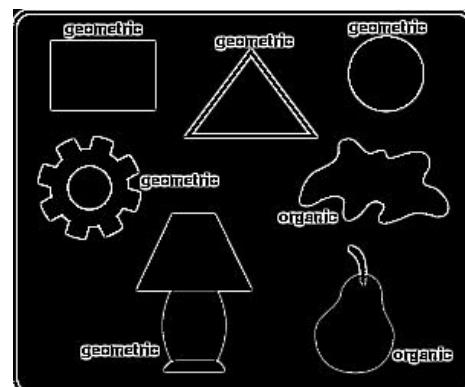


Fig. 28. "Shapes.jpg" ideal edge map

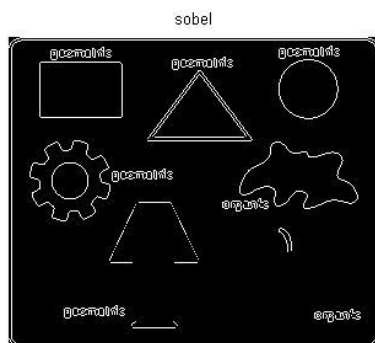


Fig. 29. Edge extracted from “Sobel” operator

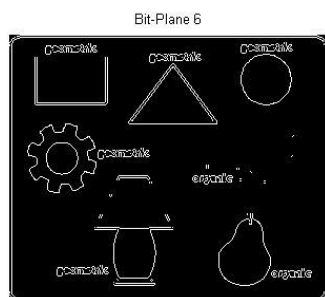


Fig. 30. Sobel applied on Bitplane6 of “Shapes.jpg”

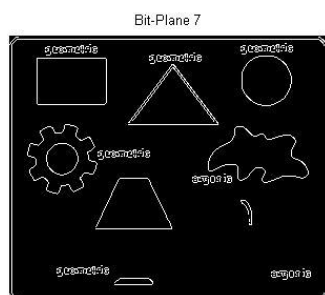
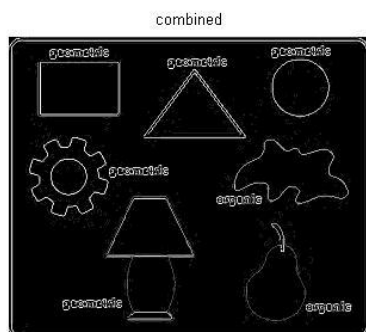


Fig. 31. Sobel applied on Bitplane7 of “Shapes.jpg”



Pixel info: (X, Y) Intensity

Fig. 32. Edge extracted from “MBSO” operator

We can clearly see that the edges of the fruit as in the ideal edge map in figure are completely lost by the Sobel filter as seen in figure whereas due to bitplane 6 edge in figure and bit-plane 7 edge in figure, the final MBSO filter gives a much better result as shown in figure . The quantitative result is as shown:

Table 2. FOM for both the algorithms on the image “Shapes.jpg”

Algorithm	FOM
Sobel	0.68750
MBSO	0.99000

Another parameter Signal to Noise Ratio (SNR) [15,16] is a chosen criteria to compare different algorithms. Both the original Sobel edge detector and the bit-plane Sobel edge detector in the context of the above mentioned classification, are selected and then tested. The above sets of images resulting from the applying of those algorithms are presented next. Table 3 and 4 describe the corresponding values of the SNR for each edge detector.

Table 3. Signal to noise ratio with noise =1 using Sobel operator

IMAGE	SNR=1	SNR=6
LENA	0.91	0.56
SHIPS	0.84	0.51
CAMERAMAN	0.88	0.52
BARN	0.92	0.60

Table 4. Signal to noise ratio with noise =1 using Modified Bit-planes Sobel operator

IMAGE	SNR=1	SNR=6
LENA	0.95	0.62
SHIPS	0.88	0.53
CAMERAMAN	0.89	0.55
BARN	0.94	0.64

4. CONCLUSION

It seems that the Figure of Merit is perhaps the most appropriate way to analyze optimal edge detection results when an ideal edge map is derivable. Humans tend to tolerate more noise. However, given that the result is a machine vision application which might not tolerate noise to the same extent that a human would, then perhaps the FOM gives a good indication of edge detection performance. Furthermore, the best way to assess the results of an edge detector is by studying its effect on the application for which it is meant. The Figure of Merit trades off showing more correct edges versus not showing erroneous edges thus leading to images that show mostly correctly detected edges and little erroneously detected edges.

In most cases edge detection constitutes only a preliminary step in an image understanding process. The edge detectors shown here should be evaluated in a broader context to verify that their functioning is consistent with our preliminary results. To this effect, more tests should be carried out on artificial and real images to fully assess the usefulness of the methods.

The Sobel mask is easy to implement as compared to other operators. Although the Sobel operator is slower to compute, it's

larger convolution kernel smoothes the input image to a greater extent and so makes the operator less sensitive to noise. The larger the width of the mask, the lower its sensitivity to noise and the operator also produces considerably higher output values for similar edges. Sobel operator effectively highlights noise found in real world pictures as edges though the detected edges could be thick. The Bit-plane Sobel operator is computationally a little expensive but the results when compared take it worth it. The visual comparison of the above sets of images can lead us to the subjective valuation of the performances of selected edge detectors. The SNR values also show improvement with the new developed technique.

5. REFERENCES

- [1] M. Sharifi, M. Fathy, M. T. Mahmoudi, "Classified and Comparative Study of Edge Detection Algorithms "Information Technology: Coding and Computing, Proceedings. International Conference 117 – 120(2002).
- [2] I.E. Abdou and W.K. Pratt, "Quantitative design and evaluation of enhancement edge detectors, " Proceedings of IEEE, Vol. 67, pp 753-763 (1979).
- [3] J. Canny, "A computational approach to edge detection", IEEE Trans. Pattern Anal. Mach. Intell. PAMI-8 679–698(1986).
- [4] G. Deng, J.-C. Pinoli, "Differentiation-based edge detection using the logarithmic image processing model", J. Math. Imaging vision 8 161–180 (1998).
- [5] R.P. Johnson, "Contrast based edge detection", Pattern Recognition, 23, 311–318 (1990).
- [6] S.S. Iyengar, W. Deng, "An efficient edge detection algorithm using relaxation labeling technique", Pattern Recognition 28, 519–536 (1995).
- [7] P. Perona, J. Malik, "Scale-space and edge detection using anisotropic diffusion", IEEE Trans. Pattern Anal. Mach. Intell. 12, 629–639 (1990).
- [8] R. Agarwal, "Bit Planes Histogram Equalization for Tone Mapping of High Contrast Images", IEEE proceedings of Computer Graphics, Imaging and Visualisation Conference, Singapore, 33-38, (2011).
- [9] R. C. Gonzalez and R. E. Woods. "Digital Image Processing". 2nd ed. Prentice Hall, (2002).
- [10] O. R. Vincent and O. Folorunso, "A Descriptive Algorithm for Sobel Image Edge Detection", Proceedings of Informing Science & IT Education Conference InSITE (2009).
- [11] Mike Heath, Sudeep Sarkar, Thomas Sanocki, and Kevin Bowyer. "A Robust Visual Method for Assessing the Relative Performance of Edge-Detection Algorithms," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 19, no. 12, pp. 1338-1359, December 1997.
- [12] Mike Heath, Sudeep Sarkar, Thomas Sanocki, and Kevin Bowyer. "Comparison of Edge Detectors: Methodology and Initial Study," Computer Vision and Image Understanding, vol. 69, no. 1, pp. 38-54, January 1998.
- [13] O. Hebb. "The organization of behaviour," in J.A. Anderson and E. Rosenfeld (eds) Neuro Computing, 1988.
- [14] W.K. Pratt Digital Image Processing, 1978 :Wiley-Interscience
- [15] Speeuwens, L.J. Heijden, F. van der, " An Edge Detector Evaluation Method Based On Average Risk", Robust Computer Vision, 40-49 (1992).
- [16] M.Heath, S.Sarker, T.Sanocki and K.Bowyer, "Comparison of Edge Detectors: A methodology and Initial Study ", Proceedings of CVPR'96 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 143-148 (1996).