A technique for signature preserving enhancement

Shilpa Palnitkar
Space Applications Centre
pshilpa@ipdpg.gov.in

Bakul Vaghela
Space Applications Centre
bakul@ipdpg.gov.in

B Kartikeyan
Space Applications Centre
bkartik@ipdpg.gov.in

Santanu Chowdhury
Space Applications Centre
santanu@ipdpg.gov.in

Abstract

Conventional techniques for image sharpening tend to introduce artifacts around edges producing a ringing effect. In this paper we propose a technique based on local derivatives to produce signature preserving sharpening. The performance of the technique is evaluated on synthetic imagery and remote sensing imagery. It is found that applying the technique in an iterative manner on zoomed imagery produces corner preserving sharpening to sub-pixel accuracy.

1. Introduction

The techniques for image enhancement can be classified into two types – point operations which are global, and spatial neighborhood techniques which are local [2]. Typical techniques for enhancement attempt to increase the contrast. The global techniques usually based on histogram analysis attempt to extend the dynamic range of gray values and also try to emphasize the brighter or darker portions of the image. However, there are many applications especially in the analysis of remote sensing imagery where the basic signature needs to be preserved for performing quantitative analysis.

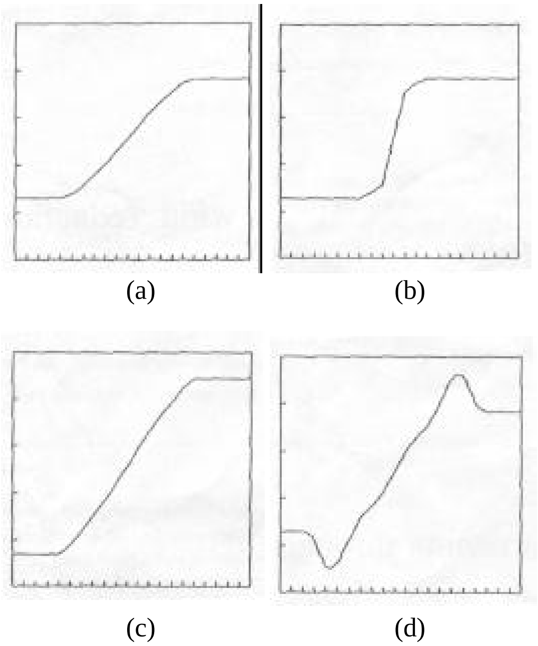


Figure 1: (a) Original ramp profile, (b) Step edge profile, (c) Effect of Contrast enhancement, (d) Effect of conventional sharpening.

The signature denotes the reflectance properties of the land cover represented by the pixel. The requirement in such images is to sharpen the edges, that is, reduce the spatial width of edges without modifying the intensity of homogeneous regions.

A typical profile of a blurred edge (without noise) is shown in figure 1(a), and the objective of sharpening is to achieve the profile shown in figure 1(b). The conventional contrast enhancement techniques only increase the dynamic range of gray value but leave the width of the edges untouched Fig.1 (c). The conventional technique for sharpening involves either a Fourier analysis, where the high frequency components are emphasized or they involve spatial convolution masks attempting to do the same. However, such techniques introduce over-shoots and undershoots around the edges as seen in Fig.1 (d). Recently there was an attempt to perform the sharpening by local analysis of gradients (Leu [3]). However the steps in his approach involved certain heuristics. In this paper we use the same basic philosophy of Leu, and develop a robust method for sharpening.

In the following section we describe our technique. This is followed by application of the technique on synthetic and real data. The results of the analysis are presented in section 3. Finally we conclude with directions for further work in automatic sharpening.

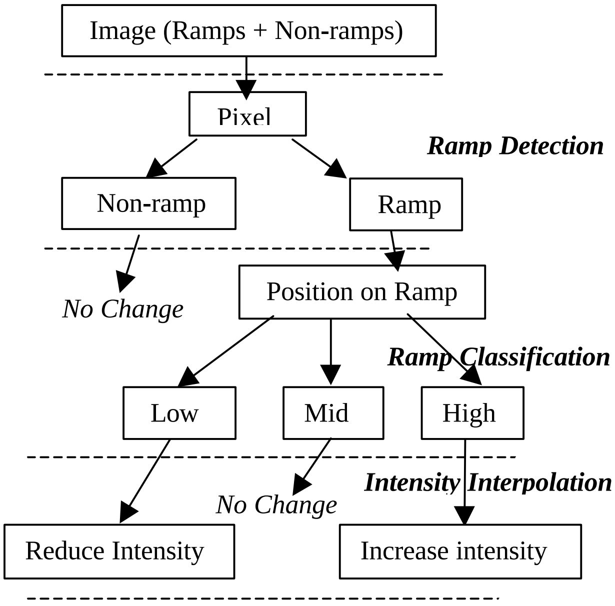


Figure 2: Process flow of Sharpening

2. The technique

The basic image model assumed is the flat facet region model, with edges at region boundaries having a ramp profile (figure 1(a)). Under this model, the sharpening process attempts to convert ramp edges into step edges (figure 1(b)). The process can be decomposed into three sub processes. For each pixel, the first step is to decide whether it is part of a flat facet or not (i.e., it is part of a ramp edge). If not, the next step is to detect which part (low, mid or high) of the edge it lies on. If it lies on the lower or higher part, the next step is to decide the sharpened value to be assigned to the pixel. The process flow is shown in figure 2.

2.1 Ramp pixel detection

The first step called 'ramp detection' can be performed using a suitable gradient estimation technique. As per the image model, the flat facet pixels should have zero (or low) gradient magnitude. Since we are dealing with a variable width edges, we propose using the Differential of Gaussian (DOG) operator for estimating the gradient. Such an operator has the advantages of a scale parameter, as also tolerance to noise due to its gaussian smoothing (see Canny [1]). Leu [3] had used the 'Sobel operator' which is of limited spatial extent. It is highly noise sensitive and unable to handle wide edges as will be shown in the next section. The DOG pair can also be used to find a reliable estimate of the direction of the edge. In this direction we expect to find a ramp profile (figure 1(a)) which we want to sharpen to a step edge profile (figure 1(b)).

2.2 Identification of pixel's position on ramp

The domain of a ramp can be divided into three regions – low, mid and high. The second step is to determine which part a given pixel belongs to. An analysis of second derivatives in the ramp direction shows that the lower part has positive 2nd derivative, and the higher part has negative 2nd derivative, with a zero crossing at the middle (figure 3).

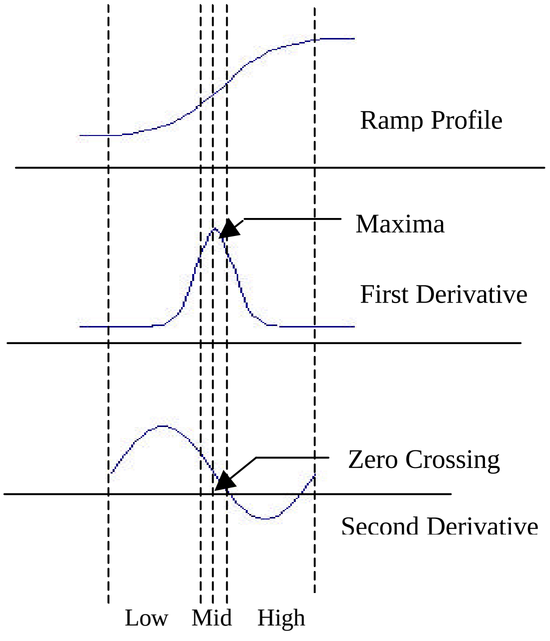


Figure 3: Derivatives of Ramp function; the extent of the Mid portion is only one pixel.

The zero-crossing of 2nd derivative is one of the well known techniques for unique edge detection (see Canny [1]). We use this property of second derivative to detect the region of the ramp that a pixel belongs to.

The directional 2nd derivatives can easily be obtained using masks for 2nd derivatives of Gaussian (Steger[4]). Let R_x, R_y be the first derivatives of image function obtained by convolving the pixel neighborhood by corresponding gaussian masks. A unit vector in gradient direction can be expressed as:

$$\mathbf{n} = n_x *\boldsymbol{i} + n_y *\boldsymbol{j} \text{ where}$$

$$n_x = \frac{Rx}{\sqrt{Rx^2 + Ry^2}} \text{ and } n_y = \frac{Ry}{\sqrt{Rx^2 + Ry^2}}$$

The second derivatives *Rxx*, *Ryy*, *Rxy* can also be obtained by convolving pixel neighborhood with corresponding gaussian 2nd derivative masks. The directional 2nd derivative in the gradient direction is given by

$$f'' = Rxx*n_x^{\,2} + Ryy*n_y^{\,2} + 2*Rxy*n_x*n_y$$

The zero-crossing may not occur in the middle of a pixel. To detect if there is a zero-crossing within a pixel's domain, we assume a 3rd order polynomial local behavior. The location of zero-crossing under this assumption is estimated as follows. Let p(t) be the gray level function in the gradient direction, with t=0 being the pixel centre

$$\begin{aligned} p(t) &= a_0 + a_1*t + a_2*t^2 + a_3*t^3 \\ p'(t) &= a_1 + 2a_2*t + 3a_3*t^2 \\ p''(t) &= 2a_2 + 6a_3*t \\ p'''(t) &= 6a_3 \end{aligned}$$

Let the zero crossing of 2nd derivative occur at $t= t_0$:

$$p''(t_0) = 0 \quad \Rightarrow \quad 2a_2 + 6a_3*t_0 = 0$$

$$t_0 = \frac{-2a_2}{6a_3} = \frac{-p''(0)}{p'''(0)}$$

If $|t_0| < 0.5$, the ramp centre lies within the domain of the pixel. In such case, we decide that the pixel is part of the middle portion of the ramp. Otherwise, the sign of the 2nd derivative indicates whether it belongs to the lower or higher part. By this approach, we ensure that the middle portion is only one pixel wide whatever be the width of the ramp. This is a necessary condition for converging to a step edge.

2.3 Estimating sharpened value

The sharpened value for a pixel is ideally the value of the flat facet on the corresponding side (lower or higher) of the ramp. A reliable estimate of this value can be obtained by interpolating from suitable neighbors. The suitable neighbors are those pixels lying on the same side of the ramp, and, the interpolation is to be obtained for one unit away from the pixel in the direction opposite to the ramp centre (figure 4)

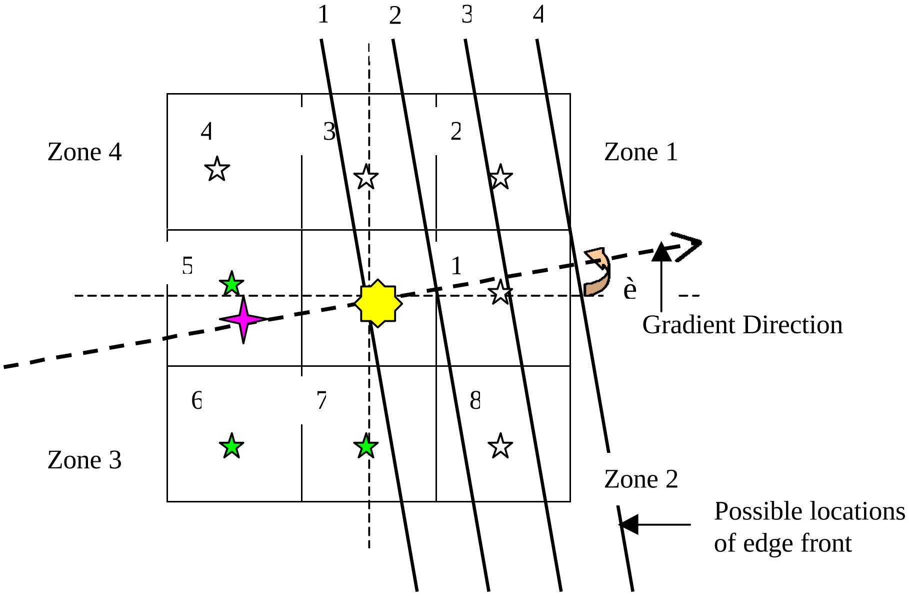


Figure 4: 3x3 pixel neighborhood and possible locations of edge front

In figure 4, we have considered a 3x3 pixel neighborhood of a pixel having an acute angle gradient direction. Each neighbor is identified by a number ranging between 1 and 8. The edge front (center of ramp) may lie on any of the bold lines 1,2,3,4. Lines 1,2 indicate that center of ramp lies in the pixel's domain, therefore the pixel is considered as central case and its intensity is left unchanged. If edge center lies on line 3 or 4, the pixel is considered to be of lower case, and its intensity is interpolated. For very small angle theta (i.e., theta almost equal to zero, the edge may pass through pixels 3,8 ,1,2 and for very large theta (i.e., theta very near to 90 degree), edge may pass through 4,3,1,2. We are absolutely sure that pixels 5,6,7 are on the lower side of the ramp since the edge will never pass through them. Hence these are the suitable pixels for calculating sharpened value. The location of the one-pixel away point is indicated by four-corner star. It can be shown that this point will always lie inside the triangle formed by centres of pixels 5,6, and 7. The choice of suitable neighbors from the knowledge of ramp direction is shown in table 1

Table 1: Choice of neighbors for interpolation

Theta (in degrees)	Zone	Pixels used for Interpolation
0-90	1	5,6,7
90-180	2	1,7,8
180-270	3	1,2,3
270-360	4	3,4,5

The parameters for the surface passing through these pixels is calculated, and used to obtain the value for the required location (one pixel away from centre). There are various options for the model of the surface passing through these points. We have attempted a simple linear model:

$$G(x,y) = a + b*x + c*y.$$

Since we assure that the suitable neighbors are not collinear, we can always find unique values for a, b and c. We then substitute the value for (x,y) for the required location to get the sharpened estimate. Other models that may be considered are the exponential decay models.

Since the one pixel away point may not be a flat facet, and may still be a ramp pixel, the technique may have to be applied iteratively till it becomes a non-ramp pixel. In practice, however, the number of such pixels do not converge to zero, since we are not handling corners (edge crossings). This also leads to rounding of corners as will be shown in the data analysis. To circumvent this problem, one simple solution we propose is to zoom the image and apply the technique so that we are able to achieve sub-pixel sharpening, and simultaneously reduce the rounding effect.

3. Data Analysis

We analyze the performance of the technique on synthetic imagery and real remote sensing imagery.

3.1 Synthetic data

We consider two synthetic images – one a circle and the other a right angled isosceles triangle. The circle image is of size 64x64 with a circle of radius 16. The circle is filled with gray level of 192 and the background with gray level 64. We also consider the same image in opposite contrast, that is fill circle with 64 and background with 192. The triangle has legs of size 32. It is also considered in the two contrasts – bright in dark and dark in bright. Each image is blurred by convolving with a 'gaussian filter mask' of varying scale parameter sigma. These input images are shown in the top rows of figures 5 and 6. The result of applying our technique on these images is also shown in these figures as difference image. The difference is between the original (unblurred) image and sharpened images. The sharpening has been attempted by both Leu's technique and our technique with different values of sigma. The difference images only show locations where the sharpened image is different from the original. The number of different pixels, and the RMS of the error are summarized in table 2.

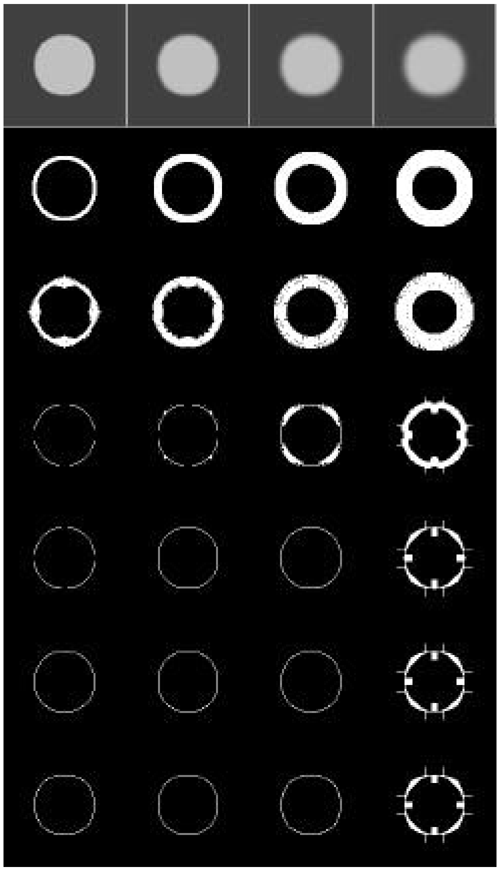


Figure 5: Blurred circle images and difference images after restoration

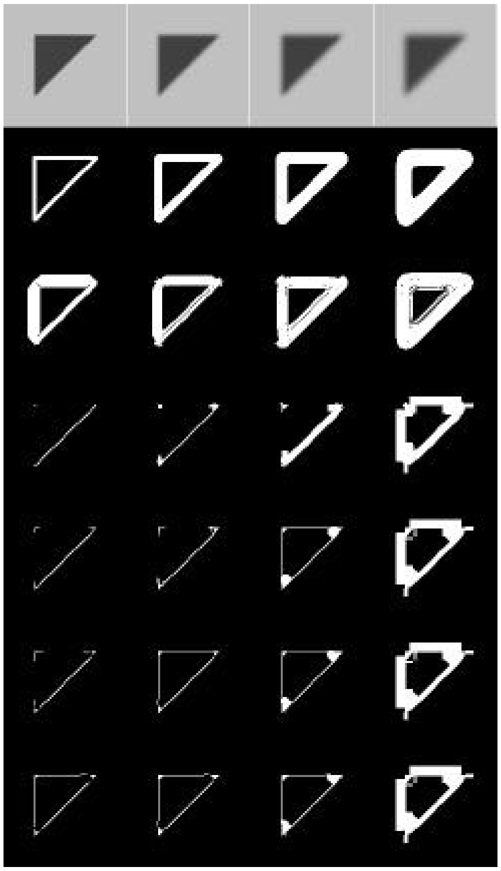


Figure 6: Blurred triangle images and difference images after restoration

Table 2A: Restoration error analysis for Circle images

Blur = 0.8		Blur = 1.6		Blur = 2.4		Blur = 3.2		
RMSE	NDIFF	RMSE	NDIFF	RMSE	NDIFF	RMSE	NDIFF	
5.8512	868	8.6273	1720	10.6157	2564	12.2715	3428	Blurred Image
7.4186	1284	8.5618	1796	8.9161	2484	7.7824	3416	LEU
4.2182	136	5.3006	184	6.2668	732	6.4603	1524	DOG sig=0.8
4.2855	152	5.4576	180	6.2616	188	6.4542	864	DOG sig=1.6
4.4201	188	5.7510	196	6.0852	188	6.2525	852	DOG sig = 2.4
4.5532	196	5.3333	188	5.9110	204	6.6561	852	DOG sig = 3.2

Table 2B:Restoration error analysis for Triangle images

Blur = 0.8		Blur = 1.6		Blur = 2.4		Blur = 3.2		
RMSE	NDIFF	RMSE	NDIFF	RMSE	NDIFF	RMSE	NDIFF	
5.8680	888	8.8685	1743	10.9995	2574	12.7606	3484	Blurred Image
8.8101	1965	10.0913	1744	10.3749	2470	10.1531	3341	LEU
4.0079	128	5.2134	212	6.2759	754	8.1834	2232	DOG sig=0.8
4.5710	139	6.1829	172	7.9406	418	8.6630	2015	DOG sig=1.6
6.2511	135	8.1157	230	9.1642	416	9.9125	2015	DOG sig = 2.4
7.8385	193	9.9838	211	10.7540	403	11.2271	2013	DOG sig = 3.2

An inspection of the images show that except for very high blurring (3.2) our technique is able to restore the images with corresponding and lower scale parameters. However, Leu's technique fails to restore perhaps due to improper choice of neighbors during the interpolation phase. The interesting patterns of the difference images for high sigma (3.2) can be attributed to the fact that the kernel size of the gaussian operators is very large in comparison to the feature size.

The top most rows of tables 2A,2B show the number of differing pixels and RMS error in the blurred image and original unblurred image. We attempt to reduce these numbers. The second row shows that Leu's method does not result in significant improvement in these numbers, and in some cases results in an increase in both RMSE and ndiff. Subsequent rows show that upto blur=2.4 (i.e when the feature size is comparable to convolving kernel size) wherever blur matches scale parameter used for sharpening, ndiff and RMSE are lowest. Some variations are seen in the triangle (table 2B) due to the corner rounding effect as seen in the patterns of the last column of figure 6.

3.2 Real remote sensing data

For real data we consider a remote sensing image of size 256x256 over a semi-urban area (figure 9a). The image is part of a scene obtained from the PAN sensor of IRS-1D, and has a ground resolution of 5.8 m. The portion of the image chosen shows a wide variety of features ranging from dense urban, large buildings, vegetation, and canal.

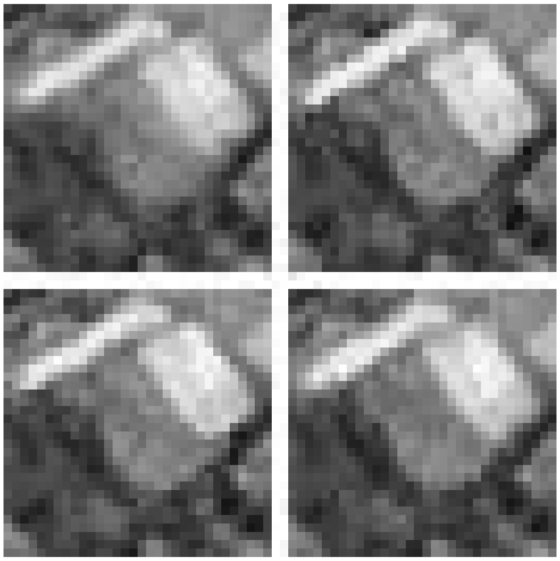


Figure 7: Effect of scale parameter; (left top) original, (right top) sharpening by Leu's method, (left and right bottom) sharpening with sigma = 0.8 and 1.6.

Scale Parameter: We have experimented with different values of sigma, the scale parameter, and the results are shown in figure 7 comparing with Leu's method which uses the Sobel mask. This mask can be thought of as equivalent to a sigma of 0.4. The use of large sigma is able to sharpen wide edges. However, it also tends to round the corners.

Iterative application: In figure 8 we show the iterative application of the technique for one and three iterations. Table 3 summarizes the number of pixels detected as either non-ramp or as different parts of ramp for 16 iterations. As expected, the number of mid-ramp pixels stabilize after a few iterations, and the number of high and low ramp pixels decrease as they are converted to non-ramp pixels. Further, there is no perceptible difference in the image after 3-4 iterations. While initially 35% of the image was either high or low ramp, within 4 iterations it has come down to 5%. However, here too we observe that the corners tend to get rounded.

Table 3: Ramp statistics for iterative processing

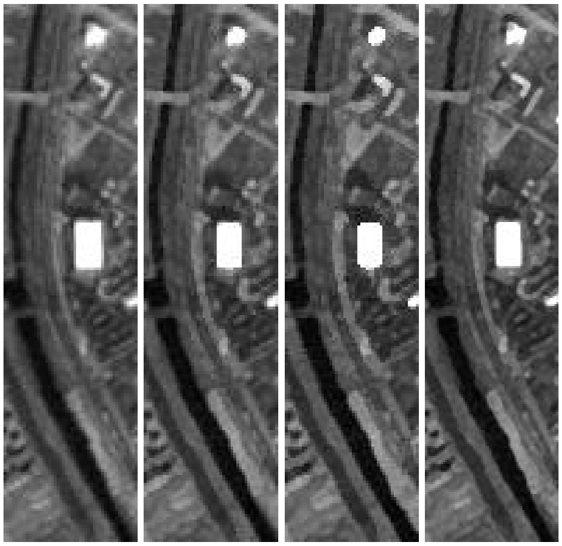
Non-ramp	Low	High	Middle
25873	10656	10252	12755
33977	6630	6570	12358
39089	3979	4278	12190
42593	2268	2565	12109
44591	1332	1572	12041
45850	754	951	11981
46553	475	570	11938
46992	284	326	11934
47224	175	209	11928
47414	87	119	11916
47487	57	85	11907
47547	32	56	11901
47566	22	41	11907
47588	12	32	11904
47610	7	14	11905
47616	4	9	11907

Processing on zoomed image: To avoid the rounding of corners, we suggest zooming the image and applying the sharpening. The image was zoomed twice using the bi-cubic kernel and sharpened. The results are again shown in the same scale as the original in figure 8(d). It can be clearly seen that the rounding effect is reduced when we analyze the zoomed image. Figure 9(b) shows the result of zooming the image 4 times, using scale parameter 1.6, and iterative application of the sharpening technique after 4 iterations.

4. Conclusions

We have proposed a robust method for sharpening of images under the assumption of flat facet image model. The technique is based on the behavior of the first and second directional derivatives of the gray level function. The derivatives are estimated by convolving the discrete image with gaussian derivative masks so as to handle noise tolerance and variable width edges without resorting to a parametric model for the image function. The analysis of results on synthetic images show that the technique is robust. We suggest applying the technique in an iterative manner and show the results on real remote sensing panchromatic images.

Further investigations have to be carried out to dynamically determine the appropriate scale parameter for use in the sharpening process. Further, model for detecting and sharpening of corners has to be developed.



(a) (b) (c) (d)

Figure 8: Effect of iteration and zooming; (a) original, (b) after 1 iteration, (c) after 3 iterations, and (d) zoomed twice and sharpened after 3 iterations. Sigma = 1.6.

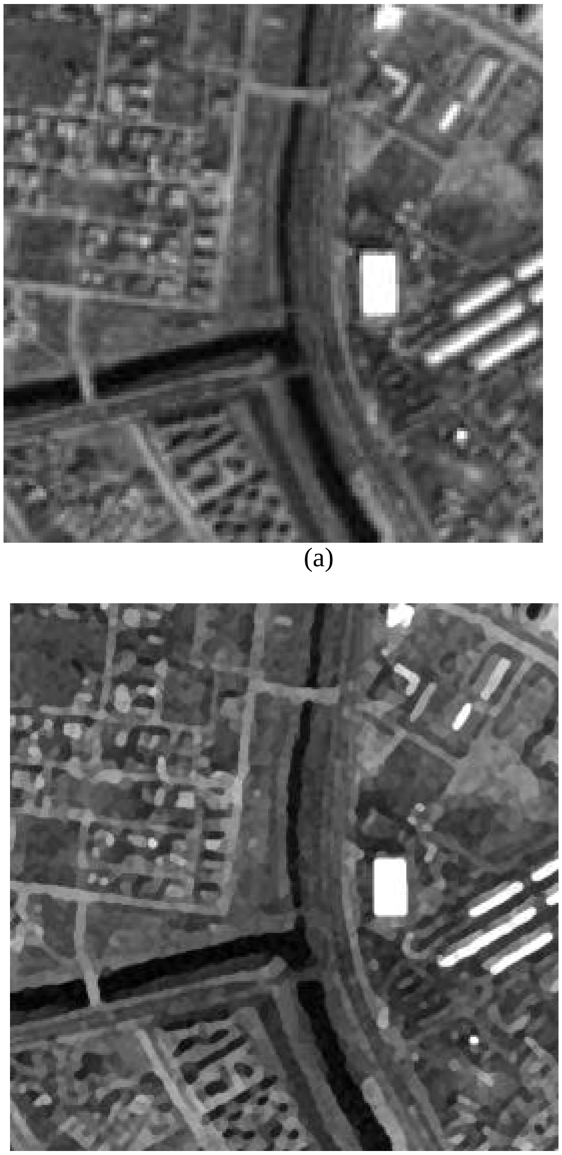
References

[1] J Canny. A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI*- 8, 1986, pages 679-698.

[2] A K Jain. *Fundamentals of Digital Image Processing*, Prentice-Hall, Englewood Cliffs NJ, 1989.

[3] J G Leu. Edge sharpening by ramp width reduction, *Image and Vision Computing*, *PAMI*-18, 2000, pages 501-514.

[4] C Steger. An unbiased detector of curvilinear structures, *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*-20, 1998, pages 113-125.



(a)

(b)

Figure 9: (a) Original 256x256 image and (b) final sharpened result after zooming 4 times and sharpening after 4 iterations, with sigma = 1.6