Edge preservation with space-filling curve half-toning

John W. Buchanan
Oleg Verevka
Department of Computing Science
University of Alberta
Edmonton, AB., Canada
T6G 2H1
{juancho,oleg}@cs.ualberta.ca

Abstract

Accurately displaying a grey-scale image on a printer requires that the image be half-toned. That is, the image is approximated by sets of white and black pixels whose local average intensity is similar to that of the original image. In the case of laser printers these black and white pixels should be clustered because pixels cannot be set independently. By using a space-filling curve it is possible to develop clustered sets of pixels that approximate the image.

Unfortunately this technique can destroy the edges in the resulting image. In this paper we present two solutions to the edge destruction problem. The first solution uses an edge detection filter to determine when the region size should be changed. By ensuring that none of the regions cross an edge the resulting image will contain a good representation of the edges. The second solution uses a local sort of the region in order to determine where the black and white pixels are placed. When the regions are small the resulting black and white pixels are still clustered but are positioned in such a way that edges are highlighted.

Résumé

L'affichage précis d'une image en niveaux de gris sur une imprimante exige que l'image soit similigravée. Il s'agit d'approximer l'image par un ensemble de pixels noirs et blancs dont l'intensité moyenne est similaire à celle de l'image originale. Dans le cas d'imprimantes au laser, ces pixels noirs et blancs se doivent d'être agglomérés parce que les pixels ne peuvent être sélectionnés indépendamment. En utilisant une courbe de remplissage de l'espace, il devient possible de construire des ensembles d'agglomération de pixels qui approximeront l'image.

Malheureusement, cette technique peut détruire les arêtes dans l'image produite. Dans cet article, nous présentons deux solutions au problème de destruction des arêtes. La première solution utilise un filtre de détection des arête pour déterminer quand la grandeur de la région doit être changée. En s'assurant qu'aucune région ne traverse un arête, l'image résultante contiendra une bonne représentation des arêtes. La deuxième solution utilise un ordre local d'une région pour déterminer où les pixels blancs et noirs doivent être placés. Quand les régions sont petites, les pixels blancs et noirs résultants sont encore agglomérés, mais sont positionnés de façon à ce que les arêtes soient rehaussées.

Keywords: Half-toning, dithering, grey-scale, space-filling curve, error propagation.

1 Introduction

Current printing technology is mostly limited to a discrete ink deposition process. This means that grey-scale images or colour images must be processed so that they can be printed. Grey-scale images (and colour images) can be printed by distributing black and white (or colored) dots on the paper so that the perceived local intensity approximates the grey-scale (colour) value of the image. In this paper we will be dealing with the half-toning of grey-scale images [Floy76, Witt82, Knut87, Ulic87, Ulic88, Velh91, Ostr94, Zhan93]. One approach, known as error propagation half-toning, is to approximate a region of the image[1] with an appropriate selection of white and black pixels. Once the pixels are approximated the quantization error can be computed and distributed to neighboring regions. The propagated error is incorporated in the half-toning of the neighboring regions.

An interesting way of distributing the error was first proposed by Witten and Neal [Witt82] and later extended by Velho and Gomez [Velh91]. The basic idea is to distribute the error along a space-filling path. At each pixel a decision is made based on a threshold whether or not to set the pixel. The quantization error resulting from setting or not setting the pixel is computed and added to the next pixel along the path.

Velho and Gomez [Velh91] showed that by processing the space-filling curve in segments clustered sets of pixels were produced. For each of these segments a local average is computed. This local average determines the number of pixels to be set in the region. They divided the curve segment into white and black segments. The white segment of the curve is centered around the brightest pixel in the region. As with Witten and Neal the quantization error is propagated to the next segment of the curve.

The space-filling algorithm presented by Velho and Miranda can be described by the algorithm in figure 1.

```
error = 0;
WHILE ( pixels left to be processed )
| Find the next n pixels to be processed |

  pixels = get_next_run(n);
  average = average_intensity(pixels);
  white = ⌊average*n ⌋;
| Set the white pixels around the pixel with maximum intensity. |

  set_white(pixels,white);
  set_black(pixels,n-white);

| Add any left over error to the accumulator |

  error += average - white/n;
  IF ( error > 1 ) then
    pixels = get_next_run(1);
    set_white(pixels,1);
    error = error -1 + pixels[1];
  END if
END while
```

Figure 1:
Pseudo-code for space-filling curve half-toning

2 Edge preservation

In an attempt to highlight edges or fine details of an image Velho and Gomez centered their white pixels around the brightest pixel in the region. A simple example illustrates the weakness of this approach. Consider the pixels in figure 2. The path defining the region crosses an edge between a dark and light area. The pixel with the circle in it is the brightest pixel in the light area. If the dithering of the region requires that three pixels be set to white then the resulting group of white pixels will destroy the edge.

The destruction of edges is a known problem with half-toning techniques. A solution that works for most half-toning techniques is to enhance the edges of the original image [Jarv76, Knut87]. Another approach is to use a half-toning method that tends to preserve edges in the resulting images[Ulic87, Ulic88].

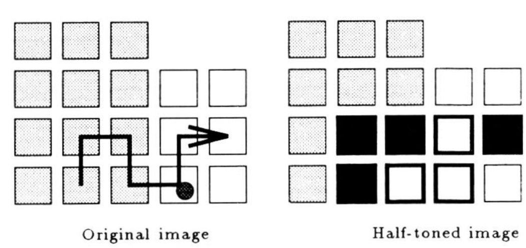

Figure 2:
The cluster defined by the space-filling curve can destroy an edge. In this example the circle indicates the brightest pixel. Centering the three white pixels around it means that one of the original black pixels is set to white.

[1] A region is either a pixel or a set of pixels depending on the technique.

Enhancing the edges of the original image does improve the half-toned image when the space-filling curve method is used. However, it does not completely remove the problem since clusters of pixels may still straddle an edge. In this paper we introduce two methods for edge preservation using the space-filling curve half-toning method. The first method uses an edge detection filter to determine the size of the regions used for half-toning. By adjusting the region size so that no region crosses an edge we ensure that edges are preserved. Our second method uses a sort operation to ensure that the brightest pixels in the region are the pixels that are set to white. Setting the pixels in this manner ensures that the white and black pixels lie on the correct side of any edge present in the region.

Two of the images we used in our testing are presented in figures 3 and 4. The first is digitized from a lithograph print [Hurd68] and the second is similar to the test image in Velho and Gomez's paper [Velh91]. An additional test image is derived from Knuth's paper [Knut87] and is used at the end of the paper.

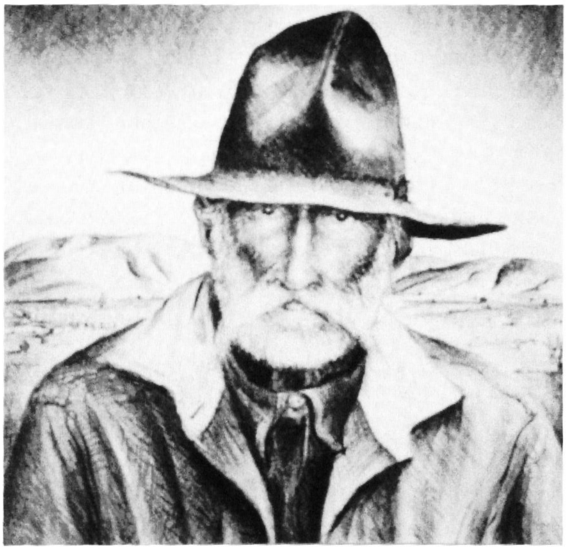

Figure 3:
Photograph of Sheepherder test image. The image was digitized from [Hurd68].

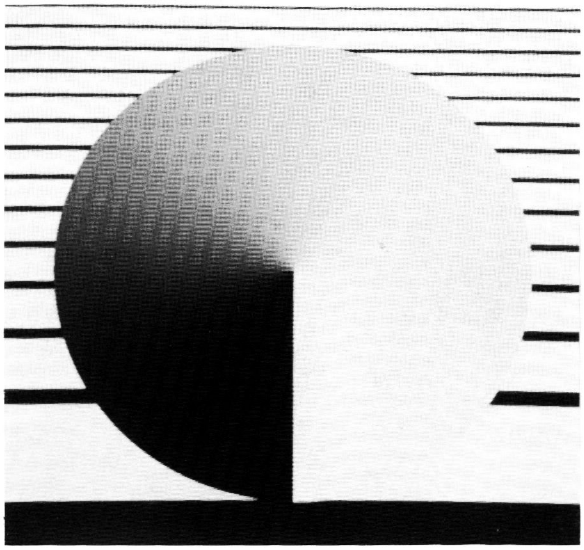

Figure 4:
1k × 1k computer generated test image displayed on a monitor and photographed. This image is based on the image used in [Velh91].

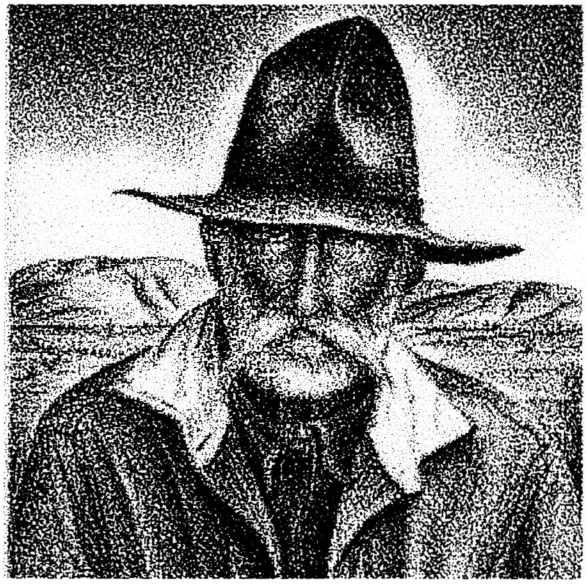

Figure 5:
Test Sheepherder image displayed using the space-filling algorithm with a region size of 19. This results in 20 levels of gray.

3 Preserving the edges

The edge destruction problem is illustrated in the display of the two test images (figures 5 and 6. The Sheepherder image by Peter Hurd contains a large number of fine details that have been blurred in

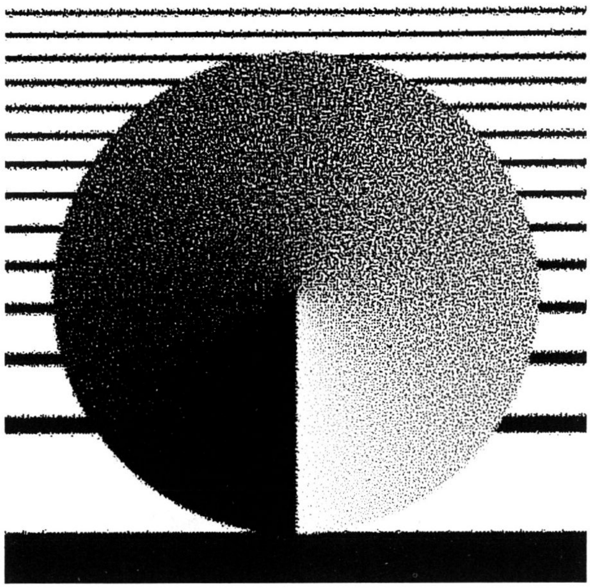

Figure 6:
Test computer generated image displayed using the space-filling algorithm with a region size of 19. This results in 20 levels of gray.

the display (figure 5). The edge blurring problem is more visible in the display of the computer generated image (figure 6 As we discussed earlier this edge blurring is caused by the clustered sets of pixels straddling the edge. Jarvis and Roberts [Jarv76] found that by preprocessing the image with an edge enhancing filter the resulting dithered images were better. Even though dithering an enhanced edge version of the test image improves the edge representation somewhat, it was found that the use of edge enhancement did not address the problem of edge destruction with the space-filling half-toning technique.

3.1 Edge detection

The destruction of the edges is caused by the clusters straddling an edge in the original image. If we can ensure that none of the regions straddle an edge then we can also ensure that none of the approximating clusters straddle an edge. With this view in mind we developed our first solution to the problem, namely the use of a edge detection filter.

Our first attempt to solve the problem used a two-dimensional edge detection[2] filter that was applied to the image. The resulting edge information was stored in a separate frame buffer. In order to use this information we altered the space-filling curve method so that when the regions were being generated an edge check was performed prior to the addition of a new pixel to the region. If an edge is encountered then the region growth process is terminated and the shortened region is used to generate the approximating clusters of black and white pixels. This first implementation produced the desired results, but required the evaluation of a two-dimensional edge detection filter for each pixel. The cost of this approach was prohibitive and in fact the edge detection operation was costing more than the half-toning.

A simple observation allowed us to use the appropriate edge information with less computational overhead. When a region is being constructed we only need to know whether moving to the next pixel along the path causes the crossing of an edge. Determining if an edge is being crossed can be accomplished by comparing the difference in pixel intensities to a user-defined threshold. If the intensity of two pixels differs by more than this threshold then the region will not be allowed to cross this edge. This new edge-detection method only costs an additional compare per pixel.

Our method is quite similar to a method proposed by Velho and Gomez [Velh92]. They used the local gradient of the pixel to determine the size of the regions. They also observed that only the component of the gradient parallel to the path needed to be considered.

Edge detection results

In figure 7 we present the sample image displayed with a region size of 19 and an edge detection threshold of 40. The image of the Sheepherder displayed with the same parameters is presented in figure 8.

3.2 Reordering the region

In their original paper Velho and Gomez attempted to preserve the high frequency components of the image by centering the white approximating pixels around the brightest pixel in the region. In general this approach does not preserve the edges in an image since the brightest pixel may not be in the center of the brightest portion of the image segment.

However, if we could place the white pixels in the positions of the brightest original pixels then the local intensity distribution is better approximated in the resulting image. The preservation of

[2] The edge detection filter was the one available in the xv image processing program.

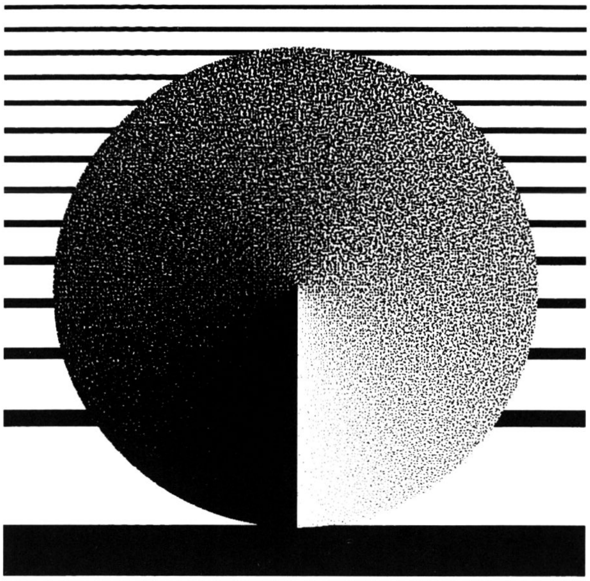

Figure 7:
Test image displayed using edge-detection half-toning with a region size of 19 and an edge threshold of 40.

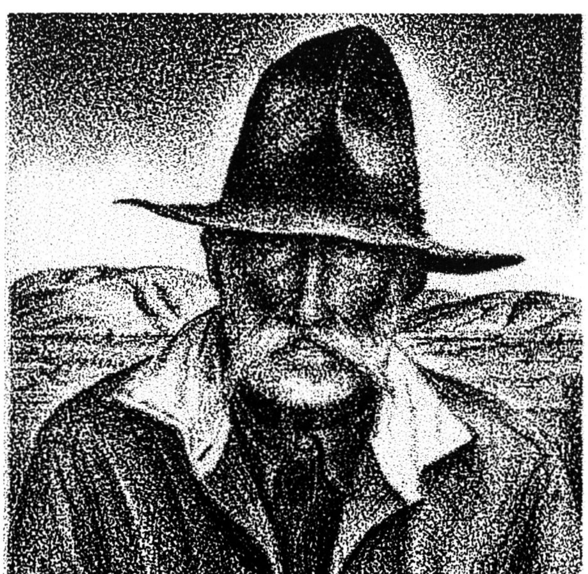

Figure 8:
Sheepherder displayed using edge-detection half-toning with a region size of 19 and an edge threshold of 40.

the local intensity distribution results in a highlighting of the fine details of the image. An easy way to ensure that the local intensity distribution is preserved is to sort the pixels of a segment according to their intensity. The resulting ordered pixels are then used to indicate where the white and black pixels must be set.

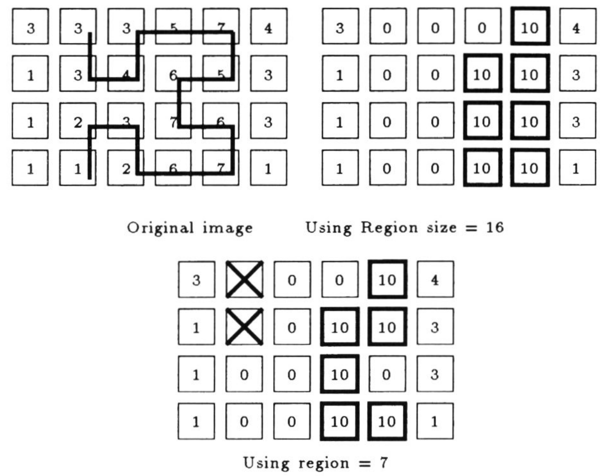

Figure 9:
Black and white pixels resulting from sorting the pixels and setting the brightest to white. Two region sizes are illustrated (16 and 7). The pixels marked with X correspond to two pixels that belong to the next region.

A potential problem with this approach is that the sets of black and white pixels chosen may no longer be clustered. However, if we choose a space-filling curve whose segments define tight regions of the image and the size of these regions is kept small the black and white pixels will be clustered. This is illustrated in figure 9 where we show the result of a sorted half-toning of two different region sizes. The resulting clusters of white and black pixels are still connected.

By setting the brightest pixels to white in a region we ensure that the fine details of the image are preserved. As is often the case there is a trade off in this technique. As we increase the region size the quality of the fine detail display seems to get better. However, the increased quality of fine detail display is achieved at the cost of introducing artifacts in regions with a uniform gradient. These artifacts are due to the consistent positioning of the white pixels in the bright area of the region. If the

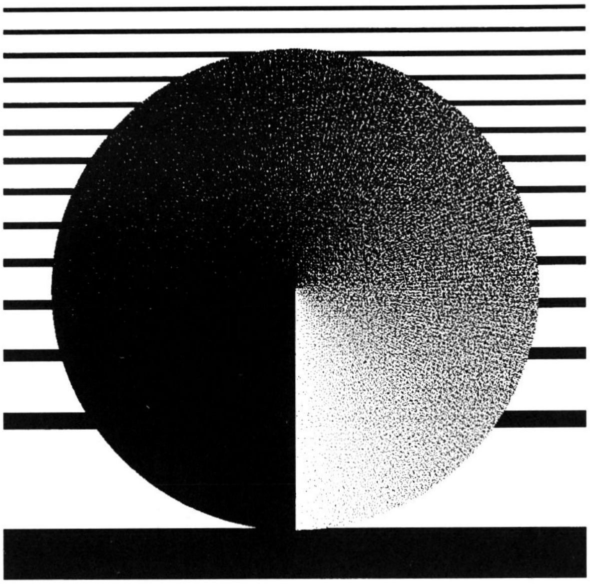

Figure 10:
Test image displayed using sorted space-filling half-toning with a region size of 7. The pixels in the region are sorted according to their original intensity before being assigned a black or white value.

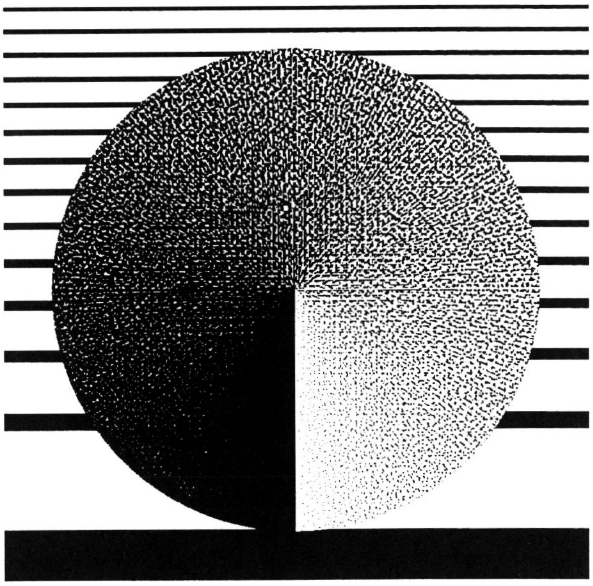

Figure 11:
Test image displayed using sorted space-filling half-toning with a region size of 41. The pixels in the region are sorted according to their original intensity before being assigned a black or white value. Notice that the clusters are exhibiting some regular patterns due to the uniformity of the local gradient.

gradient field is not uniform these artifacts are not quite so visible.

Sorted pixel results

We have found that reasonable displays are generated using curve segments in the range of 2-50 pixels (see figures 10 - 11). As the size of the regions increases the images take on a distinct look, this is illustrated in figure 14, where the size of the approximating regions varies as a quadratic function of the x coordinate. By sorting the pixels in the regions the high frequency components of the image are preserved. In fact, as the cluster size increases towards the right side of the image we see that lines and edges are accentuated greatly and that the grey scale reproduction quality decreases. Compare this image with a similar image produced using edge detection for adaptive region size (figure 15). For completeness sake we present the same image using no edge enhancement whatsoever in figure 13

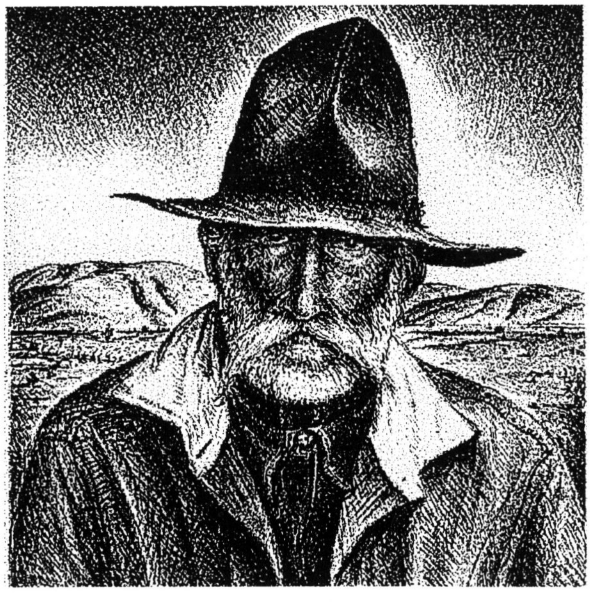

Figure 12:
Test image displayed using sorted space-filling half-toning with a region size of 19.

Figure 13:
Test image displayed using space-filling half-toning with a region size of ranging from 1 to 400 as a function of $(20x)^2$.

Figure 15:
Test image displayed using edge-detection space-filling half-toning with a region size of ranging from 1 to 400 as a function of $(20x)^2$. Adaptive regions are used with an edge threshold of 40.

Figure 14:
Test image displayed using sorted space-filling half-toning with a region size of ranging from 1 to 400 as a function of $(20x)^2$. Notice the different look of the image from right to left.

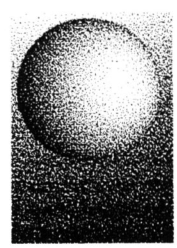

Figure 16:
Knuth's test image using a step size of 7.

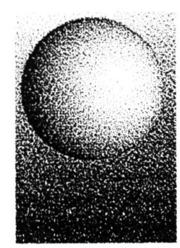

Figure 17:
Enhanced edge version of Knuth's test image using a step size of 7.

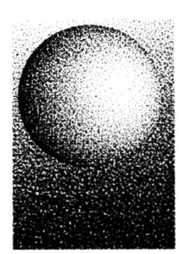

Figure 18:
Knuth's test image using a step size of 7 and an edge threshold of 40.

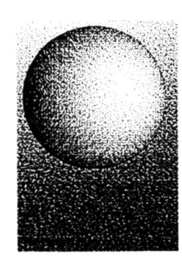

Figure 19:
Knuth's test image using a step size of 7. Clusters are sorted.

4 Conclusions

Two techniques were presented for preserving edges when a space-filling curve half-toning method is used. The first used an edge detection filter along the path. Our method is closely related to an edge enhancement method proposed by Velho and Gomez [Velh92]. By altering the region size so that no region is straddling an edge we ensure that no cluster crosses an edge and thus edges are preserved. The cost of this method is an additional compare per pixel.

The second method uses a sort operation to ensure that the brightest pixels in a region are those that are set to white. Relying on the local structure of the Hilbert curve and restricting the size of the curve segments ensures that the resulting black and white pixels are fairly well clustered. In figures 16-19 we present the display of Knuth's test image from [Knut87]. In each of these displays the region size chosen is 7. Applying Jarvis's edge enhancement filter to the image enhances the edges in the resulting image (figure 17). The quality of the edges are similar to those achieved by using an edge detection threshold of 40 (see figure 18). By far the best display of the edges is achieved by sorting the pixels (see figure 19. Naturally the cost of this method increases with the region size. However, the method provides a good way in which half-toned images can highlight the fine details of the image. This highlighted display of the edges is achieved at the cost of a lower quality display of areas in which there is a uniform gradient distribution.

Thanks

To the members of the computer graphics lab at the University of Alberta for their feedback and patience while we printed out these images again and again and.....

To members of our families who endured the late hours.

References

[Floy76] R.W. Floyd and L. Steinberg. "An adaptive algorithm for spatial grey scale". *Proc. Soc. Inf. Display*, Vol. 17, pp. 75–77, 1976.

[Hurd68] P. Hurd. *Peter Hurd, The lithographs.* The Baker Gallery Press, P.O. Box 1920, Lubbock, Texas, 1968.

[Jarv76] J.F. Jarvis and C.S. Roberts. "A new technique for displaying continuous tone pictures on bilevel displays". *IEEE Transactions on Communications*, Vol. COM-24, pp. 891–898, 1976.

[Knut87] D. E. Knuth. "Digital halftones by dot diffusion". *ACM Transactions on Graphics*, Vol. 6, No. 4, pp. 245–273, October 1987.

[Ostr94] V. Ostromoukhov, R. D. Hersch, and I. Amidror. "Rotated Dispersed Dither: A new Technique for Digital Halftoning". *Computer Graphics (SIGGRAPH '94 Proceedings)*, pp. 123–130, July 1994.

[Ulic87] R. Ulichney. *Ditital Halftoning.* MIT Press, Cambridge, Massachusetts, 1987.

[Ulic88] R. A. Ulichney. "Dithering with blue noise". *Proc. IEEE*, Vol. 76, No. 1, January 1988.

[Velh91] Luiz Velho and Jonas de Miranda Gomes. "Digital halftoning with space filling curves". *Computer Graphics (SIGGRAPH '91 Proceedings)*, Vol. 25, No. 4, pp. 81–90, July 1991.

[Velh92] L. Velho and J. M. Gomes. "Space Filling Curve Dithering with Adaptive Clustering". *Proceedings of SIBGRAPI'92 – V Simposium of Computer Graphics and Image Processing*, pp. 1–9, 1992.

[Witt82] I.H. Witten and R.M. Neal. "Using Peano curves for bilevel display of continuous-tone images". *IEEE Computer Graphics and Applications*, Vol. 2, No. 3, pp. 47–52, May 1982.

[Zhan93] Y. Zhang and R. E. Webber. "Space diffusion: An improved parallel halftoning technique using space-filling curves". *Computer Graphics (SIGGRAPH '93 Proceedings)*, pp. 305–312, August 1993.