October 1989

Active Vision With Multiresolution Wavelets

Howard Choset
University of Pennsylvania

Ruzena Bajcsy
University of Pennsylvania

Helen Anderson
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-73.

Active Vision With Multiresolution Wavelets

Abstract

A wavelet decomposition for multiscale edge detection is used to separate border edges from texture in an image, toward the goal of a complete segmentation by Active Perception for robotic exploration of a scene. The physical limitations of the image acquisition system and the robotic system provide the limitations on the range of scales which we consider. We link edges through scale space, using the characteristics of these wavelets for guidance. The linked zero crossings are used to remove texture and preserve borders, then the scene can be reconstructed without texture.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-89-73.

Active Vision With Multiresolution Wavelets

MS-CIS-89-73
GRASP LAB 196

Howard Choset
Ruzena Bajcsy
Helen Anderson

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104-6389

October 1989

Acknowledgements:
This work was in part supported by NSF grant DCR-841077, Air Firce grant AFOSRF 49620-85-K-0018, Army/DAAG-29-84-K-0061, NSF-CER/DCR82-19196 A02, NASA NAG5-1045, ONR SB-35923-0, NIH grant NS-10939-11 as part of Cerebro Vascular Research Center, NIH 1-R01-NS-23636-01, NSF INT85-14199, NSF DMC85-17315, ARDP N00014-K-0632, NATO grant 0024/85. by E. I. DuPont & Nemours Co., DEC Corp., and Lord Corp.

Abstract

A wavelet decomposition for multiscale edge detection is used to separate border edges from texture in an image, toward the goal of a complete segmentation by Active Perception for robotic exploration of a scene. The physical limitations of the image acquisition system and the robotic system provide the limitations on the range of scales which we consider. We link edges through scale space, using the characteristics of these wavelets for guidance. The linked zero crossings are used to remove texture and preserve borders, then the scene can be reconstructed without texture.

1 Introduction

Scene segmentation on digitized images is an ill defined and difficult task. This is a data reduction process, which consists of requantizing the sensory measurements into regions which are determined to be uniform by some measurement criteria. Then some level of sensory resolution, less than the original resolution, is substituted. We use a multiscale approach to find borders for use in segmentation and to remove texture within the regions. We limit the number of scales to consider by using a wavelet decomposition, then combine edge information at different scales to obtain information on the nature of edges for use in segmentation.

In this work, we limit ourselves to visual, non-contact measurements obtained from a CCD camera. In general, we would not have a priori reasons to choose a particular scale, but we are always limited by the format of the image itself to a particular minimum and maximum resolution which we consider [1]. Therefore, perceptual limitations can guide our range of scales. Also, the camera noise level guides the selection of the minimum amplitude of a signal which we consider. Using Active Perception, we assume that the camera can and should be moved or the lenses adjusted to best sense the scene [2]. In addition, there are physical limits for a two robot system, where one robot has an end effector and the other has a camera. The distance from the camera to the boundary of the workspace can be calculated using kinematics [3]. The minimum size object that can be handled by the end effector is also known. In this way, physical and perceptual limitations guide our choice of scales.

To visualize a coarse to fine representation, consider the appearance of the outside of a building. First you see the the outline of the building; you notice its borders and differentiate the building from the rest of the scene. After a small amount of inspection, you see windows and entrances. These "features" are object borders, but they are subregions of the original. As you approach the building, you notice the bricks on the face of the building and the shingles on the roof. More specifically, you observe the texture inside each of the regions in the scene. You started at a coarse resolution and worked your way towards the fine resolution. When you were far from the building, the pattern of windows was the texture, but as you got closer, the windows became the features while individual bricks became the texture. Closer still, the bricks become features and the patterns of imperfections and coloring of the bricks become the texture. In other words, what constitutes texture is dependent on scale, both for the maximum size and the minimum size object available within the resolution limits of the imaging system. Similarly, the size of an object in the scene which is big enough to be considered for further analysis (a "feature") depends on resolution. The use of multiple scales for edge detection was discussed in [4],[5],[6],[7] and others.

As a first step toward segmentation, our goal is to separate borders, shading, and texture in an image. Both the largest and the smallest resolvable object sizes are determined by the optics of the camera lens, the size of the CCD and the distance between objects and the camera. Since segmentation, by its very nature, is a data reduction process, we assume that the number of regions is much smaller than the number of pixels in the original picture.

For example, an original image of 512 by 512 pixels might be reduced into approximately 25 regions, a reduction of 4 orders of magnitude. A reduction of more than 5 orders of magnitude or less than 2 is not likely to be a "good" segmentation, since the segmentation would not optimally be using the resolution of the camera. In other words, if the size of a particular object in the image is only a few pixels, then analysis of the shape of the object would be difficult. Similarly, if the object is larger than the whole image, the outer shape of the object cannot be determined. In order to do shape analysis, the camera would have to zoom in or out, or physically be moved to appropriately sense the scene. In a robotic vision domain, we assume that this can and should be done.

Within the resolution of the image, the information present only at high spatial frequency can be considered *texture*, while the information present at all resolutions is likely to be *object borders*, or edges. Finally, the information present only at low spatial frequencies can be considered *shading*.

We decompose an image into localized information at different spatial frequencies using wavelets as a set of basis functions. Wavelets are families of basis functions that can be represented by $f(x) = \sum b_{jk} W(2^j x - k)$ [13]. They are used for the purpose of localization of signals in both space and spatial frequency, as far as possible, simultaneously.

First, we discuss the one dimensional case. We work with one dimensional images which are also referred to as signals. Using wavelets in a fine to coarse manner, we decompose the signal onto four levels varying in resolution by a power of two. Then we create the zero crossing representation. Using the zero crossing representation, we identify the texture in the image and show the effect of texture deletion.

Finally, we extend our results into two dimensions. We show scenes where both horizontal and vertical texture are deleted.

2 Dyadic Wavelet Decomposition

A dyadic wavelet transform is a nonlinear multiscale transform which translates when the signal translates. It can be viewed as a discretization along the scale axis of a continuous wavelet transform and was first investigated by [8]. We choose wavelets as our basis functions for several reasons. First, we can represent a signal with only a few wavelets without losing too much information. This allows us to select a small number of scales which effectively represent the edge information [9]. Second, wavelets are highly localized in both real and frequency space. The wavelets we choose are the second derivative of a smoothing function, thus the zero crossings of the dyadic wavelet decomposition provide the locations of edges in the signal.

Because only four different scales are used, the wavelet transform produces data compres-

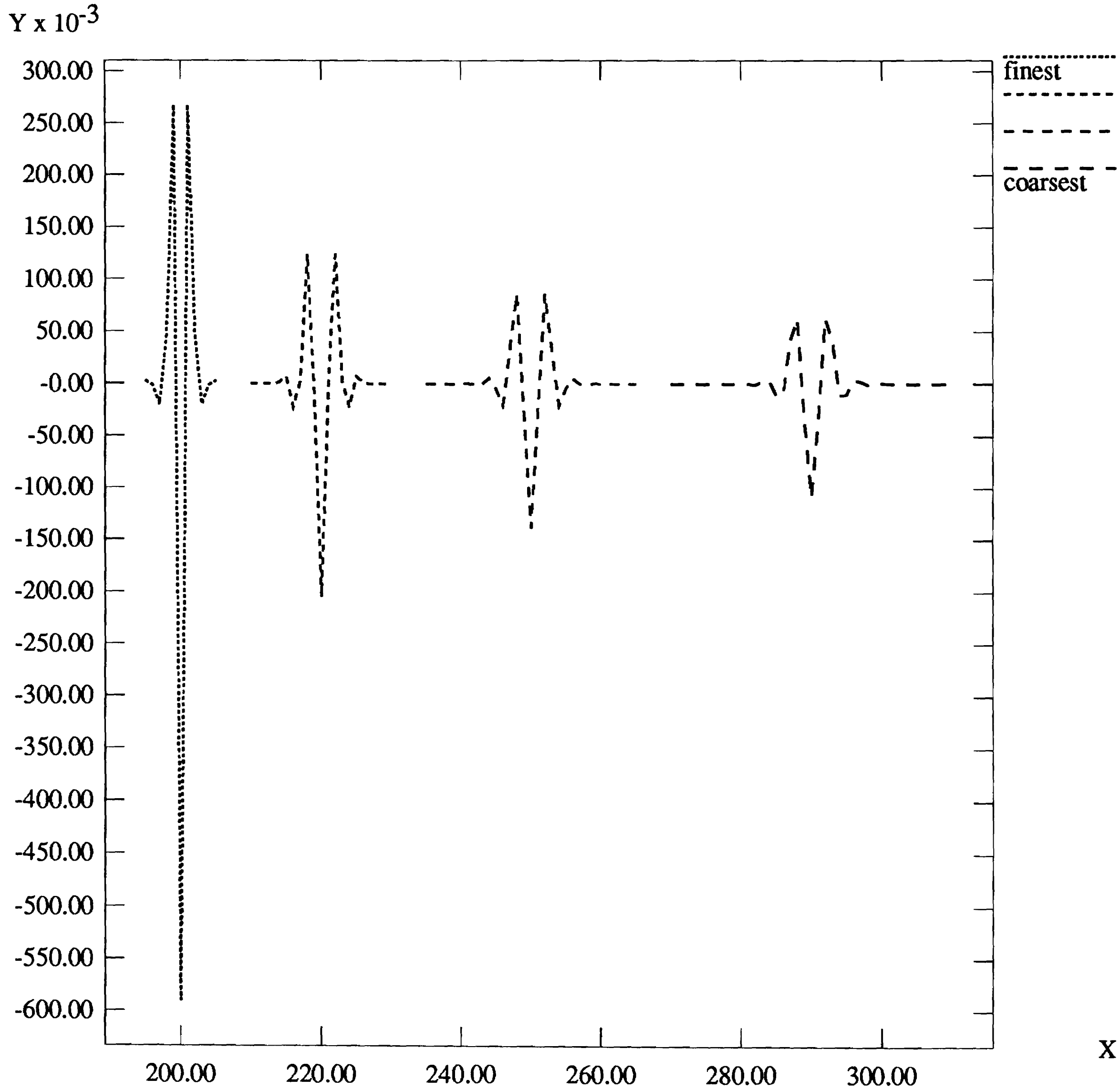

Figure 1: *The four wavelets used in decomposition.*

sion; however, the representation has been shown to be experimentally complete, in that a signal can be reconstructed from its dyadic decomposition with minimal loss of information. For example, a printed image constructed from the 2-dimensional wavelet transformation appears the same as the original and the intensities change by $< 5\%$ [10]. However, for this application, the selection of the appropriate scales is more important than data compression.

To decompose a signal, we use two different filters: a smoothing function, and a zero crossing function. Using these two filters, we create a series of wavelets whose spatial resolution increases in width by powers of two.

The decomposition process consists of a series of convolutions. The original signal is convolved with a smoothing filter 3 times, yielding the singly-smoothed signal, the doubly-smoothed signal and the triply-smoothed signal. The smoothed signals each have half of the resolution of the previous signal. The original and the smoothed signals are each convolved with the zero crossing wavelet. This produces zero crossings at 4 different scales, separated by powers of 2 in resolution. Four levels of decomposition is usually sufficient, as shown in figure 2. This is similar to the pyramidal decomposition in [11].

The points where each decomposition crosses the abscissa are called zero crossings. The portion of the function between two adjacent zero crossings is called an arch. The energy of an arch is the integral of the square of the function between two adjacent zero crossings. See figure 2.

2.1 Energy Zero Crossing Representation

To reduce the storage requirements of the data, we convert the dyadic representation to a zero crossing representation, which contains only the positions of the zero crossings and their associated energies. The result is called the Energy Zero Crossing (EZC) representation. The EZC representation consists of two linked lists, one containing the energy (as length and square of the dyadic decomposition's amplitude within the arch) and the other containing the positions of the zero crossings (see figure 3). The EZC representation is displayed in figure 4 on page 8. Using an iterative algorithm, we can reconstruct the original signal from this compressed representation with minimal loss of information [9]. An EZC representation contains information which came from only one scale/orientation pair.

2.2 Multiorientation Multiresolution Decomposition

The above one dimensional analysis has been generalized into two dimensions using multiorientation multiresolution wavelets. We decompose an image in different orientations, compute the zero crossings for a particular orientation, and reconstruct the image with very little loss of information.

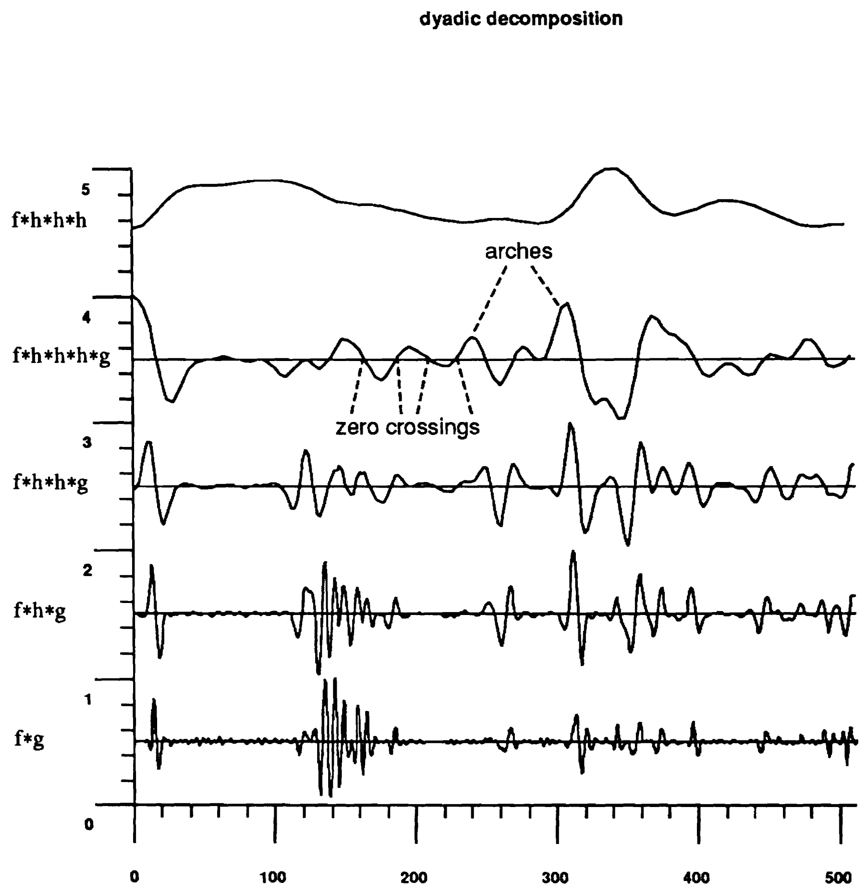

Figure 2: *Second order dyadic decomposition*

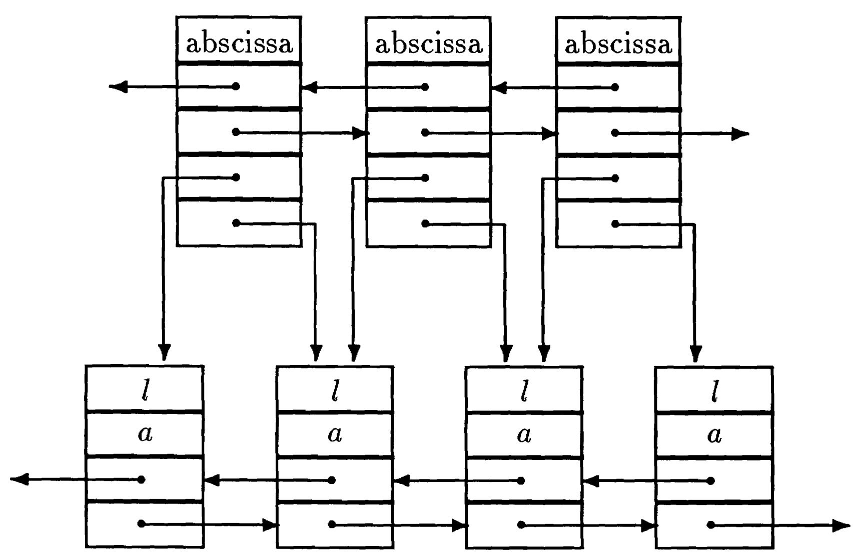

Figure 3: *Energy-zero-crossings structure.*

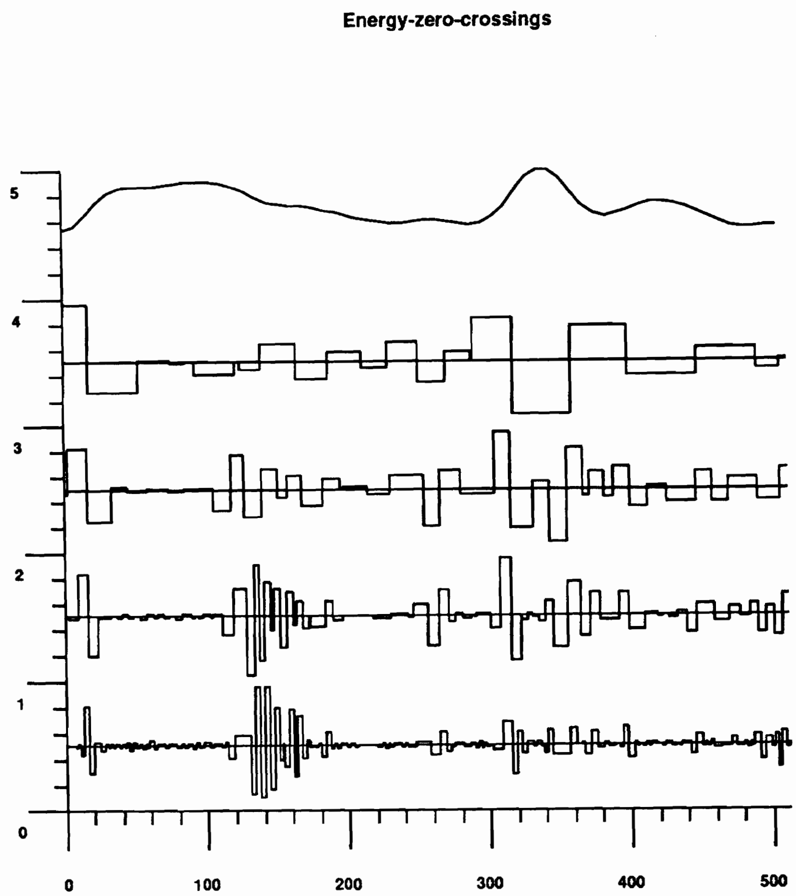

Figure 4: *Zero Crossing Representation.*

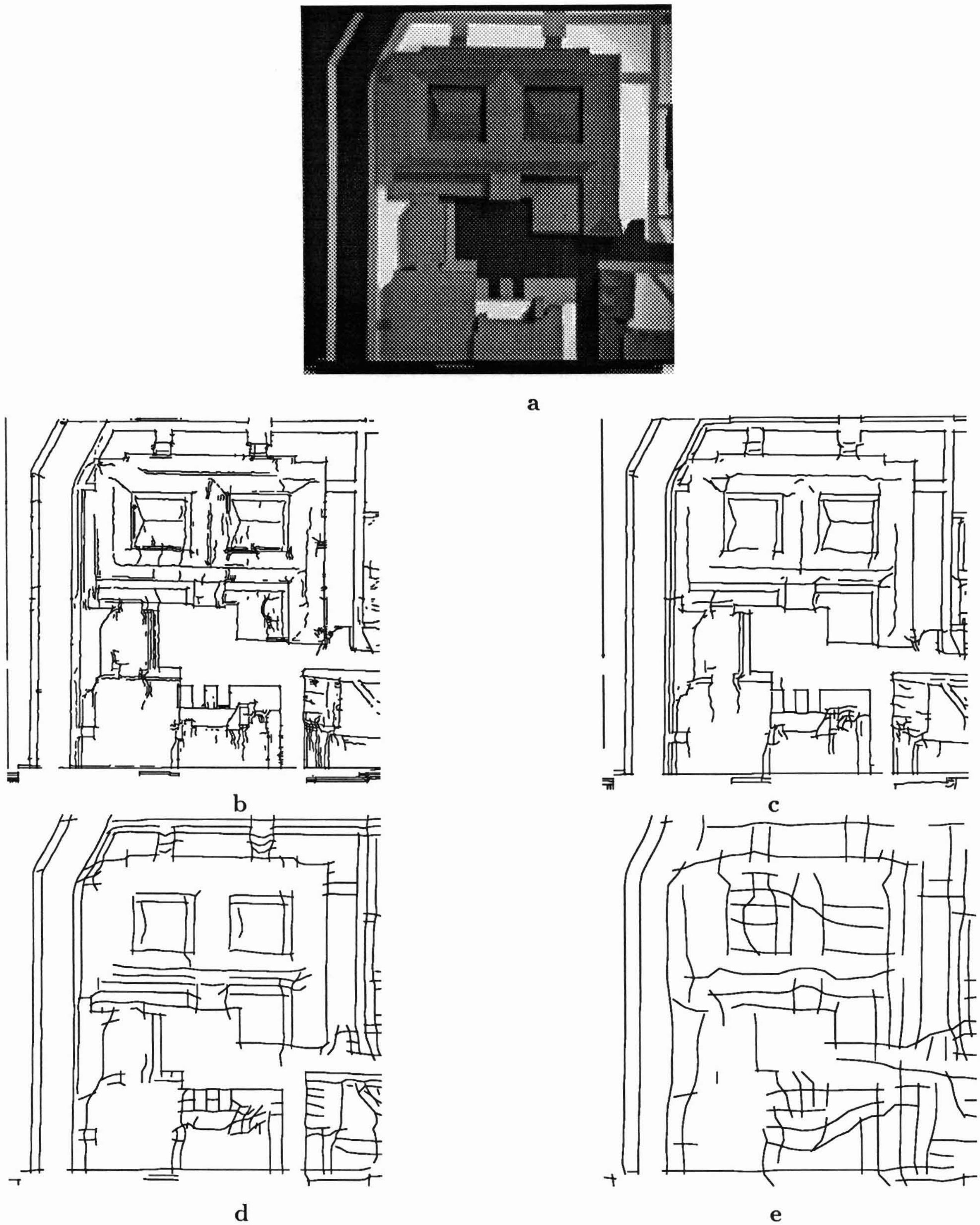

Figure 5: Edge Detection of the Penn Campus at varying resolutions:
a: Original, **b**: Resolution 2^j, **c**: Resolution 2^{j-1}, **d**: Resolution 2^{j-2}, **e**: Resolution 2^{j-3}

Using the energy zero crossing representation, we implemented an edge detection algorithm at one orientation and at one resolution. The results of this algorithm are shown in figure 5 (p. 9).

3 Zero Crossing Analysis in One Dimension

In figure 4, zero crossing locations are shown for several different scales. Intensity transitions in the image cause zero crossings at different resolutions, some only at coarse resolutions, some at fine resolutions and others all resolutions. Our hypothesis is that the edges present at fine resolutions only constitute texture, edges present at coarse resolutions only are shading, and edges present at all resolutions are borders edges. Lu and Jain [12] have shown that an isolated edge curve has a corresponding zero crossing curve at every scale using a Laplacian of Gaussian edge detector. Thus an isolated edge curve would be a border in our analysis. As edges approach other edges, the edges interfere with one another. For example, for the wavelet we use, edges must be 10 pixels apart to produce a response at all 4 resolutions, 5 pixels apart for the finest 3 resolutions, 2 pixels apart for the finest 2 resolutions, and 1 pixel apart for the finest resolution.

3.1 Matching Zero Crossings through Scale Space

We begin the analysis of the zero crossings by matching zero crossings through scale space. This links zero crossings in different scales which were produced by a single intensity transition in the image. We implemented a coarse to fine algorithm which matches zero crossings through scale space. The criteria for matching zero crossings are defined by the signal to noise characteristics of the image acquisition system and by the widths associated with the individual scales. In other words, the signal must be strong enough, and the distance that the zero crossings can move is limited by the half-width of the scale.

We developed a vector representation based on the coarse to fine matching results. The vector is an n by 1 vector where n is the number of levels of decomposition, 4 in this case. The vectors are then used for analysis on the nature of the edges.

In figure 6, a signal is superimposed over its dyadic decomposition. Though visual texture is difficult to define in 1 dimension, it is clear that small intensity changes in the the signal are smoothed, but that high amplitude changes are preserved.

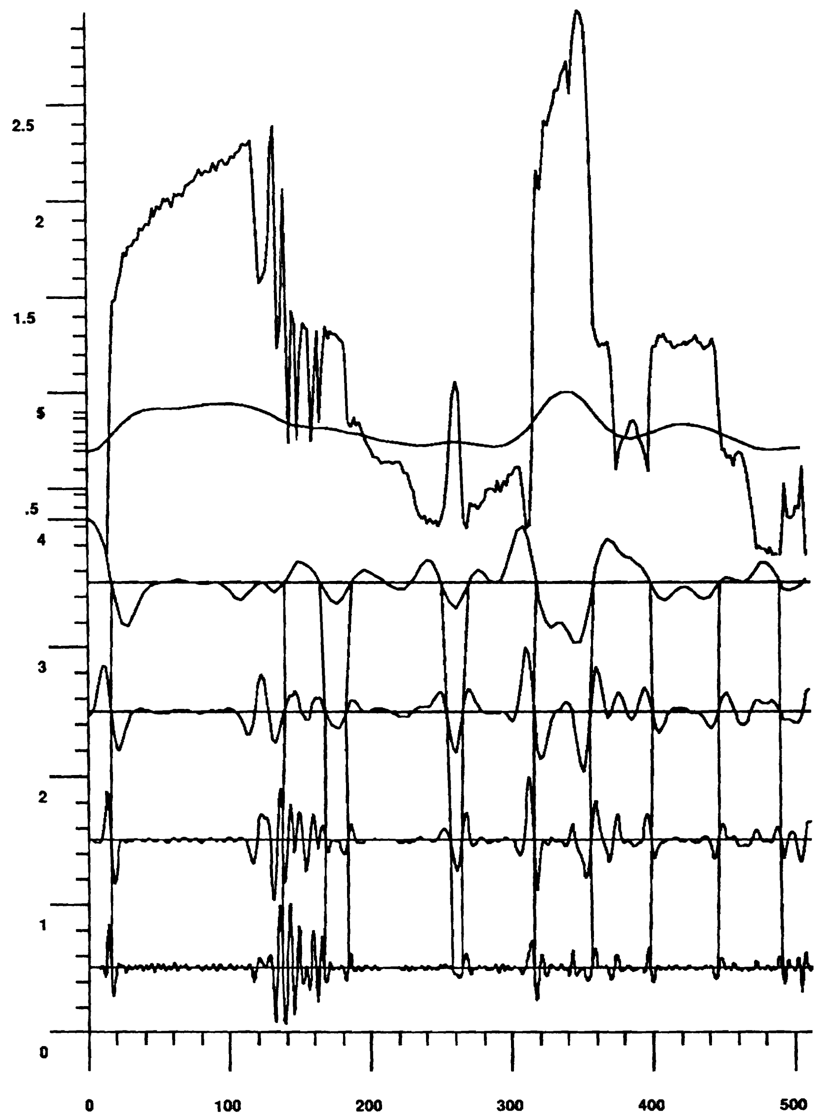

Figure 6: Original juxtaposed with its dyadic decomposition.

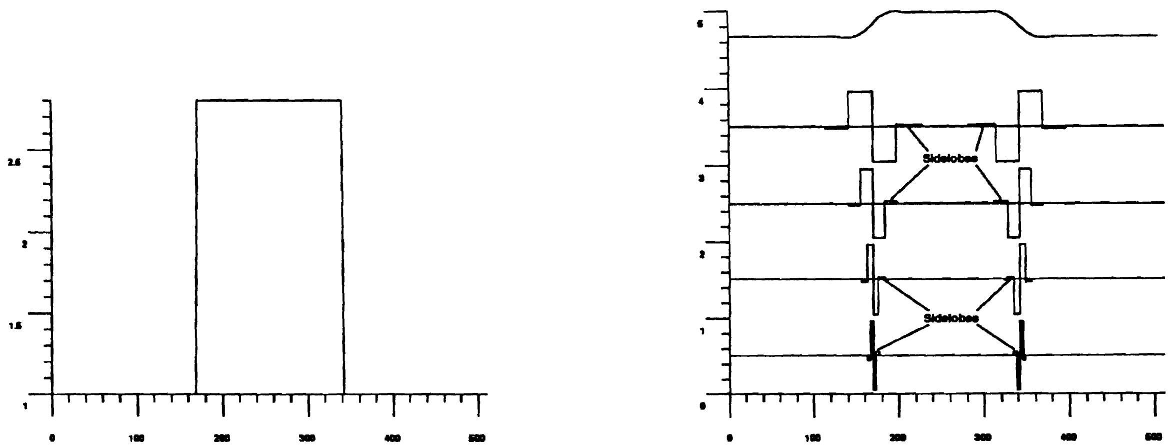

Figure 7: *Step function and its zero crossing representation.*

3.2 Sidelobes

One of the ambiguities of the zero crossing representation is that strong edges produce spurious zero crossings on both sides of the primary zero crossing response, and these are included in the zero crossing list. These sidelobes are a direct result of the shape of the wavelet. They usually exist next to border zero crossings which are zero crossings which line up through scale space. It is important to label the sidelobe zero crossings, so we do not accidentally remove or modify them, because they are part of the image reconstruction after texture removal. Figure 7 points out where sidelobes are in a zero crossing representation.

We implemented a sidelobe labeling algorithm. Because the sidelobes only occur near strong edges, we start by traversing the zero crossings which are matched through scale space. We take the nearest zero crossings to the left and the right of the matched zero crossing. As shown in figure 4, adjacent zero crossings share a common energy. The energies to the left and right of the traversed zero crossing are large, and they correspond to an edge. The next zero crossings, if they are sidelobes, share a large energy with the traversed zero crossing, and have a small energy on the opposite side. That small energy is classified as a sidelobe. The threshold of sidelobe energy classification is related to the shape of the wavelet itself.

Rather than finding and labeling sidelobes, another possible approach is to use a different filter shape, such as a Laplacian of Gaussian, which only contains two zero crossings. This is expected to reduce the accuracy of the reconstruction (using only 4 scales), but is also expected to reduce the processing time. We plan further research into the topic of comparative analysis of different wavelets and other filters, for this purpose.

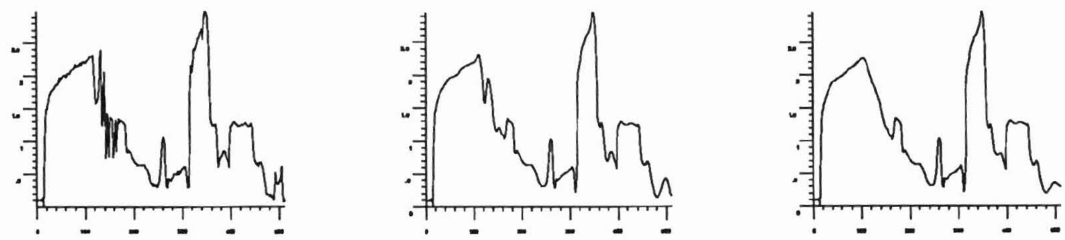

Figure 8: *Original signal; 3rd, 4th deleted; 2nd, 3rd and 4th deleted*

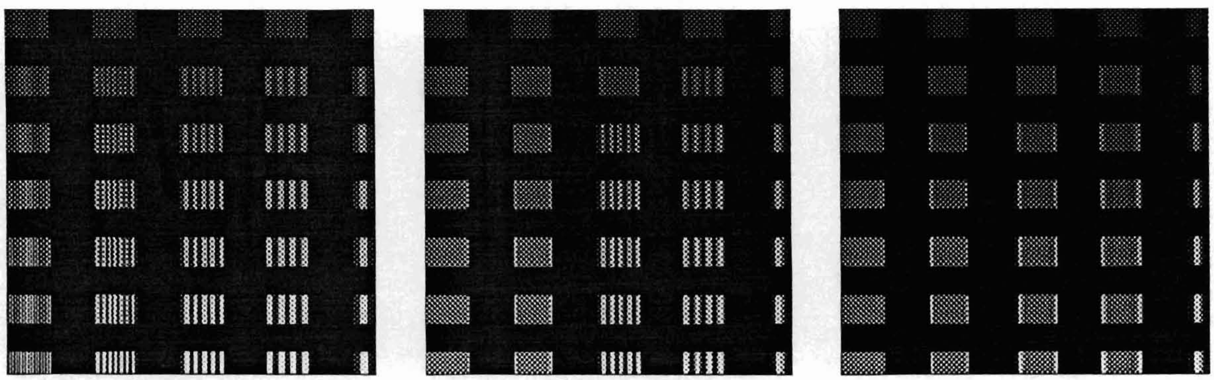

Figure 9: *Original image; 3rd, 4th deleted; 2nd, 3rd and 4th deleted*

4 Texture and Border Analysis

4.1 Texture Deletion

To delete the texture, we remove zero crossings and their energies from the list of zero crossings, then reconstruct the image. There are two options: we can delete zero crossings which are present only at the finest two resolutions or zero crossings which are present only at the finest three resolutions. We have experimented with both of the options. As expected, more smoothing occurs when zero crossings present at three resolutions are removed. The decision of how much texture to remove can be based on the number of regions that a high level segmentation process can handle.

4.2 Two Dimensional Texture Removal

To extend the texture removal to two dimensions, we use the one dimensional process along all scan lines, rotate the resulting reconstructed image by 90 degrees, and remove texture from that. Since edge spacing is an issue in texture removal, we expect that some texture with a diagonal orientation will be missed. However, preliminary results indicate that these errors are not important enough to add diagonal orientations and double the processing time.

As stated before, the the level of texture deletion is determined by the number of resolutions from which zero crossings are removed. Figure 9 shows a comparison of images with the 2 finest resolutions deleted versus the 3 finest resolutions deleted, with a total representation of 4 spatial resolutions.. In figure 10 (p. 15), texture at the lowest 3 levels have been removed from a closeup of a real image of the Philadelphia skyline. The figure also shows the texture removed in each orientation individually. At this scale, windows constitute texture, and the buildings are large enough to be treated as features. Figure 11 (p. 16) shows deleted texture at the lowest 3 levels of the real image of the Philadelphia skyline.

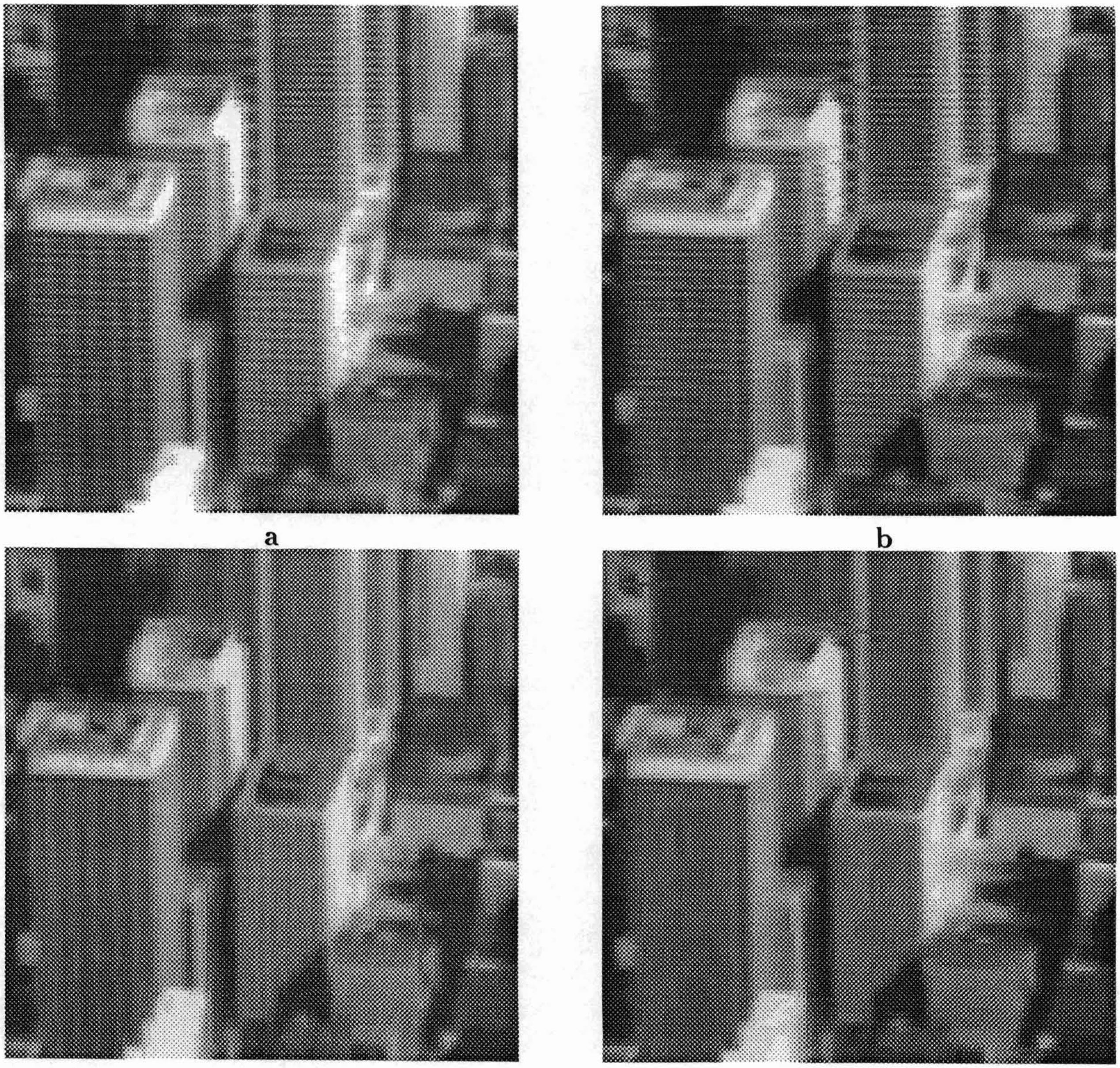

Figure 10: Texture Removal in Finest 3 Scales from Philadelphia Image.
Original(**a**), Vertical Texture Deleted (**b**), Horizontal Texture Deleted (**c**), Both Horizontal and Vertical Texture Deleted (**d**)

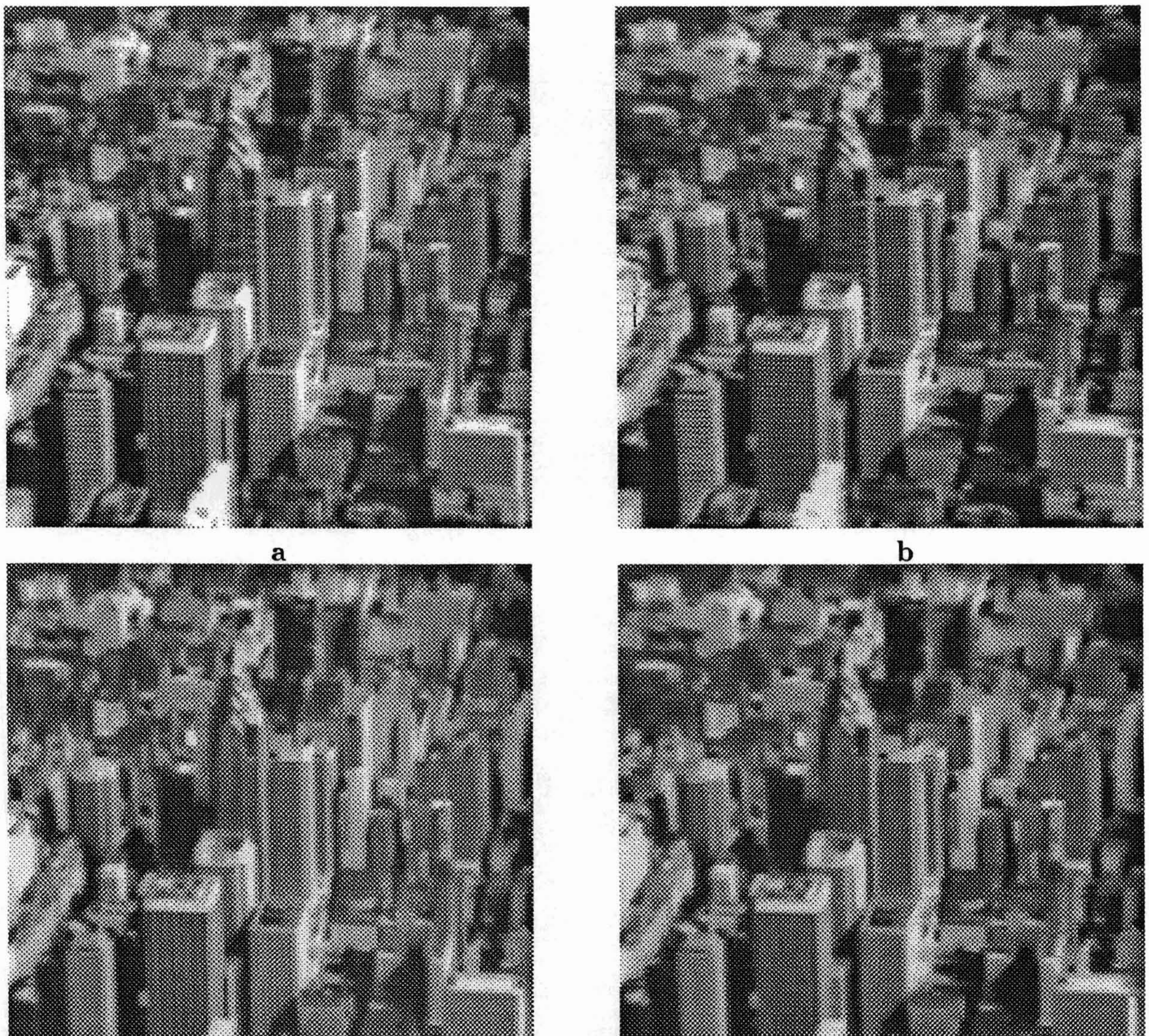

Figure 11: Texture Removal in Finest 3 Scales from Philadelphia Image.
Original(**a**), Vertical Texture Deleted (**b**), Horizontal Texture Deleted (**c**), Both Horizontal and Vertical Texture Deleted (**d**)

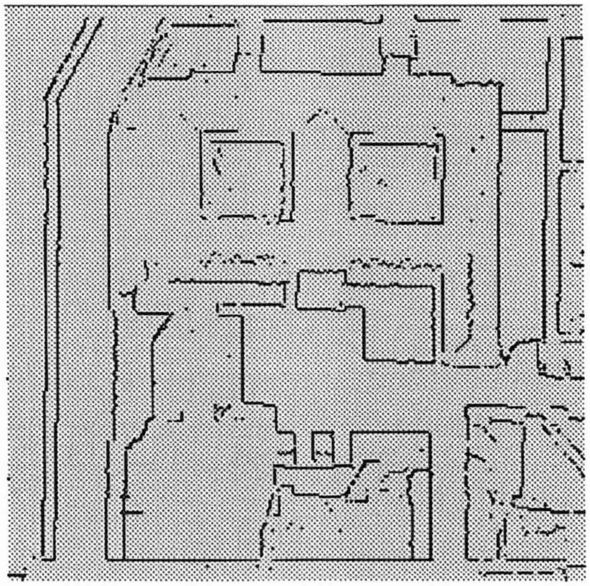

Figure 12: *Detected Borders from Image in figure 5a*

4.3 Border Edge Detection

To test the hypothesis that edges present at all spatial frequencies are borders, we decompose an image whose borders are well defined into the multiscale representation. The borders can be defined because the image consists of an architect's scale model of the University of Pennsylvania campus, and the borders are actual cardboard structures whose locations can be measured in the laboratory. Edge information which is not present at all frequencies are expected to be borders, as shown in figure 12, and the borders shown match the physical edges of the structures. The edge locations are determined by the finest scale, though the edge detection is done at all four scales. An extension of this work will be to connect border edge pixels to form closed boundaries, particularly when some of the edges are found on all scales and others on fewer scales.

5 Conclusion and Future Work

The hypothesis that multiscale edge detection can separate border edges from texture was confirmed visually by these experiments. The next step is to continue this work toward a complete segmentation by Active Perception, and then use the segmentation to guide robotic exploration of a scene. This will provide a physical confirmation that our definition of texture is appropriate to robotic exploration of a scene.

The physical limitations of the image acquisition system and the robots provide the limitations on the range of scales which we consider. Then we link edges through scale space, using the characteristics of these wavelets for guidance. The linked zero crossings are used to remove texture and preserve borders, then the scene can be reconstructed without texture.

References

[1] Jan J. Koenderink. The structure of images. *Biological Cybernetics*, 50:363–370, 1984.

[2] Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):996–1005, August 1988.

[3] R. Paul. *Robot Manipulators.* MIT Press, 1983.

[4] Demetri Terzopoulos,Andrew Witkin and Michael Kass. Signal matching through scale space. *International Journal of Computer Vision*, 1(2):pp.714–719, 1987.

[5] Stephane G. Mallat. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on PAMI*, 11(7):pp. 674–693, July 1989.

[6] Petros Maragos. Pattern spectrum and multiscale shape representation. *IEEE Transactions on PAMI*, 11(7):pp.701–716, July 1989.

[7] Ming-Hua Chen and Ping-Fan Yan. A multiscaling approach based on morphological filtering. *IEEE Transactions on PAMI*, 11(7):pp.694–700, July 1989.

[8] A. Grossman and J. Morlet. Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM J. Math.*, 15:p. 723–736, 1984.

[9] Stephane Mallat. *Multiresolution Representations and Wavelets.* PhD thesis, University of Pennsylvania, Department of Computer and Information Science, School of Engineering and Applied Science, August 1988.

[10] Nicolas Treil,Stephane Mallat and Ruenza Bajcsy. *Image Wavelet Decomposition and Applications.* Technical Report MS-CIS-89-22, University of Pennsylvania, Department of Computer and Information Science, School of Engineering and Applied Science, May 1989.

[11] Peter J. Burt. Smart sensing within a pyramid vision machine. *Proceedings of the IEEE*, 76(8):pp.1006–1015, 1988.

[12] Yi Lu and Ramesh Jain. Behavior of edges in scale space. *IEEE PAMI*, 11(4):p. 337ff, April 1989.

[13] Gilbert Strang. *Wavelets and Dilation Equations: A Brief Introduction.*