9-1-2000

Multiresolution gradient-based edge detection in noisy images using wavelet domain filters

Yunwoo Lee

Samuel Peter Kozaitis

Multiresolution gradient-based edge detection in noisy images using wavelet domain filters

Yunwoo Lee
Samuel P. Kozaitis, MEMBER SPIE
Florida Institute of Technology
Division of Electrical and Computer Science and Engineering
150 West University Blvd.
Melbourne, Florida 32901
E-mail: kozaitis@zach.fit.edu

Abstract. We detected edges in noisy images using multiresolution analysis with the wavelet transform. Products of wavelet coefficients at several scales were used to identify and locate edges. We found that it was important to consider the changes in edge position at different scales to detect edges in noisy imagery. We analyzed one-dimensional edges and compared the results of our approach with the first derivative of the signal. In addition, we compared the results of noisy images with another wavelet-based edge detection method. Our results led to improved edge detection in noisy images. [S0091-3286(00)01709-8]

Subject terms: edge detection, multiresolution, wavelet transform.

Paper 990226 received June 8, 1999; revised manuscript received Mar. 16, 2000; accepted for publication Mar. 17, 2000.

1 Introduction

Edges in noisy images are often difficult to detect because edge detection algorithms may be sensitive to noise. Most edge detection algorithms can perform well on clear images, but in noisy images may miss images or produce false edges caused by discontinuities in gray levels due to noise. For these reasons, some approaches attempt to find edges in smoothed images rather than the original ones to reduce the effect of noise. For example, a combination of a Gaussian low-pass filter and a Laplacian edge detection filter has been used to detect edges.[1] Canny discovered a useful approach that used a Gaussian filter to smooth images before edge detection.[2] The Canny edge detector generally performs better than gradient methods, but the effectiveness of Canny's approach is different for different choices of the thresholding and the standard deviation of the Gaussian filter. Hence, finding the proper values of these parameters is important in detecting real edges. In an optimal approach with respect to the signal-to-noise ratio, edges were detected by a curve-segment-based functional guided by the zero-crossing contours of the Laplacian-of-Gaussian approach.[3] These methods have been shown to be useful, but in the presence of noise may detect false edges or miss edges by excessive smoothing.

Nonlinear filters have been investigated because of their ability to suppress noise and preserve signal features such as edges. For example, filters have been used as nonlinear edge enhancers functioning as prefilters for edge detectors; filters were able to convert smooth edges to step edges and suppress noise simultaneously.[4] A classification of various approaches to nonlinear filtering has been presented, resulting in three types of estimators according to the process of the filter.[5] Generally, the filtering window size and the parameters of the filter must be set manually according to the input image. Thus, comparison of edge detection results with different values of filtering window size and parameters are necessary to get pertinent values for the best edge detection.

Mallat showed that a multiscale approach using a wavelet transform could be equivalent to a multiscale Canny edge detector.[6] Multiscale approaches included those that represented edge positions by local maxima in the absolute-value distribution of the wavelet coefficients.[7–9] Information on local extrema and the modulus of the wavelet transform has been used for multiscale corner detection.[10] In addition, the M-band wavelet transform has been used for multidirectional and multiscale edge detection by decomposing an image into $M \times M$ channels.[11] The channels are combined to produce zero crossing at locations of edges corresponding to different directions and resolutions. These and other wavelet-based approaches have often shown promise by using the correlation between different scales of an image.[12–19]

The correlation of noisy wavelet coefficients between different scales has been shown to detect intensity discontinuities better than traditional methods. A spatially selective noise filtration technique based on the product of the wavelet transform at adjacent scales was shown to detect edges.[20] Although edges could be detected, edge locations may be ambiguous or shift at multiple scales of the wavelet transform. Therefore, a more robust technique is needed for more general edge detection in noisy images.

In our approach, we also used the product of noisy wavelet coefficients on different scales to reduce the effect of noise. However, we also considered shifting edge locations through multiple scales for robust edge detection in the presence of noise. In the next section we describe our approach. Then, we show the results of experiments with one-dimensional (1-D) signals, and compare our result with that for the first derivative of the signal. We also compared the results of noisy images to another wavelet-based edge detection method. Finally, we discuss our conclusions.

2 Wavelet Transform

2.1 *Basis Functions*

Like the Fourier transform (FT), the wavelet transform (WT) describes a function with basis functions. However, the basis functions of the WT, scaling functions and wavelets, are often more complicated than the basis functions of the FT, sines and cosines. Unlike the FT, the basis functions in the WT are localized in both the input and the wavelet domain. Like the FT, the WT is a linear operation that is invertible and can be made orthogonal.

The general idea behind the wavelet transform is to represent any arbitrary function as a superposition of wavelets based on a mother wavelet. Scaled and shifted versions of the mother wavelet can be summed to represent an arbitrary function, as in the FT scaled and shifted versions of sine and cosine functions can be summed to represent an arbitrary function.

The WT maps a signal in the space domain into a scale-translation domain using scaled and translated versions of a mother wavelet. The continuous 1-D WT of the function $f(x)$ with respect to a mother wavelet $w_{jk}(x)$ is

$$b_{jk}=\int f(x)w_{jk}(x)\,\mathrm{d}x, \tag{1}$$

where b_{jk} are the wavelet coefficients, and j and k indicate the scale and shift of the wavelet. In digital implementations, a dyadic family of wavelets is often used, so the wavelet is written as $w_{jk}(x)=2^{-j/2}w_{jk}(2^{-j}x-k)$, where $j=1,\ 2,\ \ldots$. In this case, the wavelet is dilated by a factor of two at each increasing scale. In addition, the amplitude of the wavelet increases by a factor of $\sqrt{2}$ at each increasing scale, so scaled wavelets have the same energy as the original mother wavelet.

In the WT, the input function $f(x)$ is compared with a wavelet $w_{jk}(x)$ through a correlation or projection. A wavelet-domain coefficient is computed for each particular scale and shift value; it is equal to the correlation coefficient between $f(x)$ and $w_{jk}(x)$. These coefficients determine the WT. In other words, a function is approximated by a weighted sum of the scaled and shifted versions of the mother wavelet. When they are added together, the original signal is obtained. In addition, the WT is linear and superposition holds.

The inverse WT reconstructs the original function by summing weighted, scaled, and shifted versions of the mother wavelet. The weights are the wavelet coefficients b_{jk}. The inverse WT sums over the 2-D scale-translation space as

$$f(x)=\sum_{j,k} b_{jk}w_{jk}(x). \tag{2}$$

2.2 *Multiresolution*

The wavelet transform actually contains a dual basis of wavelets and scaling functions represented by $w_{jk}(x)$ and $\phi_{jk}(x)$. At a given scale j, the scaling functions $\phi_{jk}(x)$ are a basis for a signal. The signal at scale j and the *details* at scale j combine into a multiresolution of the signal at the finer scale $j-1$:

$$\text{signal at level } j+\text{details at level } j = \text{signal at level } j-1. \tag{3}$$

The details come from the wavelet coefficients. Therefore, at each scale $j-1$, we have two bases for the signal, either the $\phi_{jk}(x)$'s at level $j-1$ or the $\phi_{jk}(x)$'s and $w_{jk}(x)$ at level j:

$$\sum_k a_{j-1k}(x)\phi_{j-1k}(x)=\sum_k a_{jk}(x)\phi_{jk}(x) + \sum_k b_{jk}(x)w_{jk}(x), \tag{4}$$

where a_{jk} are scaling function coefficients,

$$a_{jk}=\int f(x)\phi_{jk}(x)\,\mathrm{d}x. \tag{5}$$

The third term in Eq. (4) is the detail signal at scale j and contains the difference of information between two successive approximations at scales j and $j-1$. The multiresolution approximation is completely characterized by the scaling function, and it is possible to choose scaling functions with good localization properties in both the frequency and input domains.

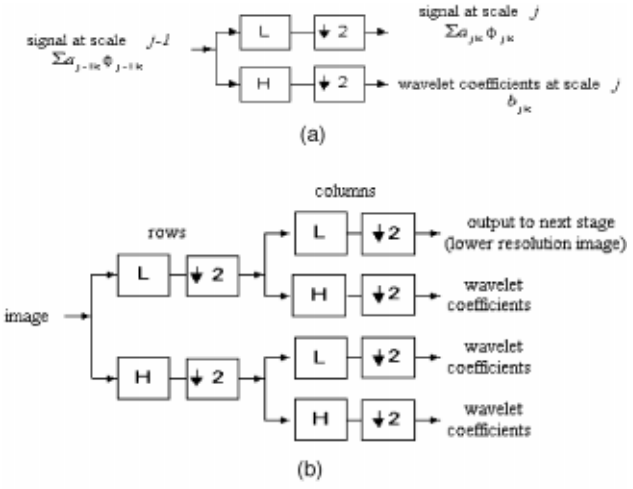

Fig. 1 Filter-bank implementation of one level of the wavelet transform: (a) one dimension, (b) two dimensions.

2.3 *Implementation*

The wavelet transform is usually implemented with a series of identical filter banks; one filter bank is used for each scale of the WT. The most popular configuration is a filter bank that consists of a low-pass and a high-pass filter. A block diagram of the filter bank is shown in Fig. 1(a). In many cases the outputs of the filters are downsampled.

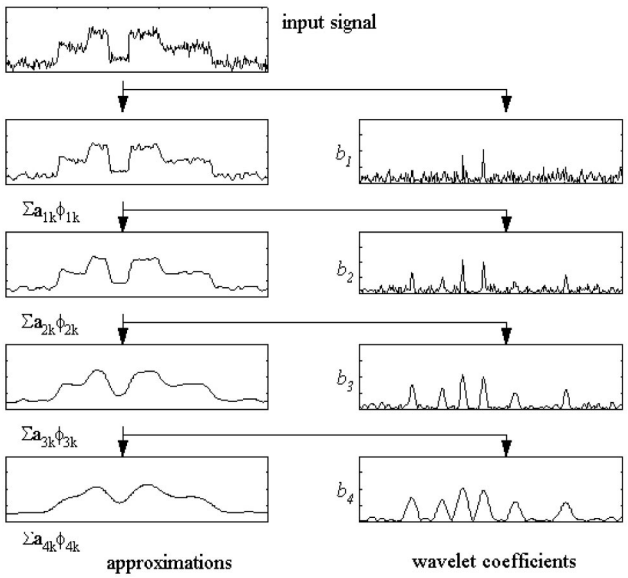

Fig. 2 Decomposition of input signal into four approximations with associated wavelet coefficients.

In the case of the two-dimensional separable WT, the rows (columns) are processed as in the 1-D case, followed by the columns (rows), as shown in Fig. 1(b). In this case an image is decomposed to an approximation and three detail images. Because the WT has fewer coefficients at decreased resolutions, the wavelet coefficients, which contain the edge information, could be distorted by downsampling. Therefore, we did not use downsampling in our edge-detection method described in the next section.

We show a one-dimensional decomposition of a signal in Fig. 2. Below the input signal are four approximations, $\Sigma_k a_{jk}(x)\phi_{jk}(x)$ using $j=1,2,3,4$. At the same scale we also show the nondownsampled wavelet coefficients b_{jk}.

3 Edge Detection Method

The product of noisy wavelet coefficients between different scales has been shown to detect intensity discontinuities better than traditional methods.[20] For example, to detect edges using scales $j=1$ and 2, a product C_{12} is formed between a signal's nondownsampled wavelet transform coefficients of the two scales, b_{1k} and b_{2k}. Then, the energy of the product C_{12} is rescaled so that it has the same energy as b_{1k}:

$$\sum_k (b_{1k})^2=\sum_k (C_{12})^2=\sum_k (b_{1k}b_{2k})^2; \tag{6}$$

we drop the subscript k in the following discussions to simplify the notation. An edge is identified at a position where $|C_{12}|>|b_1|$. The edge locations are saved in a binary spatial mask, and the values of C_{12} and b_1 are set to zero at the locations of the edges. The energy of C_{12} is again rescaled to that of b_1 as in Eq. (6), and a new product is formed between C_{12} and b_1. The energy of the new product is rescaled to that of b_1 as before, edges are identified, their locations saved, and another product formed between the new C_{12} and b_1. This process is repeated until the energy in b_1 reaches some reference noise power in b_1.

3.1 *Product Method*

The simplest way to examine the consistency of wavelet coefficients of different scales is by forming their product between scales.[20] When only considering two scales, we can multiply the first scale ($j=1$) of wavelet coefficients, b_1, by the second scale ($j=2$), b_2. If we are considering an edge across n specific scales, we can multiply wavelet coefficients in each pair of adjacent scales together, so that there will be $n-1$ signal multiplications. However, in the most general case the best scales to use to detect edges may be unknown. Therefore, for n scales there are $\binom{N}{2}$ possible scale combinations to be considered. Figure 3 shows a noisy step signal where four levels of the wavelet transform are considered for edge detection. First, the wavelet transform without downsampling is calculated from the input signal, where b_1, b_2, b_3, and b_4 indicate the wavelet coefficients for levels 1 to 4. The products between the four levels were found, and edges identified by the iterative method described previously in this section. The edge location vector is a binary vector and contains only edge location information. Then, the product of b_1 and the edge location vector is formed to produce edge information. This method works best for step edges, because it is clear at every resolution where the edge is located. However, with other types of edges, the position of the edge may not be well defined or may appear to change at different resolutions. Therefore, forming the products of different scales may not indicate the consistency of an edge over multiple scales.

3.2 *Product with Shift Method*

Because edge locations may shift according to scale, the product method can be modified to form the product between shifted versions of the wavelet coefficients of a signal. For example, we may consider a 1-D example of the product between the first two scales of a signal allowing an edge to be shifted by one pixel in the right or left direction. There are three possible products: b_1b_2 (the product of wavelet coefficients at scales $j=1$ and 2), b_1b_{2L1}, and b_1b_{2R1}, where b_{2L1} and b_{2R1} refer to shifts of b_2 to the left and right by one location, respectively. To allow for the shift of an edge between scales, we used the maximum value of the three products at each location of b_1 as the result of the correlation between scales. For three specific scales, considering that an edge may shift by one pixel at each scale, we get nine different products. For n specific scales, allowing a one-pixel shift between scales for edges, 3^{n-1} signal multiplications are needed for a 1-D signal.

When considering edge positions that can shift by two pixels between scales, more signal multiplications are needed. For example, with edges that may shift up to two pixels between scales, we get five products, b_1b_2, b_1b_{2L2}, b_1b_{2L1}, b_1b_{2R1}, and b_1b_{2R2}, where b_{2L2} refers to shifting b_2 by two pixels to the left, and b_{2R2} refers to shifting b_2 by two pixels to the right. With three specific scales we must consider 25 signal multiplications; for n scales, 5^{n-1} signal multiplications are needed, allowing a two-pixel shift

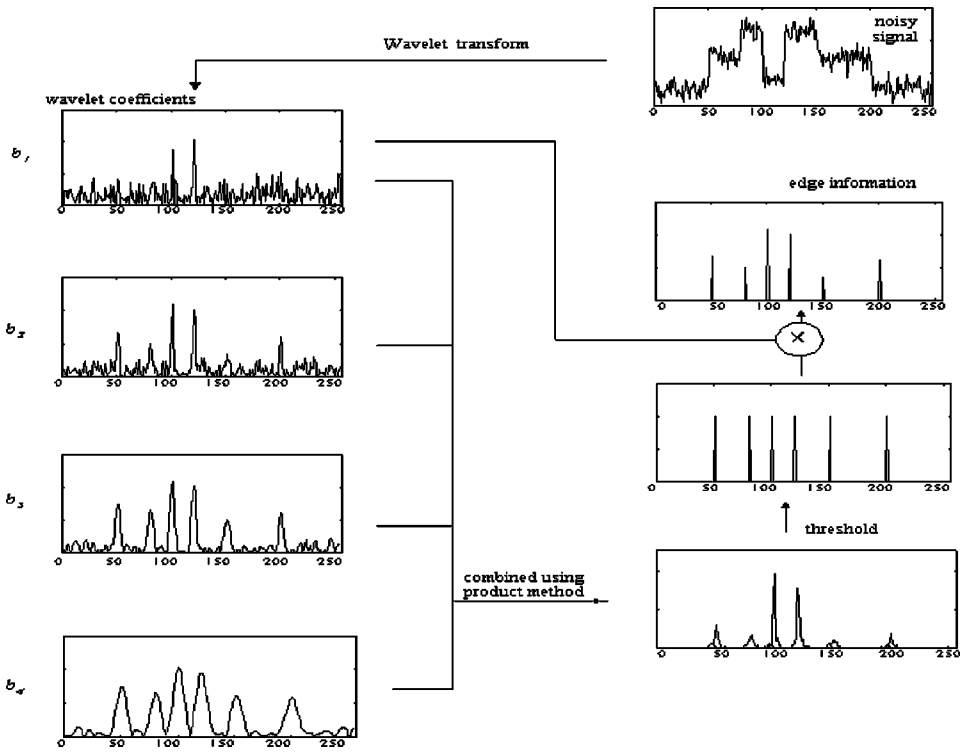

Fig. 3 Process of edge detection using the product method and four levels of the wavelet transform.

between scales for edges in a 1-D signal. For an edge location to shift a maximum of x locations, and considering the product of n specific scales, the product-with-shift method uses the product

$$C_{1n}=\max\{b_{1,k}b_{s,k+t}\}, \tag{7}$$

where $s=2,\ 3,\ \ldots,\ n,\ t=0,\ +1,\ +2,\ \ldots\ +x$, and $(2x+1)^{n-1}$ signal multiplications are needed to determine the edge locations. The data in C_{1n} are the result of the iterative process similar to C_{12} in the previous section. We found that this method works well with a variety of edges; however, the number of computations increases rapidly as the number of scales is increased.

3.3 *Alternative to Product-with-Shift Method*

We wanted the same effect as the product-with-shift method but without the computational expense. We approximated the product-with-shift method by initially finding the edges considering only two scales, then combining the result with wavelet coefficients from each additional scale, one scale at a time, using the product-with-shift method. For example, using wavelet coefficients from four scales b_1, b_2, b_3, and b_4, we used the product-with-shift method initially on the two largest scales, b_3 and b_4. Then, the resulting edge information was combined with b_2 using the product-with-shift method. The result of that step was then combined with b_1 using the product-with-shift method to determine the edge locations. For n specific scales, the number of signal multiplications is $(2x+1)(n-1)$, which can be significantly less than for the product-with-shift method as n increases.

3.4 *Image Edge Detection*

The individual wavelet transforms in the horizontal and vertical directions can be used to extend the 1-D edge detection algorithm to images. The block diagram of the image edge detection algorithm is shown in Fig. 4, using four scales or levels of the wavelet transform. First, the wavelet transform is applied to the input image in both the horizontal and vertical directions as shown in Fig. 4(a). Then, the vertical and horizontal edge images are produced separately as in the 1-D example using the alternative to product with shift, and labeled W_{cv} and W_{ch}, respectively in Fig. 4(b). The binary edge location images W_h and W_v in the horizontal and vertical directions are determined from W_{cv} and W_{ch} by thresholding them with some noise reference value as indicated in Fig. 4(c). Directional edge images are found by forming the product of the edge location images and the wavelet coefficients from the smallest scale b_{1h} and b_{1v} as indicated in Fig. 4(d). Finally, the modulus of the sum of these directional edge images,

$$M(x,y)=(|W_h b_{1h}|^2+|W_v b_{1v}|^2)^{1/2}, \tag{8}$$

represents the edge information in Fig. 4(e). Figure 5 shows an example of the edge detection algorithm in Fig. 4 using images instead of block diagrams.

4 Experiments and Analysis in 1-D

We implemented the edge detection algorithm with Mallat's wavelet, which corresponds to a first derivative. We considered noisy 1-D signals and compared the performance of the edge detection algorithm with the first derivative of noiseless signals to determine how well the method

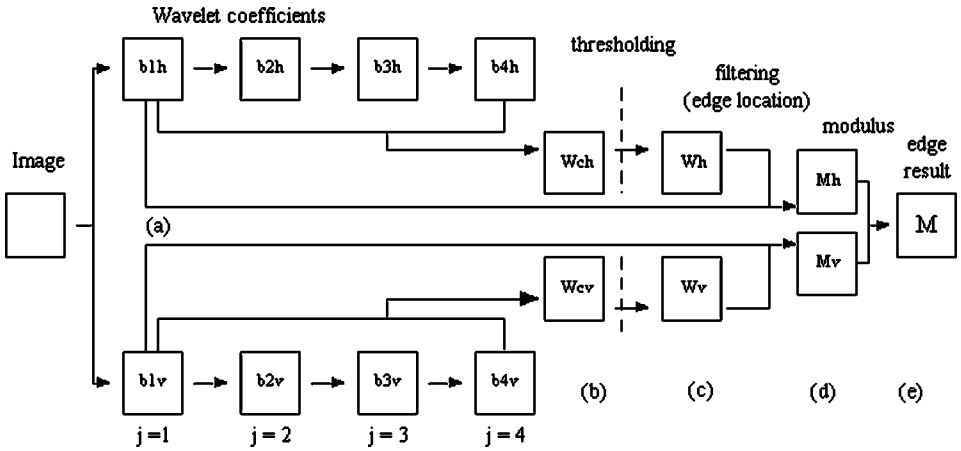

Fig. 4 Block diagram of the image edge detection algorithm: (a) wavelet transforms in horizontal and vertical directions for $j=1,2,3,4$; (b) results of alternative method; (c) binary edge location; (d) edge images formed by product of binary edge locations and wavelet coefficients at smallest scale; (e) result (modulus of the sum of directional edge images).

rejected noise while detecting edges. We considered the two edges shown in Fig. 6, which we refer to as a step edge and a rounded edge; their first derivatives are also shown. We found the mean squared error (MSE) between the edge function produced by the alternative method from a noisy edge and the first derivative of the edge function, for signal-to-noise ratios (SNRs) of 1 to 50. For each value of the MSE at a particular SNR we used 300 independent noise signals.

We found the MSE between the first derivative of the noiseless step edge and the noisy edge processed with the edge detection algorithm as a function of SNR using sev-

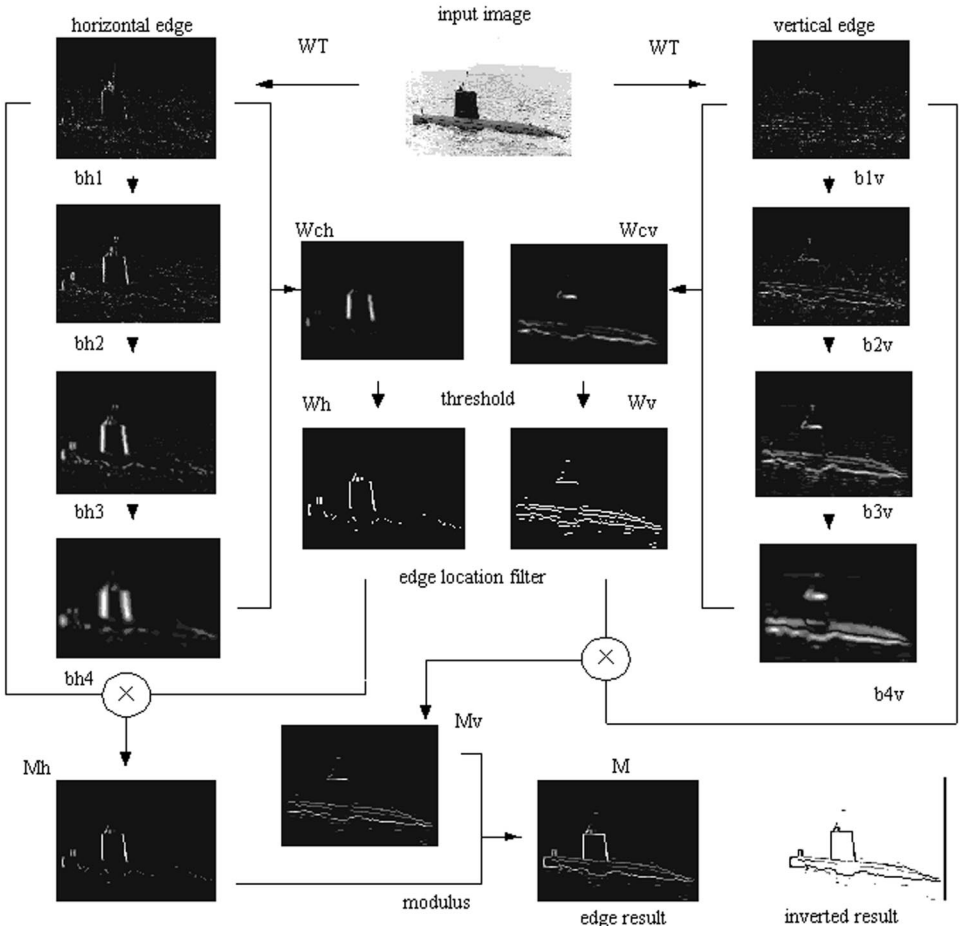

Fig. 5 Same steps as Fig. 4, but using images.

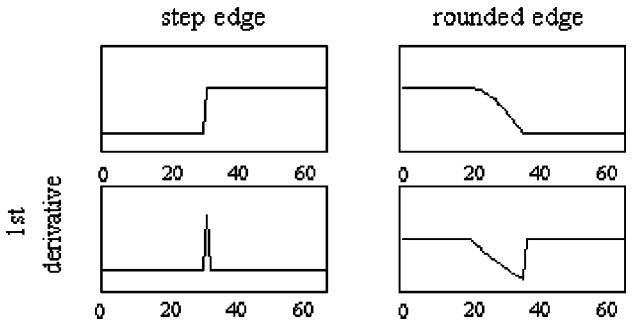

Fig. 6 Sample step and rounded edges and their derivatives used for 1-D analysis.

eral different combinations of scales, which are shown in Fig. 7. In Fig. 7, corr 1*2*3 indicates the correlation of the three smallest wavelet scales, b_1, b_2, and b_3, and similarly for other scales. Note that the correlation here and in Ref. 20 refers to a product between scales rather than a correlation operator. The dotted curves represent the first derivative of the noiseless edge, and the solid curves represent the alternative method. In general, the results show that increasing the number of levels in each algorithm lowered the MSE by a small amount. In addition, eliminating the lowest wavelet scales generally made the results less noisy at the expense of increasing the MSE. At almost all data points, the results of the product and alternative methods were similar, and the product method had a lower MSE than the alternative method.

Using the rounded edge, we found similar results: the MSE generally decreased as the number of scales used increased, as shown in Fig. 8. As before, we found that the

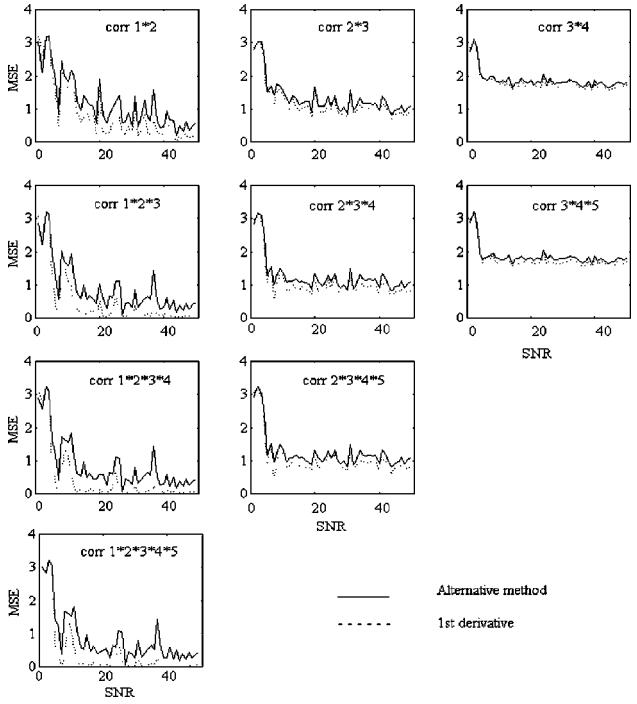

Fig. 7 MSE between a noisy step edge processed by the alternative method and the first derivative of a noiseless step edge, as a function of the SNR.

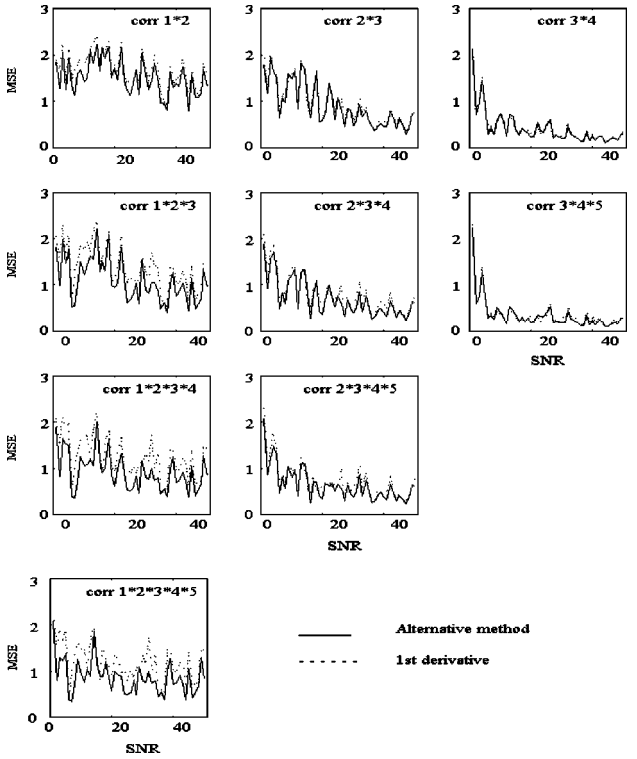

Fig. 8 MSE between a noisy rounded edge processed by the alternative method and the first derivative of a noiseless rounded edge, as a function of the SNR.

results became less noisy as the smaller scales were eliminated. In contrast to the previous results, we found that as the smaller scales were eleiminated, the MSE generally decreased. In addition, the alternative method almost always had a lower MSE than the product method.

5 Experiments on Images

We applied our approach to images and examined the performance with different combinations of scales. In addition, we compared the results with those of Mallat's method. In Fig. 9 we show a 256×256-pixel gray-scale image without noise and with SNRs of 50 and 10. We also show in the same figure the results (inverted) using several different combinations of scales in the edge detection algorithm with the alternative method. In this example, it can be seen that different combinations of scales can be used to detect edges. However, as the SNR decreases, results that include the smallest scales appear to be noisy. When the SNR was 10, the combination of scales 3 and 4 seemed to give the best performance from a visual standpoint. These scales could also be used for higher SNRs; however, using the smaller scales gave more detail in the images. Using the smaller scales is possible at higher SNRs because they are less noisy.

We used four different 256×256 pixel images, shown in Figs. 10–13, for comparison with Mallat's method. The images along with their noisy versions are also shown in these figures. In each figure, the results of using Mallat's and our methods for scale 2 and scale 3 are shown. In each of Figs. 10–13, the first row shows a noiseless image, and the second and the third rows show the results of noisy

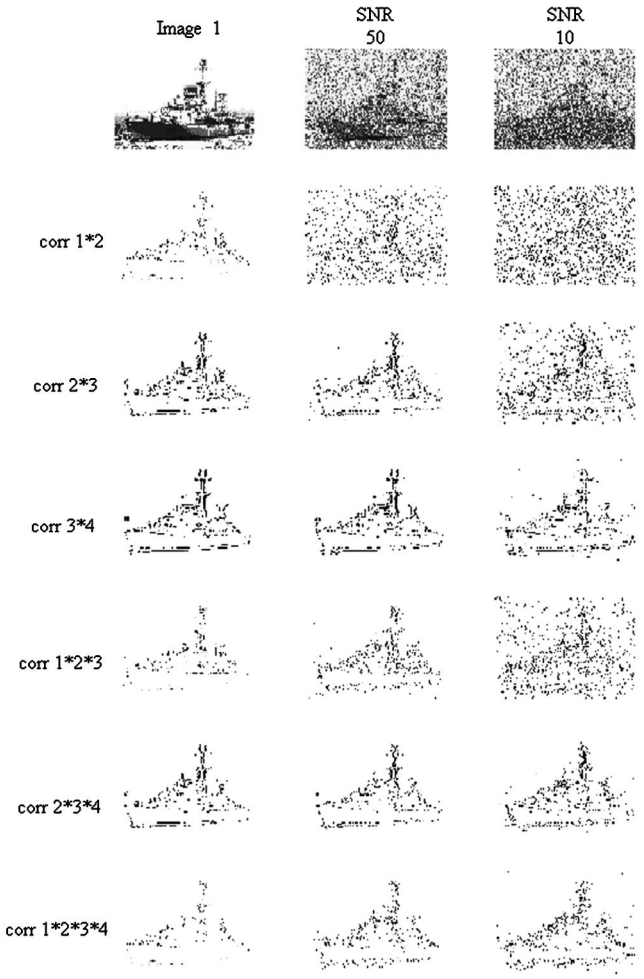

Fig. 9 Results of alternative method on image 1 using various scales for a few different SNRs.

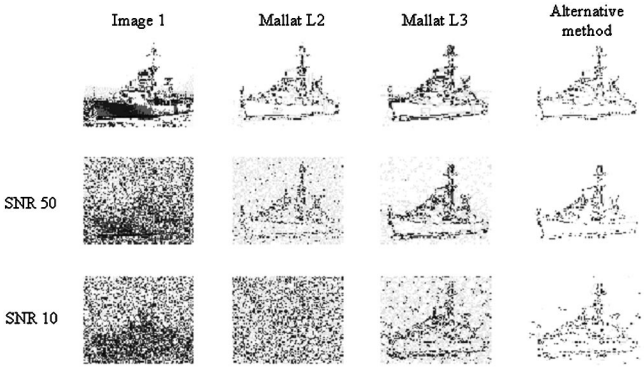

Fig. 10 Comparison of results on image 1 produced by Mallat's method for scales 2 and 3, and by the alternative method using the same scales.

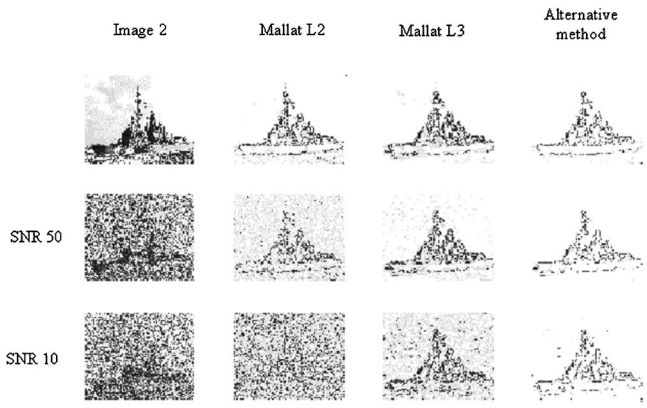

Fig. 11 Comparison of results on image 2 produced by Mallat's method for scales 2 and 3, and by the alternative method using the same scales.

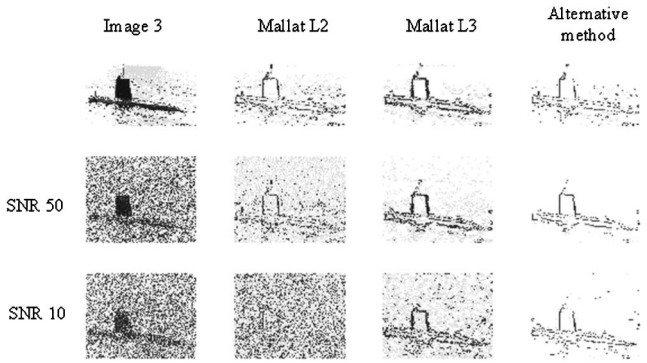

Fig. 12 Comparison of results on image 3 produced by Mallat's method for scales 2 and 3, and by the alternative method using the same scales.

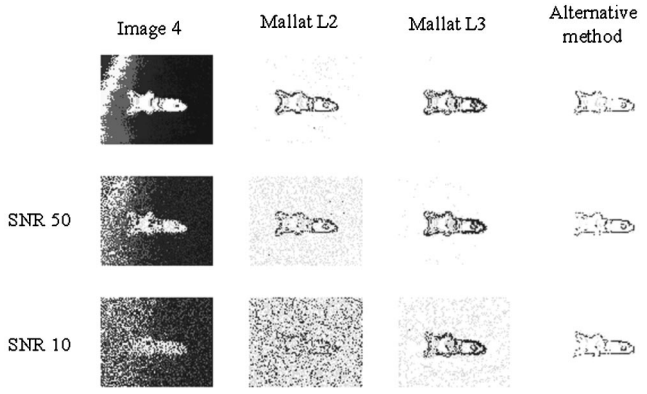

Fig. 13 Comparison of results on image 4 produced by Mallat's method for scales 2 and 3, and by the alternative method using the same scales.

images at SNRs of 50 and 10 respectively. The two middle columns show the results using Mallat's algorithm, and the rightmost column shows our results. Visually, the results of our method have less noise than those using Mallat's method in all cases.

6 Conclusion

We have shown that edges can be detected in noisy images using a multiresolution approach involving the products of wavelet coefficients of an image. Although the product of wavelet coefficients at multiple scales was adequate for the detection of step edges, it was important to follow the edge from scale to scale for improved edge detection. Allowing the edges to shift between scales increased the robustness of the approach while only minimally affecting the performance for step edges. Increasing the number of scales generally improved the results; however, using the smallest scales often decreased performance at low SNRs. Generally, our approach showed improved noise immunity with respect to edge detection. In addition, our results compared favorably with Mallat's wavelet-based method.

References

1. D. Marr and E. Hildreth, ''Theory of edge detection,'' *Proc. R. Soc. London, Ser. B* **207**, 87–217 (1980).
2. J. Canny, ''A computational approach to edge detection,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **8**(6), 679–698 (1986).
3. R. J. Qian and T. S. Huang, ''Optimal edge detection in two-dimensional images,'' *IEEE Trans. Image Process.* **5**(7), 1215–1220 (1996).
4. R. C. Hardie and C. G. Boncelet, ''Gradient-based edge detection using nonlinear edge enhancing prefilters,'' *IEEE Trans. Image Process.* **4**(11), 1572–1577 (1995).
5. Y.-S. Fong, C. A. Pomalaza-Raez, and X.-H. Wang, ''Comparison study of nonlinear on nonlinear filters in image processing,'' *Opt. Eng.* **28**(7), 749–760 (1989).
6. S. Mallat, ''Multifrequency channel decompositions of images and wavelet models,'' *IEEE Trans. Acoust., Speech, Signal Process.* **37**(12), 2091–2110 (1989).
7. S. Mallat, ''Zero-crossing of a wavelet transform,'' *IEEE Trans. Inf. Theory* **37**(4), 1019–1033 (1991).
8. S. Mallat and W. L. Hwang, ''Singularity detection and processing with wavelets,'' *IEEE Trans. Inf. Theory* **38**(2), 617–643 (1992).
9. S. Mallat and S. Zhong, ''Characterization of signals from multiscale edges,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(7), 710–732 (1992).
10. J.-S. Lee, Y.-N. Sun, and C.-H. Chen, ''Multiscale corner detection by using wavelet transform,'' *IEEE Trans. Image Process.* **4**(1), 100–104 (1995).
11. T. Aydin, Y. Yemez, E. Anarim, and B. Sankur, ''Multidirectional and multiscale edge detection via M-band wavelet transform,'' *IEEE Trans. Image Process.* **5**(9), 1370–1377 (1996).
12. J. W. Hsieh, M.-T. Ko, H.-Y. M. Liao, and K. C. Fan, ''A new wavelet-based edge detector via constrained optimization,'' *Image Vis. Comput.* **15**(7), 511–527 (1997).
13. J. I. Siddique and K. E. Barner, ''Wavelet-based multiresolution edge detection utilizing gray level edge maps,'' in *IEEE International Conf. on Image Processing*, Part 2, pp. 550–554 (1998).
14. J. Y. Xu and Q. X. Zhu, ''The usage of wavelet transform in image edge extracting,'' *Proc. SPIE* **3391**, 596–601 (1998).
15. A. Talukder and D. Casasent, ''Multiscale Gabor wavelet fusion for edge detection in microscopy images,'' *Proc. SPIE* **3391**, 336–341 (1998).
16. F. Truchetet, O. Laligant, E. Bourenanne, and J. Miteran, ''Frame of wavelets for edge detection,'' *Proc. SPIE* **2303**, 141–152 (1994).
17. H. M. Liao, M. Ko, J. Hsieh, and K. Fan, ''A new wavelet-based edge detector via constrained optimization,'' *Image Vis. Comput.* **15**(7), 511–527 (1997).
18. J. R. Beltran, J. Garcia-Lucia, and J. Navarro, ''Edge detection and classification using Mallat's wavelet,'' in *IEEE International Conf. on Acoustics, Speech, and Signal Processing*, Vol. 2, pp. 293–297 (1994).
19. J. Y. Xu and Q. X. Zhu, ''The usage of wavelet transform in image edge extracting,'' *Proc. SPIE* **3391**, 596–601 (1998).
20. Y. Xu, J. B. Weaver, D. M. Healy, Jr., and J. Lu, ''Wavelet transform domain filters: a spatially selective noise filtration technique,'' *IEEE Trans. Image Process.* **3**(6), 747–758 (1994).

Yunwoo Lee: Biography not available.

Samuel P. Kozaitis received his PhD in electrical engineering in 1986 from Wayne State University, where he was an assistant research professor. He has also been with General Motors Research Laboratories, was a research fellow at the Photonics Center of the Rome Laboratory from 1988 to 1997, spent two summers at the National Aeronautics and Space Administration Kennedy Space Center, and has performed consulting for government and industry. He is currently an associate professor in the Division of Electrical, Computer Science, and Engineering at the Florida Institute of Technology. His major research interests include optical pattern recognition, wavelet transforms, feature extraction, and higher-order statistics.