

Implementation of Oriented Filters for Arc Detection

Yuming Wen* and John P. Oakley

Division of Electrical Engineering

School of Engineering

University of Manchester, Manchester M13 9PL, UK

email j.oakley@man.ac.uk

Abstract

This paper is concerned with the implementation of oriented Gaussian-type filters with several possible decomposition schemes, including the steerable approach and the wavelet transform. A novel spatial decomposition scheme is introduced, in which the filters are projected onto a set of Gaussian basis functions which are not dependent on the orientation of the filters. This scheme only costs a few multiply/add operations per pixel per orientation and provides a considerable improvement in efficiency.

1 Introduction

This paper is concerned with the implementation of matched filters for curve and line detection. It is known from basic communication theory that the matched filter is the optimal estimator for signal (structure) in the presence of additive Gaussian noise, and many workers have sought to exploit the matched filter in line detection [1] [2] [3].

Recently these ideas have been extended to the detection of circular arc segments [4]. It is generally accepted that oriented elongated filters offer a performance advantage over simple edge detectors and this has been confirmed in a previous study [4]. However, there are a number of problems which arise in exploiting matched filters for vision applications:

- a) Computational cost: This is high because one convolution operation is required for each filter orientation. In the applications which we have studied at least 20 different orientations are required, implying 20 large convolutions per image.
- b) Peak finding: When a matched filter is used to find lines and arcs, the transformed image typically contains many ridge-like structures, with only 'shallow' local maxima. The generation of an accurate list of local maxima is difficult using a conventional window search. Phase correlation promises much sharper local maxima but this is at the expense of a loss in robustness.

*This work was supported by Sino-British Friendship Scholarship Scheme

- c) Data processing: When oriented filters are used to characterise curved boundaries, the local maxima correspond to samples of tangent space (at a discrete set of orientations). Previous work has addressed the case of continuous variation in orientation; for example steerable filters can be used to produce ‘super’ edge maps [5]. Convincing studies have been published on the benefit of oriented filters in image restoration [2]. However, few studies on vision applications are available. With long extended oriented filters a small change in orientation does not give independent data from the image. This fact motivates our study of the discrete orientation problem.

In this paper, we address problem a) by examining several possible decomposition schemes for a matched arc detector. The outcome of this comparative study is an algorithm which costs only a few multiply/add operations per pixel per orientation.

We will address problems b) and c) in a future contribution.

2 Arc detector

The arc detector which is extensively used in our image characterisation work, is the first directional derivative of an elongated and curved Gaussian function [4]. This is defined as

$$F^\theta(x, y) = -\frac{\partial k^\theta}{\partial x} \cos \theta + \frac{\partial k^\theta}{\partial y} \sin \theta, \quad (1)$$

where the filter kernel $k^\theta(x, y)$ is

$$k^\theta(x, y) = \exp\left(-\frac{d^2}{2\sigma_\alpha^2}\right) \exp\left(-\frac{p^2}{2\sigma_\beta^2}\right), \quad (\sigma_\alpha < \sigma_\beta) \quad (2)$$

and

$$\begin{aligned} d &= r - \sqrt{(x - r \cos \theta)^2 + (y - r \sin \theta)^2}, \\ p &= \sqrt{x^2 + y^2}, \end{aligned} \quad (3)$$

where θ is the orientation of the filter and r is the radius of the filter curve. σ_α determines the width of the filter and σ_β determines the length of the filter. When $\theta = 0$ and $r \rightarrow \infty$, then equation (2) will represent a elongated Gaussian function.

$$k^0(x, y) = \exp\left(-\frac{x^2}{2\sigma_{\alpha'}^2}\right) \exp\left(-\frac{y^2}{2\sigma_\beta^2}\right), \quad (4)$$

where

$$\sigma_{\alpha'}^2 = \sigma_\alpha^2 / (1 + R^{-2}), \quad \text{and} \quad R = \sigma_\beta / \sigma_\alpha \quad (5)$$

It can be seen that $\sigma_{\alpha'}$ is the standard deviation of the short axis and σ_β is that of the long axis of the elongated Gaussian function.

In our application, the aspect ratio R usually takes values from 5 to 10. This high aspect ratio leads to difficulties in decomposing the filter as discussed in the following sections.

3 Steerable decomposition

Generally, an edge map can be obtained by convolving an oriented filter with the image at either a continuous set or a discrete set of orientations. If N orientations are required, then N convolutions are needed. The computational cost obviously increases linearly with N . Since convolution is a linear operation, the filtering process may be decomposed into a linear sum of the convolutions of the image with a set of basis functions, which may not be dependent on the orientations of the filter.

Supposing an image $I(x, y)$ is filtered using an oriented filter $f^\theta(x, y)$. The filtered image is given by

$$I_{conv}^\theta(x, y) = I(x, y) * f^\theta(x, y), \quad (6)$$

where ‘*’ denotes convolution. If $f^\theta(x, y)$ can be written as

$$f^\theta(x, y) = \sum_{m=1}^M K_m(\theta) G_m(x, y), \quad (7)$$

where $G_m(x, y)$ is a set of base functions and M is the number of base functions, then

$$I_{conv}^\theta(x, y) = \sum_{m=1}^M K_m(\theta) (I(x, y) * G_m(x, y)). \quad (8)$$

It can be seen from equation (8) that the number of convolutions is fixed by M . If M is small, then $(N - M)$ convolutions will be saved, where N is the number of orientations required.

Freeman and Adelson [5] have noted that higher order derivatives of Gaussians have the property called steerability. Here the basis functions are rotated versions of the filter itself. A simple example of steerable filter is the first directional derivative g_1 of the Gaussian function g , where

$$g(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \quad (9)$$

which can be written as

$$g_1^\theta(x, y) = \frac{\partial g}{\partial x} \cos \theta + \frac{\partial g}{\partial y} \sin \theta = g_1^{0^\circ} \cos \theta + g_1^{90^\circ} \sin \theta. \quad (10)$$

However, for many filter functions of interest, a compact decomposition as in equation (7) can not be found. Perona [3] suggested an approximation scheme to solve this difficulty. If $f^\theta(x, y)$ does not satisfy equation (7), he suggested the use of a function $f_M^\theta(x, y)$ which satisfies equation (7) and $f_M^\theta(x, y) \approx f^\theta(x, y)$. The Fourier transform of $f^\theta(x, y)$ in the orientation domain was introduced to obtain $f_M^\theta(x, y)$. The compactness of $f_M^\theta(x, y)$ depends on the complexity of $f^\theta(x, y)$ and the error which can be tolerated in the approximation. For example, for the second derivative of elongated Gaussian functions with aspect ratios of 1:1, 1:2, and 1:3, 3, 7, and 12 components respectively, are needed to achieve reconstruction error of 5%. Twelve components for a ratio 1:3 means 12 convolutions plus 12 additions per pixel are required to process one image. The computational advantage is not great if we only need 20 orientations. For higher ratios, there is even less advantage.

4 Wavelet decomposition

We have investigated a scheme involving the wavelet transform. Equation (7) is rewritten as follows

$$f^\theta(x, y) = \sum_{m, n_1, n_2} C_{m, n_1, n_2}^\theta \psi_{m, n_1, n_2}(x, y) \quad (11)$$

where $\psi_{m, n_1, n_2}(x, y)$ is a wavelet basis function defined by

$$\psi_{m, n_1, n_2}(x, y) = 2^{-m} \psi(2^{-m}x - n_1, 2^{-m}y - n_2) \quad (12)$$

and C_{m, n_1, n_2}^θ are the coefficients of the wavelet transform of $f^\theta(x, y)$ defined by

$$C_{m, n_1, n_2}^\theta = \sum_{x, y} f^\theta(x, y) \psi_{m, n_1, n_2}(x, y) \quad (13)$$

The filtered image can then be written as

$$\begin{aligned} I_{conv}^\theta(x, y) &= I(x, y) * f^\theta(x, y) \\ &= \sum_{m, n_1, n_2} C_{m, n_1, n_2}^\theta (I(x, y) * \psi_{m, 0, 0}(x - 2^m n_1, y - 2^m n_2)) \end{aligned} \quad (14)$$

Since $I(x, y) * \psi_{m, 0, 0}(x - 2^m n_1, y - 2^m n_2)$ can be obtained by translating $I(x, y) * \psi_{m, 0, 0}(x, y)$ by $(2^m n_1, 2^m n_2)$, then m convolutions are needed. Here m is the level of multi-resolution decomposition. For a 512×512 image, m is 9.

The key element of this scheme is the sparseness of C_{m, n_1, n_2} . Normally, most coefficients C_{m, n_1, n_2} of the wavelet transform are zero or near zero. We can approximate $f^\theta(x, y)$ using only a few coefficients, whilst maintaining an acceptable approximation error. Figure 1(a) and figure 1(b) give RMS error defined by equation (15) against number of coefficients, where 5 different wavelets are used. They are two compact supported orthonormal wavelets 'db4' and 'db12', suggested by Daubechies [6] and one uncompact symmetric wavelet 'ma', given by Mallat [7](TABLE I) and two biorthogonal wavelets 'bi1' and 'bi2', given by Antonini [8](TABLE I and TABLE II). Notice that, for 5% error, more than 100 coefficients are needed for the second directional derivative Gaussian filter and more than 200 coefficients for the curved filter with aspect ratio 4. With this number of coefficients, the computational benefit from the decomposition is small.

The normalised Root Mean Square(RMS) error between two functions $f_1(x, y)$ and $f_2(x, y)$ will be defined as

$$\sqrt{(\int \int (f_1 - f_2)^2 dx dy) / (\int \int f_1^2 dx dy)} \quad (15)$$

5 Spatial decomposition

5.1 1-D decomposition

Consider two one-dimensional Gaussian functions g_1 and g_2 , defined by

$$g_1(x) = \exp\left(-\frac{x^2}{2\sigma_1^2}\right), \quad \text{and} \quad g_2(x) = \exp\left(-\frac{x^2}{2\sigma_2^2}\right) \quad (16)$$

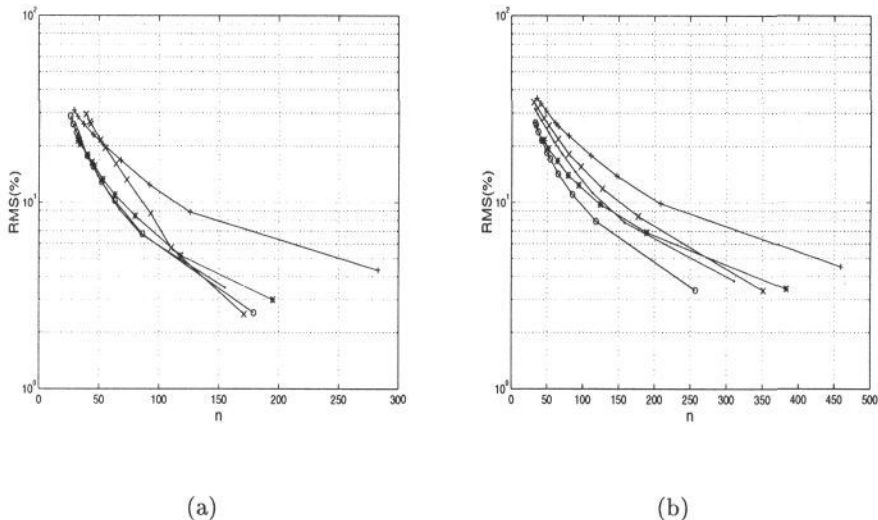


Figure 1: The wavelet decomposition error against number of components ('*', 'o', '·', '+' and 'x' denote wavelet bil, bi2, ma, db4 and db12 respectively), (a) for a second derivative of Gaussian. (b) for the curved filter with parameters $\sigma_\alpha = 5, \sigma_\beta = 20, r = 60$.

let $g'_1(x)$ be an approximation to $g_1(x)$ with the form

$$g'_1(x) = \sum_{m=-l}^l C_m \exp\left(-\frac{(x-mh)^2}{2\sigma_2^2}\right) \quad (17)$$

Notice that $g'_1(x)$ is the sum of a set of translated versions of g_2 . The coefficients $\{C_m\}$ are the coefficients of the least squares projection of the function $g_1(x)$ onto the basis set $\{g_2(x-mh)\}$. Figure 2 shows that the RMS error between g_1 and g'_1 can be arbitrarily small if the translation step h is reduced. Figure 3 shows the optimal value of C_m which are obtained by the least square projection and denoted by '*' in the figure. It can be seen that C_m follows the profile of the Gaussian function $\exp(-(mh)^2/(2\sigma_1^2))$, denoted by the solid line in the figure.

Therefore we can rewrite equation (17) as following

$$g'_1(x) = \sum_{m=-l}^l C \exp\left(-\frac{(mh)^2}{2\sigma_1^2}\right) \exp\left(-\frac{(x-mh)^2}{2\sigma_2^2}\right), \quad (18)$$

where C is a normalizing constant. The RMS error obtained between g_1 and g'_1 defined by equation (18) is identical to that in figure 2.

Summarising the above discussion, it can be seen that, given a Gaussian function g_1 , an approximation function g'_1 can be found which is the projection of g_1 to a series of translated Gaussian function of the form g_2 . The projection coefficients follow the profile of g_1 . The difference between g_1 and its approximation g'_1 is determined by the translation step h , and can be as small as required.

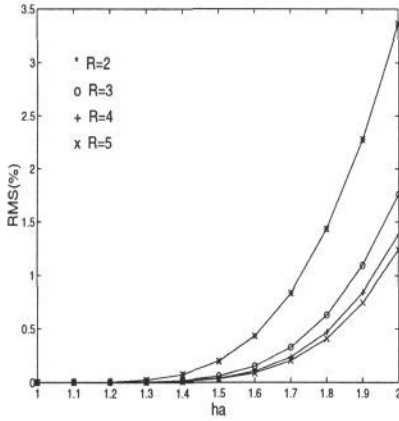


Figure 2: The 1-D decomposition error against translation step ha ($ha = h/\sigma_2$).

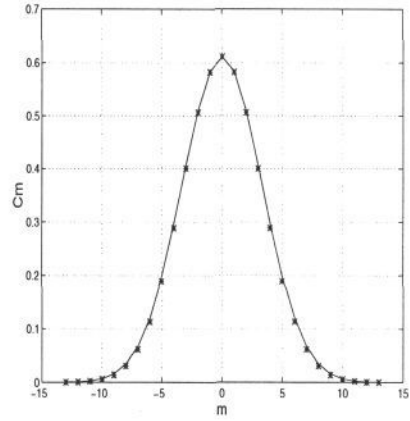


Figure 3: The optimal coefficient C_m denoted by '*', and the Gaussian function $\exp(-(mh)^2/(2\sigma_1^2))$ denoted by solid line for $\sigma_1 = 10$, $\sigma_2 = 2$, $h = 3$.

5.2 2-D decomposition

Extending the above idea to the two-dimensional case, we decompose the filter kernel defined by equation (2) into the sum of a series of translated versions of a 2-D Gaussian function. An approximation function $k^{l\theta}(x, y)$ of $k^\theta(x, y)$ is defined as

$$k^{l\theta}(x, y) = \sum_{m=-l}^l C_m g_b(x - T_{xm}, y - T_{ym}) \quad (19)$$

where

$$g_b(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma_{\alpha'}^2}\right) \quad (20)$$

is the basis function, and

$$C_m = C \exp\left(-\frac{(mh)^2}{2\sigma_o^2}\right) \quad (21)$$

$$T_{xm} = r (\cos(mh/r + \theta) - \cos \theta) \quad (22)$$

$$T_{ym} = r (\sin(mh/r + \theta) - \sin \theta) \quad (23)$$

$$\sigma_o = \sigma_\beta / \sqrt{R^{-2} + 1}. \quad (24)$$

C is a normalizing constant. Assume

$$h_0 = \frac{h}{\sigma_\alpha} (1 + R^{-2}) \quad (25)$$

Then the approximation, $F^{i\theta}(x, y)$, to the arc detector equation (1) can be obtained by

$$F^{i\theta}(x, y) = -\frac{\partial k^{i\theta}}{\partial x} \cos \theta + \frac{\partial k^{i\theta}}{\partial y} \sin \theta \quad (26)$$

Then using the least square method, the optimal normalization coefficient C and the RMS error of $F^\theta(x, y)$ and $F^{i\theta}(x, y)$ can be obtained. Figure 4 depicts the RMS error against h_0 . Notice that for different value of aspect ratio R , all the curves are similar. This means that h_0 is an uniform measure for different R .

During the above discussion, the factor l in equation (19) which controls the number of basis functions used, is unlimited. Notice that C_m in the equation is very small when (mh) is large. For example, when $mh/\sigma_0 = 3.5$, $C_m = 0.0022C$. Hence, we can truncate l to reduce the computational cost. Define

$$n_0 = lh/\sigma_0 \quad (27)$$

Then, replacing h with equation (25) and (27), we have

$$l = \frac{n_0}{h_0} \sqrt{1 + R^2} . \quad (28)$$

Figure 5 shows the RMS error against n_0 when $h_0 = 1.85$. It can be seen that the RMS error tends to a constant as $n_0 > 3.5$, i.e., the RMS error, to a good approximation, depends only on h_0 when $n_0 > 3.5$. Thus, given a RMS error which can be tolerated, say 0.45%, the decision rule can be summarized as:

1. Find the corresponding h_0 from figure 4, which is about 1.85.
2. Fix $n_0 = 3.5$.
3. calculate l by equation (28).
4. The total decomposition components required is $(2l + 1)$.

For example, for $R = 3$ and 10, n will be 13 and 39 respectively. The number of components is still high for high ratio R , though for low ratio R , the result is close to Perona's method [3].

Translation in one of the two functions being convolved only results in the translation of the convolution result. Considering the basis functions in equation (19), they are all translated versions of g_b , and have steerability. When applying this kernel to images, we only need one convolution instead of n convolutions, because g_b is symmetric. However our arc detector is the first derivative of the kernel, and so two convolutions are needed to achieve steerability via equation (10). Thus, if $I(x, y)$ is a image, and $I_{conv}^\theta(u, v)$ is the filtered image, then we have

$$\begin{aligned} I_{conv}^\theta(u, v) &= (I(x, y) * F^{i\theta}(x, y))(u, v) \\ &= \sum_{m=-l}^l C_m [I_{conv}^x(u - T_{um}, v - T_{vm}) \cos \theta \\ &\quad + I_{conv}^y(u - T_{um}, v - T_{vm}) \sin \theta] \end{aligned} \quad (29)$$

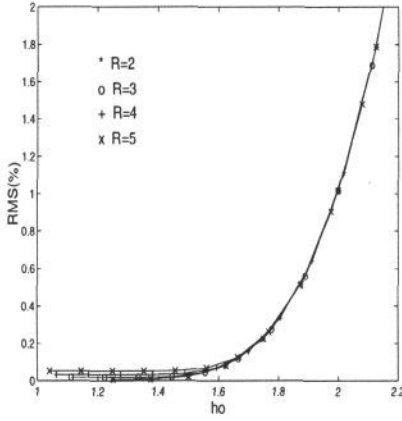


Figure 4: The decomposition error via step h_0 for the curved filter ($r = \infty$).

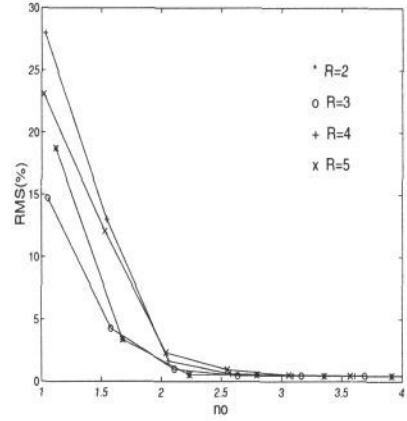


Figure 5: The RMS error against n_0 when $h_0 = 1.85$.

where

$$I_{conv}^x(u, v) = I(x, y) * \frac{\partial g_b(x, y)}{\partial x} \quad (30)$$

$$I_{conv}^y(u, v) = I(x, y) * \frac{\partial g_b(x, y)}{\partial y} \quad (31)$$

Define

$$I_{conv}^{mid}(u, v) = I_{conv}^x(u, v) \cos \theta + I_{conv}^y(u, v) \sin \theta, \quad (32)$$

and notice that $C_m = C_{-m}$. Equation (29) can be rewritten as

$$I_{conv}^\theta(u, v) = I_{conv}^{mid}(u, v) + \sum_{m=1}^l C_m [I_{conv}^{mid}(u - T_{um}, v - T_{vm}) + I_{conv}^{mid}(u - T_{u(-m)}, v - T_{v(-m)})] \quad (33)$$

Hence, we can summarize the decomposition scheme as following:

1. Calculate the two convolutions $I_{conv}^x(u, v)$ and $I_{conv}^y(u, v)$.
2. Get the intermediate value $I_{conv}^{mid}(u, v)$.
3. Translate $I_{conv}^{mid}(u, v)$ by (T_{um}, T_{vm}) and $(T_{u(-m)}, T_{v(-m)})$.
4. Multiply the translation version by C_m and sum them together.

The total computational cost will be

- 2 convolutions overhead.

- $l + 2$ multiplications and $2l + 2$ additions for every pixel.

For example, if $n_0 = 3.5$, $h_0 = 1.85$ and $R = 3$, 8 multiplies and 14 additions per pixel are needed for 0.45% reconstruction error. For a curved filter ($r \neq \infty$), the reconstruction error is larger than that of a straight filter. Figure 6(a) depicts the RMS error against rb ($rb = r/\sigma_\beta$). It can be seen that the reconstruction error is reduced as the aspect ratio R increases and the filter becomes straight.

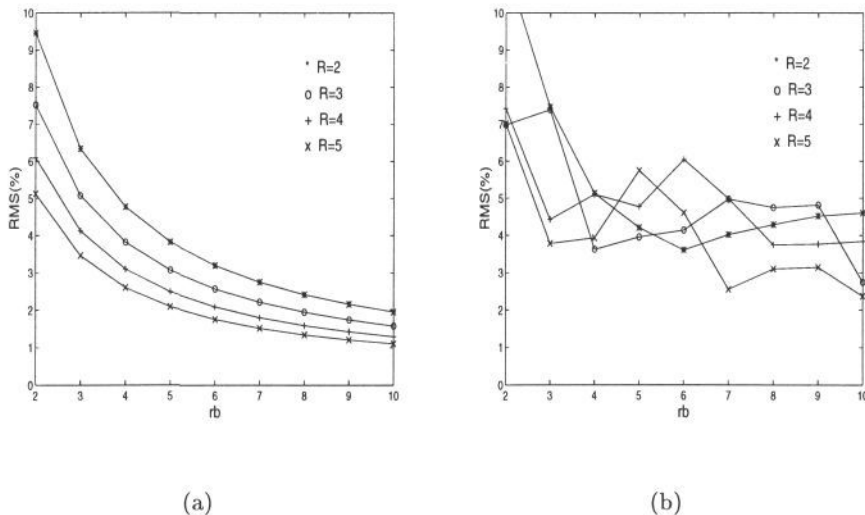


Figure 6: The decomposition error against $rb(= r/\sigma_\beta)$ for the curved filter, (a) using continuous translation offset; (b) using integer-valued offset.

5.3 Problems with translation

One step of our new scheme is to translate $I_{conv}^{mid}(u, v)$ by (T_{um}, T_{vm}) . For discrete images and filters, the intermediate value $I_{conv}^{mid}(u, v)$ is only available at a discrete set of points. Therefore the translating offset (T_{um}, T_{vm}) has to be integer-valued. This can not be guaranteed from equation (22) and (23). Random errors will be inevitably introduced by rounding (T_{um}, T_{vm}) .

$$\Delta T_{um} = T_{um} - \text{round}(T_{um}) \leq 0.5\text{pixel} \quad (34)$$

$$\Delta T_{vm} = T_{vm} - \text{round}(T_{vm}) \leq 0.5\text{pixel} \quad (35)$$

One solution to reduce the random errors is to interpolate the intermediate value, which will increase the computational cost. We use linear interpolation which is the simplest interpolation method. This introduces an extra cost of about three multiplies and three additions per pixel. In this case, the rounding error will be

$$\Delta T_{um} \leq 0.25\text{pixel}, \quad \Delta T_{vm} \leq 0.25\text{pixel} \quad (36)$$

Figure 6(b) shows the RMS error against rb ($rb = r/\sigma_\beta$) using integer-valued offset and the linear interpolation. Comparing it with figure 6(a), it can be noticed that rounding (T_{um}, T_{vm}) deteriorates the approximation accuracy with an increase of about 3% in the RMS error. Thus the overall cost of the system is 13 multiplies and 21 additions per pixel for an error of 4.5% if the elongated curved filter's parameters are $\sigma_\alpha = 5$ pixel, $\sigma_\beta = 20$ pixels and $r = 60$ pixels. Comparing this to the wavelet approach with 200 components (which implies 200 multiplies and 200 additions per pixel), the advantage is obvious.

6 Conclusion

A new technique has been presented for implementing Gaussian kernel filters. This technique gives a good approximation to elongated and curved Gaussian kernel filters. Unlike other techniques, only two convolutions are needed for any orientation or curvature. The benefit of the scheme depends strongly on the specific filter parameters. For a curved filter of $\sigma_\alpha = 2.2, \sigma_\beta = 22, r = 60$, the decomposition requires approximately 25 multiples and 45 additions per pixel. On a Sun4 workstation (for which the time for addition is roughly the same as the time for multiplication) this algorithm is faster than FFT convolution, but only by a factor of around 2. However the speed advantage is greater with filters of smaller aspect ratio.

References

- [1] J. Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, PAMI-8(6):679–698, November 1986.
- [2] H. Knutsson, R. Wilson, and G. H. Granlund. Anisotropic nonstationary image estimation and its applications: Part 1 - restoration of noisy images. *IEEE Trans. on Communication*, 31(3):388–397, 1983.
- [3] P. Perona. Deformable kernels for early vision. *Proc. of the Computer Society Conference on CVPR*, pages 222–227, 1991.
- [4] R. Shann, J. Oakley, D. Davis, and F. White. Detection of circular arcs for content-based retrieval from an image database. *IEE Proc.-Vis. Image Signal Process.*, 141(1):49–55, 1994.
- [5] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. on PAMI*, PAMI-13(9):891–906, Sept 1991.
- [6] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications in Pure and Applied Mathematics*, 41:909–996, 1988.
- [7] S. M. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. on PAMI*, PAMI-11(7), July 1989.
- [8] M. Antonini, M. Badrland, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Trans. on Image Processing*, 1(2), April 1992.