Scale-Space Derived From *B*-Splines

Yu-Ping Wang and S.L. Lee

Abstract—It is well-known that the linear scale-space theory in computer vision is mainly based on the Gaussian kernel. The purpose of the paper is to propose a scale-space theory based on *B*-spline kernels. Our aim is twofold. On one hand, we present a general framework and show how *B*-splines provide a flexible tool to design various scale-space representations: continuous scale-space, dyadic scale-space frame, and compact scale-space representation. In particular, we focus on the design of continuous scale-space and dyadic scale-space frame representation. A general algorithm is presented for fast implementation of continuous scale-space at rational scales. In the dyadic case, efficient frame algorithms are derived using *B*-spline techniques to analyze the geometry of an image. Moreover, the image can be synthesized from its multiscale local partial derivatives. Also, the relationship between several scale-space approaches is explored. In particular, the evolution of wavelet theory from traditional scale-space filtering can be well understood in terms of *B*-splines. On the other hand, the behavior of edge models, the properties of completeness, causality, and other properties in such a scale-space representation are examined in the framework of *B*-splines. It is shown that, besides the good properties inherited from the Gaussian kernel, the *B*-spline derived scale-space exhibits many advantages for modeling visual mechanism with regard to the efficiency, compactness, orientation feature, and parallel structure.

Index Terms—Image modeling, *B*-spline, wavelet, scale-space, scaling theorem, fingerprint theorem.

1 Introduction

Scale is a fundamental aspect of early image representation. Koenderink [1] emphasized that the problem of scale must be faced in any imaging situation. A multiscale representation is of crucial importance if one aims at describing the structure of the world. Both the psychophysical and physiological experiments have confirmed that multiscale transformed information appears in the visual cortex of mammals. This leads to the motivation for the interpretation of image structures in terms of spatial scale in computer vision. Some researchers such as Burt and Adelson [6], Koenderink [1], Marr and Hildreth [10], Witkin [11], and Rosenfeld and Thurston [2] had exposed the necessity and advantages of using operators of different sizes for extracting multiscale information in an image. For a more detailed review, see [3].

The Gaussian scale-space approach of a signal as introduced by Witkin [11], is an embedding of the original signal into a one-parameter family of derived signal constructed by convolution with Gaussian kernels of increasing width. One reason the traditional scale-space is mainly based on the Gaussian kernel is that the Gaussian function is the unique kernel which satisfies the causality property as guaranteed by the scaling theorem [14], [15], [16], [19]; it states that no new feature points are created with increasing scale. Another reason is that the response of the human retina resembles a Gaussian function. Neurophysiological research by Young [22] has shown that there are receptive field profiles in the mammalian retina and visual cortex whose measured response profiles can be well modeled by superposition of Gaussian derivatives. Therefore, the Gaussian function is suitable for modeling the human visual system.

In practice, since the computational load becomes extremely heavy when the scale gets larger, many techniques are proposed for efficient implementation of scale-space filtering. Among them, *B*-splines or binomials have been widely used to approximate the Gaussian kernel. Such examples include, Wells [5], Ferrari et al. [8], [9], Poggio et al. [20], Unser et al. [25], [26], etc. For a more compact representation, the pyramid technique is another widely used representation that combines the subsampling operation with a smoothing step. Historically, they have yielded important steps toward the scale-space theory. The low-pass pyramid representation proposed by Burt [6], [7] is a famous example, which is also closely related to *B*-spline techniques.

The general idea of representing a signal at multiple scales is not new to us. It is through wavelet theory that these early ideas have been well formulated and refined. In fact, this is largely due to the contribution of *B*-spline techniques. As will be shown later, the orthogonal multiresolution pyramid originally proposed by Mallat [40] and the biorthogonal pyramid [28], [45], [46] in wavelet theory can all be derived from *B*-splines [30], [33], [36]. Other types of wavelets such as the wavelets on an interval [47], the periodic wavelets [37], and the cardinal spline wavelets [29] are all related to *B*-splines.

Motivated by these observations, the purpose of this paper is to build a more general framework of scale-space representation in the context of *B*-splines as an improvement of the traditional scale-space theory. By and large, the paper is divided into two parts. Firstly, we present a systematic development of the scale-space representation in the framework of *B*-splines. In particular, we focus on two classes of scale-space design. It is shown that if an image is represented as a *B*-spline surface, efficient subdivision algo-

• *The authors are with the Wavelets Strategic Research Programme, Department of Mathematics, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260.*
E-mail: {wyp, matleesl}@wavelets.math.nus.edu.sg.

Manuscript received 5 May 1997; revised 27 July 1998. Recommended for acceptance by K. Bowyer.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 107250.

rithm can be designed to give a geometric description at varying degrees of detail. The various scale-space representations are derived from *B*-splines in different forms. For continuous scale-space representation, a general algorithm is derived for fast and parallel implementation at rational scales. Several classical fast algorithms are shown to be special cases under some conditions. Differential operators have been used for multiscale geometric description of an image. However, it is not clear whether an image can be synthesized from these differential descriptors. Using *B*-spline techniques, frame algorithms are designed to express the image as combinations of multiscale local partial derivatives. These operators include the gradient operator, second directional operator, Laplacian operator, and oriented operators. At the same time, the intrinsic relationship between wavelet theory and the traditional linear scale-space approach is exhibited. Although *B*-spline has been used in practice in place of Gaussian, there is little effort to consider its scale-space behavior directly. Therefore, the second part of the paper is devoted to examining the properties of *B*-spline derived scale-space. In particular, the advantages of such a scale-space representation are highlighted. It was shown that the *B*-spline derived scale-space inherits most of the nice properties of the Gaussian-derived scale-space. Nevertheless, the *B*-spline kernel outperforms the Gaussian kernel in that it can provide more meaningful, efficient, and flexible description of image information for multiscale feature extraction.

The organization of the paper is as follows. In Section 2, some fundamental properties of *B*-splines are reviewed, which also explain why a *B*-spline kernel is a good kernel for scale-space design. Following this section, we categorize three types of scale-space representation. In particular, we focus on the design of continuous scale-space and dyadic scale-space frame representation. The relation with compact scale-space is discussed. From the viewpoint of *B*-splines the evolution of wavelets from classical continuous scale-space is well understood. Moreover, the equivalence of several famous scale-space methods is explored. In Section 4, a general procedure is presented to study the edge models in the *B*-spline scale-space. In Section 5, some basic properties of the *B*-spline derived scale-space are investigated in parallel with that of the Gaussian-derived scale-space. These include the completeness property, causality property, orientation feature, and so on. Conclusions are given in Section 6. Finally, Appendix A and Appendix B are given in order to make the paper mathematically complete.

2 *B*-Spline Kernels

2.1 Notations and Definitions

We adopt the convention of [25]. Let $L^2(\mathbb{R})$ be the Hilbert space of measurable, square integrable functions on $\mathbb{R}$ and $l^2(\mathbb{Z})$ be the vector space of square summable sequences. We denote the central continuous *B*-spline of order n by $\beta^n(x)$, which can be generated by repeated $n+1$ convolution of a *B*-spline of order zero,

$$\beta^n(x) = \beta^0 * \beta^{n-1}(x) = \overbrace{\beta^0 * \beta^0 * \ldots * \beta^0}^{n+1}(x), \quad x \in \mathbb{R} \tag{1}$$

where the zeroth-order *B*-spline $\beta^0(x)$ is the pulse function with support $\left[-\frac{1}{2}, \frac{1}{2}\right]$. The Fourier transform of $\beta^n(x)$ is

$$\hat{\beta}^n(\omega) = \left[\hat{\beta}^0(\omega)\right]^{n+1} = \left(\frac{\sin \omega/2}{\omega/2}\right)^{n+1} = \sin c^{n+1}\frac{\omega}{2}. \tag{2}$$

For an integer $m \geq 1$, $\beta_m^n(x)$ is defined as the nth-order continuous *B*-spline dilated by a scale factor m, i.e.,

$$\beta_m^n(x) = \frac{1}{m}\beta^n\left(\frac{x}{m}\right). \tag{3}$$

The discrete *B*-spline of order n at scale m is defined as

$$B_m^n = \overbrace{B_m^0 * B_m^0 * \ldots * B_m^0}^{n+1}, \tag{4}$$

where $B_m^0 = \frac{1}{m}[1, 1, \ldots, 1]$ is a normalized sampled pulse of width m.

The discrete sampled *B*-spline $b_m^n(k)$ of order n and integer coarseness $m \geq 1$ is obtained by directly sampling the nth-order continuous *B*-spline at the scale m:

$$b_m^n(k) = \frac{1}{m}\beta^n\left(\frac{k}{m}\right), \quad \forall k \in \mathbb{Z}. \tag{5}$$

We write $b_1^n = b^n$. Consequently, the frequency response of the directly sampled *B*-spline is an aliased version of the frequency response of the continuous *B*-spline, since

$$\hat{b}^n(f) = \sum_{k=-\infty}^{\infty} b^n(k)e^{j2\pi f} = \sum_{k=-\infty}^{\infty} \hat{\beta}^n(f-k), \tag{6}$$

which is due to the Possion's summation formula.

The discrete convolution between two sequences a and b in $l^2(\mathbb{Z})$ is the sequence $b * a$:

$$b * a(k) = \sum_{l=-\infty}^{\infty} b(k-l)a(l), \quad \forall k \in \mathbb{Z}. \tag{7}$$

Under this definition, the convolution is commutative. The convolution inverse $(b)^{-1}$ of a sequence b is defined by

$$((b)^{-1} * b)(k) = \delta(k), \quad \forall k \in \mathbb{Z}, \tag{8}$$

where $\delta(k)$ is the unit impulse whose value is one at zero and zero elsewhere.

The decimation operation $[b]_{\downarrow m}$ down-samples the sequence b by the integer factor m, i.e.,

$$[b]_{\downarrow m}(k) = b(mk), \quad \forall k \in \mathbb{Z}. \tag{9}$$

Conversely, the operator $[b]_{\uparrow m}$ up-samples by the integer factor m, i.e., it takes a discrete signal b and expands it by padding $m-1$ zeros between consecutive samples:

$$[b]_{\uparrow m}(k') = \begin{cases} b(k) & k' = mk \\ 0 & otherwise \end{cases} \tag{10}$$

2.2 The Similarity Between Gaussian and *B*-Spline

B-splines are good approximations of the Gaussian kernel which is commonly used in computer vision. This is the consequence of the central limit theorem. For a review of

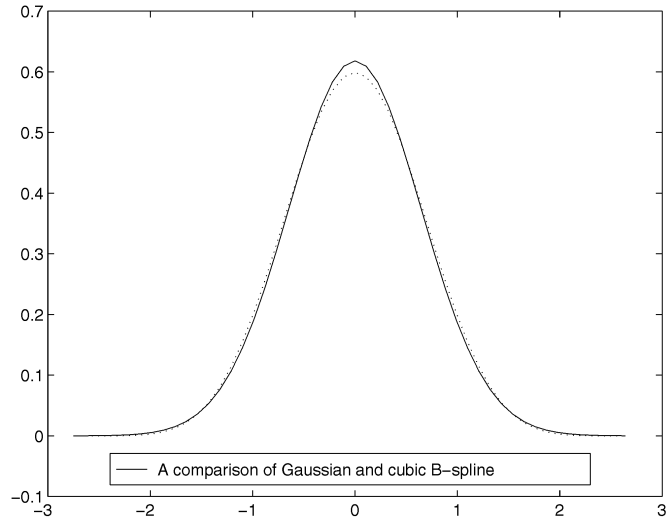

Fig. 1. A comparison between the Gaussian and the cubic *B*-spline kernel. The Gaussian function is drawn in a solid line and the cubic *B*-spline is drawn in a dotted line.

the *B*-spline window and other famous filters, see Torachi et al. [39]. In [28], Unser et al. have presented a more general proof that *B*-splines converge to the Gaussian function in $L^p(\mathbb{R})$, $\forall p \in [2, +\infty)$ as the order of the spline n tends to infinity. Since the variance of the nth order of *B*-spline is $\frac{n+1}{12}$, the approximation relation is as follows:

$$\beta^n(x) \approx \sqrt{\frac{6}{\pi(n+1)}} \exp\left(-\frac{6x^2}{n+1}\right). \tag{11}$$

Furthermore, by numerical computation [28], it was shown that the cubic *B*-spline is already near optimal in terms of time/frequency localization, in the sense that its variance product is within 2 percent of the limit specified by the uncertainty principle. A graphical comparison between the Gaussian kernel and the cubic *B*-spline is given in Fig. 1. Moreover, both the physiological and biological experiments [22] have shown that the human visual system can be modeled with the Gaussian kernel. Therefore, *B*-splines are also suitable for modeling biological vision due to their close approximation to the Gaussian kernel.

2.3 Stable Hierarchical Representation of a Signal by *B*-Splines

Another significant property of the *B*-spline of a given order n is that it is the unique compactly supported refinable spline function of order n which can provide a stable hierarchical representation of a signal at different scales. It has been proven [38] that a compactly supported spline is m-refinable and stable if and only if it is a shifted *B*-spline. Let $h > 0$ and define the polynomial spline space S_h^n consisting of the dilated and shifted *B*-splines of order n (n is odd, which we will assume throughout the paper) by

$$S_h^n = \left\{ \sum_{k=-\infty}^{+\infty} c(k)\beta_h^n(x - hk) : c \in l^2(\mathbb{Z}) \right\}. \tag{12}$$

Then

$$S_{im}^n \subset S_m^n, \quad \forall i \in \mathbb{Z}_+, \tag{13}$$

and

$$\overline{\bigcup_{h>0} S_h^n} = L^2(\mathbb{R}). \tag{14}$$

The embedding property (13) follows from the fact that the *B*-spline $\beta^n(x)$ is m-refinable, i.e., it satisfies the following m-scale relation,

$$\frac{1}{m}\beta^n\left(\frac{x}{m}\right) = \sum_{k=-\infty}^{+\infty} B_m^n(k)\beta^n(x - k). \tag{15}$$

The m-refinability of the *B*-splines can be easily verified [29], [38]. It also establishes the intrinsic relationship between the continuous *B*-spline and discrete *B*-spline. If we take $m = 2$, it is just the commonly used two-scale relation and $B_2^n(k)$ is the discrete binomial. Olkkonen [52] has used the binomial kernels for designing multiresolution wavelet bases.

From the m-scale relation (15), we can also establish the relationship between the discrete sampled *B*-spline and the discrete *B*-spline:

$$b_m^n(k) = B_m^n * b^n(k), \quad \forall k \in \mathbb{Z}. \tag{16}$$

Since *B*-splines provide a stable multiresolution representation of a signal at multiple scales, it is preferable to select *B*-splines as smoothing kernels to extract multiscale information inherent in an image. Therefore, it is not surprising that many vision models [40], [29], [30] are derived from *B*-splines.

One can refer to [25], [26], [33], [46] for a more complete and extensive exposition of the *B*-spline methods. For example, its minimal support and m-refinability properties have led to fast implementation of the scale-space algorithms [27], [5], [8], [9] in computer vision. In the following section, we will classify scale-space representations into three types and show how *B*-splines are used as a flexible tool for designing an efficient visual model according to requirements.

3 Scale-Space Representations Designed From *B*-Splines

In this section, we focus on the fast implementation of continuous scale-space filtering and the design of dyadic scale-space frame representation. Their relations with the compact scale-space representation or compact wavelet models are indicated.

3.1 Implementation of Continuous Scale-Space Filtering Using *B*-Splines

3.1.1 Discrete Signal Approximation Using B-Spline Bases

In practice, a discrete sets of points are given. Because spline spaces S_h^n provide close and stable approximations of $L^2(\mathbb{R})$, it is reasonable to parameterize the discrete signal or image using B-spline bases. We use the translated B-splines of order n_1 as bases to approximate the signal,

$$f(x) \approx \tilde{f}(x) = \sum_k c(k)\beta^{n_1}(x - k), \tag{17}$$

i.e., the signal $f(x) \in L^2(\mathbb{R})$ is projected into the spline space $S_1^{n_1}$ at resolution one. In (17), we have assumed that the sampling rate is one for convenience. We can call this procedure the generalized sampling of the original data, where the sampling basis function is taken as the *B*-spline. There are different types of approximation (see [25]). A common approach ([25], [26]) is the direct *B*-spline transform where the exact or reversible representation of a discrete signal $f(k)$ in the space of *B*-splines is obtained by imposing the interpolation condition: $\forall k \in \mathbb{Z}$, $\tilde{f}(x)\big|_{x=k} = f(k)$. Thus, the coefficient $c(k)$ can be computed as,

$$c(k) = \left(b_1^n\right)^{-1} * f(k), \tag{18}$$

where $\left(b_1^n\right)^{-1}$ denotes the inverse of the discrete sampled *B*-spline which can be computed recursively.

3.1.2 Fast Algorithm for Continuous Scale-Space Filtering at Rational Scales

In this section, we use the B-splines to derive a filter bank algorithm for fast implementation of continuous scale-space filtering.

The linear scale-space representation is to make a map of a signal at multiple scales by changing the scale parameter continuously. In the language of wavelet transform, the traditional scale-space approach can be regarded as a continuous wavelet transform of the signal $f \in L^2$,

$$Wf(s,x) = \int f(t)\psi_s(t-x)dt, \quad s>0, \tag{19}$$

where $\psi_s(x) = \frac{1}{s}\psi\left(\frac{x}{s}\right) \in L^2$ is the scaled wavelet. Because the geometric features of an image are characterized using differential descriptors, $\psi(x)$ is often taken as certain derivative of a smooth kernel or has certain order of vanishing moments. Under different physical meanings, various linear scale-space representations are proposed [4] where different kernels ψ are assumed. Here, we also use the *B*-spline of order n_2 to approximate the wavelet $\psi(x)$:

$$\psi(x) = \sum_k g(k)\beta^{n_2}(x-k). \tag{20}$$

In the classical scale-space theory, two frequently used multiscale edge detection filters are the famous Marr-Hildreth operator [10] and Canny operator [12], which are obtained by taking the first and second derivative of the Gaussian kernel respectively. Since *B*-splines are good approximations of the Gaussian kernel, we shall use the derivatives of *B*-splines instead. In such cases, the coefficients g in (20) are the coefficients of first- and second-order difference operators respectively, i.e.,

$$g^{(1)} = \{1, -1\} \quad \textit{and} \quad g^{(2)} = \{1, -2, 1\}. \tag{21}$$

Such spline wavelets are shown in Fig. 2. Using these spline wavelets, we obtain the approximate Marr-Hildreth operator [10] and Canny operator [12], respectively. Higher order derivatives of *B*-splines can detect edges with higher singularity [42] and the coefficients g are the binomial-Hermite sequences. Explicitly, in the Fourier domain, the rth-order difference of the *B*-spline of order n can be written as

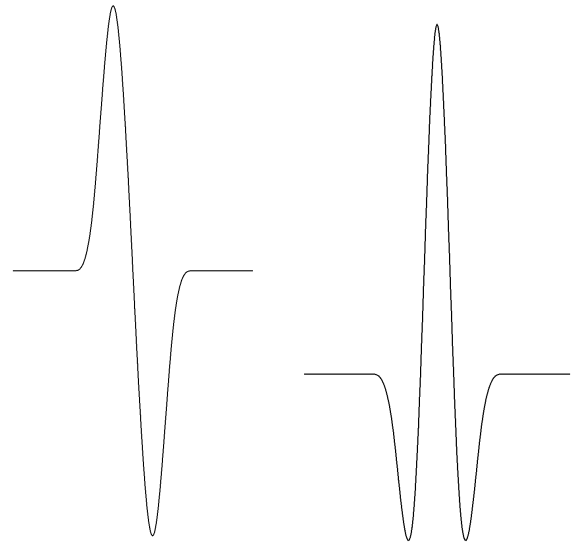

Fig. 2. Two types of wavelets for extrema and zero-crossing detection. The order of the *B*-spline is four.

$$\hat{\psi}(\omega) = \left(e^{i\omega}-1\right)^r \hat{\beta}^n(\omega) = e^{i\frac{r}{2}\omega}(i\omega)^r \hat{\beta}^{n+r}(\omega), \tag{22}$$

which is also the rth-order derivative of the *B*-spline of order $n + r$. We remark that *B*-splines can also efficiently approximate other kinds of wavelets, such as the generalized edge detectors in the $\lambda\tau$-space representation [4], and the coefficients g can be computed numerically.

Since a real number can be approximated arbitrarily close by a rational number $s = \frac{m_1}{m_2}$, $m_1, m_2 \in \mathbb{Z}$, we take rational scale and derive a general filter bank implementation of the scale-space representation of (19) using the m-refinable relation (15). The cascaded implementation of (19) with ψ given in (20) and f approximated by (17) is:

$$Wf\left(\frac{m_1}{m_2}, r\right) = m_2\left(b^{n_1+n_2+1} * B_{m_2}^{n_1} * B_{m_1}^{n_2} * c_{\uparrow m_2} * g_{\uparrow m_1}\right)_{\downarrow m_2}(r), \quad r \in \mathbb{Z} \tag{23}$$

The derivation of this algorithm is given in Appendix A. This algorithm extends that in [27]. If the scale is taken as an integer, i.e., when $m_2 = 1$, then the resulting formula is similar to that in [27]:

$$Wf(m_1, r) = \left(b^{n_1+n_2+1} * B_{m_1}^{n_2} * c * g_{\uparrow m_1}\right)(r), \quad r \in \mathbb{Z}. \tag{24}$$

The implementation of the above algorithm is illustrated in the block diagram in Fig. 3.

In the filter bank implementation of (23), we can interchange the order of convolution. The result is then equivalent to the discrete *B*-spline filtering of the difference of the discrete sampled signal (with a down-sampling)

$$S(m_1, m_2)(k) = B_{m_2}^{n_1} * B_{m_1}^{n_2} * S_1(k), \tag{25}$$

where $S_1(k)$ is the signal sampling followed by difference operation

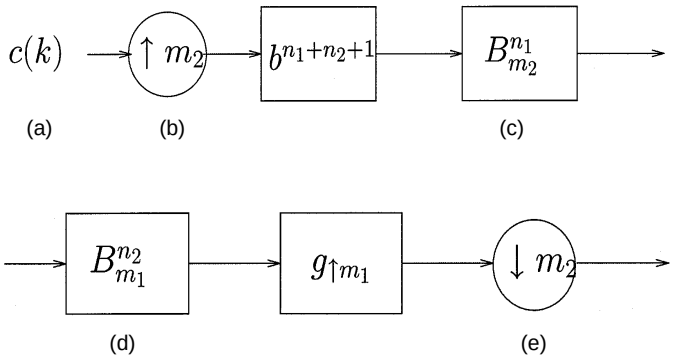

Fig. 3. Block diagram for fast realization of continuous scale-space filtering at the rational scales. (a) Signal sampling. (b) Upsampling. (c) Running average sum. (d) Running average sum. (e) Downsampling.

$$S_1(k) = \left(b^{n_1+n_2+1} * c_{\uparrow m_2} * g_{\uparrow m_1}\right)(k). \tag{26}$$

We note that the computational complexity is mainly due to the discrete *B*-spline filtering (25) which can be implemented efficiently. By (4), it turns into the cascaded convolution with the zeroth-order of discrete *B*-spline and can be implemented via the *running average sum technique*. If we define such a running average operation as

$$R_i(k) = B^0_{m_1} * R_{i-1}(k), \tag{27}$$

then it can be realized using the following iterative strategy

$$R_i(k) = R_i(k-1) + R_{i-1}(k-1) - R_{i-1}(k-m_1-1). \tag{28}$$

Therefore, starting from the initial coefficient $R_0(k) = S_1(k)$, we can compute (25) using only the addition. Then after a downsampling with a factor m_2, we obtain the scale-space filtering at the rational scale $\frac{m_1}{m_2}$. Suppose m_2 is fixed, as is usually the case in practice, the computational cost is independent of the scale. Fig. 4 shows an example of the scale-space filtering of a simulated signal using the above algorithm.

We record two important properties of this procedure.

- *Efficiency*: In practice, m_2 is usually fixed. The computational complexity at each scale $\frac{m_1}{m_2}$ is $O(N)$. The computational complexity is largely due to the convolution with the smoothing *B*-spline kernel which can be reduced by the running average sum technique. In contrast with the existing procedure based on direct numerical integration or FFT-based scheme, the computational complexity does not increase with the increasing number of values of the scale parameter.
- *Parallelism*: The structure of the above algorithm is parallel and independent across scales. This makes it inexpensive to run on arrays of simple parallel processors. In other words, this can be ideally suited for VLSI implementation.

One can recall that the above subdivision algorithm is also similar to the à trous algorithm [48], [32] for fast computation of continuous wavelet transform except with some constraints on their filters. However, the à trous algorithm can only compute the wavelet transform at dyadic scales.

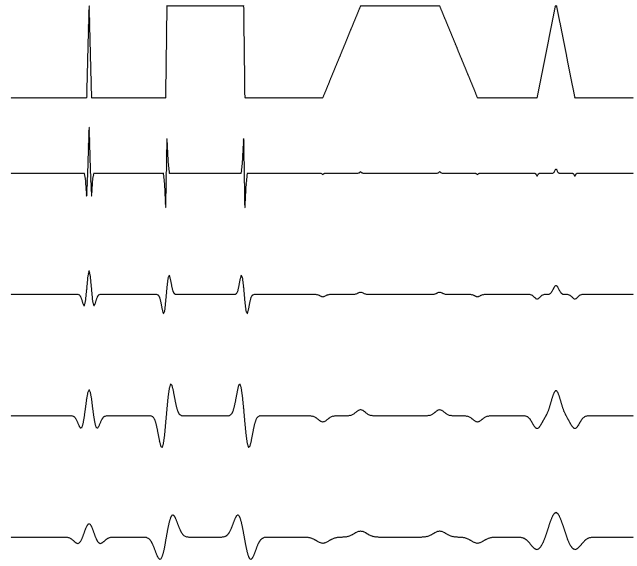

Fig. 4. Fast and parallel implementation of scale-space filtering of a simulated signal at the scale 1.5, 2.5, 3.5, and 4.5 using the second wavelet in Fig. 2.

The above algorithm can compute the wavelet transform efficiently at any scales. As will be shown in Section 3.2, if the scale is restricted to dyadic, the above algorithm is similar to the à trous algorithm. However, using *B*-splines, a more efficient scheme can be obtained which only needs an *addition* operation.

3.1.3 Extension to 2D Images

Although the above algorithm is derived in the one-dimensional case, it can be easily extended to two dimensions. The tensor-product B-spline $\beta^n(x, y) = \beta^n(x)\beta^n(y)$ is used as a basis to parameterize the image and approximate the two-dimensional wavelet kernels. For example, we can use the tensor product B-splines to approximate the Maar-Hildreth's LoG operator [10]:

$$\nabla^2\beta^n(x)\beta^n(y) = \psi^n(x)\beta^n(y) + \psi^n(y)\beta^n(x), \tag{29}$$

where

$$\psi^n(x) = \frac{\partial^2}{\partial^2 x}\left(\beta^n(x)\right) = \beta^{n-2}(x+1) - 2\beta^{n-2}(x) + \beta^{n-2}(x-1).$$

Since this two-dimensional kernel is represented by the separable one-dimensional *B*-spline bases, by performing the above one-dimensional fast algorithm along the horizontal and the vertical orientation, respectively, the LoG operator can be computed efficiently. Fig. 5 shows such results for Lena image at three different scales. Using the spline technique, the computational complexity is the same at different scales and only an addition operation is needed.

Fig. 5. Fast implementation of LoG operator at scales one, five, and seven using cubic *B*-spline.

3.2 Dyadic Scale-Space Frame Representation

3.2.1 One-Dimensional Signal Representation by Its Local Partial Derivatives at Dyadic Scales

The above continuous scale-space is too redundant for some applications. In addition, as stated by Witkin [11], an initial representation ought to be as compact as possible, and its elements should correspond as closely as possible to meaningful objects or events in the signal-forming process. A description that characterizes a signal by its extrema and those of its first few derivatives is a qualitative description to "sketch" a function. If we sample the scale of the above continuous scale-space as dyadic while keeping the time variable continuous, we can obtain a more compact scale-space representation. Moreover, such a representation is shift invariant and therefore is suitable for some pattern recognition applications. In particular, using B-spline techniques efficient frame algorithms can be designed to express the signal in terms of its local partial derivatives.

We now show the relationship between this type of transform and the above continuous scale-space implementation.

If we use the approximation $b^{n_2+n_1+1}*c$ to replace the original signal f, and the width of the B-spline m is restricted to be dyadic, say 2^m, we will get a recursive relation for the wavelet transform (19) between the dyadic scales. In this case, (23) becomes

$$W_{2^m} f(k) = Wf\left(2^m, k\right) = S_{2^m} f * g_{\uparrow 2^m}(k) \tag{30}$$

where

$$S_{2^m} f(k) = B^n_{2^m} * f(k). \tag{31}$$

It is easy to derive the two-scale relations for $B^n_{2^m}$ and S_{2^m}. From property (4) and the property of z-transform,

$$\hat{B}^n_{2^m}(z) = \left[\hat{B}^0_{2^m}(z)\right]^{n+1}, \tag{32}$$

where $\hat{B}^n_{2^m}(z)$ is the z-transform of $B^n_{2^m}(k)$. We have the following relation

$$\hat{B}^n_{2^m}(z) = \frac{1}{2^{n+1}} \sum_{j=0}^{n+1} z^{-\frac{n+1}{2}} \binom{n+1}{j} z^{-j2^{m-1}} \hat{B}^n_{2^{m-1}}(z). \tag{33}$$

In the time domain, it becomes

$$B^n_{2^m}(k) = \frac{1}{2^{n+1}} \sum_{j=-\frac{n+1}{2}}^{\frac{n+1}{2}} \binom{n+1}{j+\frac{n+1}{2}} B^n_{2^{m-1}}\left(k - j2^{m-1}\right). \tag{34}$$

By this relation, we can get a fast recursive implementation of dyadic-scale space filtering,

$$S_{2^m} f(k) = \frac{1}{2^{n+1}} \sum_{j=-\frac{n+1}{2}}^{\frac{n+1}{2}} \binom{n+1}{j+\frac{n+1}{2}} S_{2^{m-1}} f\left(k - j2^{m-1}\right). \tag{35}$$

Or simply written as

$$S_{2^m} f = S_{2^{m-1}} f * h_{\uparrow 2^{m-1}}, \tag{36}$$

where

$$h(j) = \frac{1}{2^{n+1}} \binom{n+1}{j}, \quad 0 \le j \le n+1$$

is the binomial kernel.

The approximate Marr-Hildreth operator and Canny operator at dyadic scales can now be computed as

$$W_{2^m} f = S_{2^{m-1}} f * g^{(k)}_{\uparrow 2^{m-1}}, \tag{37}$$

where $g^{(k)}$, $k = 1, 2$ is the first or second order of difference operator given in (21). The recursive refinement relation (36) and (37) can be rewritten in the z-transform domain as

$$\hat{S}_{2^m} f(z) = H\left(z^{2^{m-1}}\right) \hat{S}_{2^{m-1}} f(z), \quad H(z) = \left(\frac{z^{-\frac{1}{2}} + z^{\frac{1}{2}}}{2}\right)^{n+1} \tag{38}$$

$$\hat{W}_{2^m} f(z) = G^{(k)}\left(z^{2^{m-1}}\right) \hat{S}_{2^{m-1}} f(z),$$
$$G^{(k)}(z) = (z-1)^k, \quad k = 1, 2 \tag{39}$$

By requiring the reconstruction filter $\tilde{G}$ to satisfy the following perfect reconstruction condition

$$H(z)\overline{H}(z) + G(z)\tilde{G}(z) = 1, \tag{40}$$

we can reconstruct the signal from its multiscale partial derivatives

$$S_{2^{m-1}} f = S_{2^m} f * h_{\uparrow 2^m} + W_{2^m} f * \tilde{g}^k_{\uparrow 2^m}, \tag{41}$$

where $\tilde{g}^k, k = 1, 2$ are the time responses of $\tilde{G}^k$ which are given explicitly in Appendix B. Since all of these filters are

linear combinations of binomial and divided by $\frac{1}{2^{n+1}}$, using the Pascal triangular algorithm, only the *addition operation* and *bit shift operation* are needed. This is very suitable for hardware implementation.

3.2.2 Image Representation by Its Local Directional Derivatives

The tensor product B-spline basis is $\beta^n(x, y) = \beta^n(x)\beta^n(y)$. It is interesting that we can still derive an efficient frame algorithm to characterize an image from its local differential components. A fast algorithm for the gradient case has been proposed [42] and further refined in [34]. Now we consider the case of the second directional derivative.

For edge detection, an approach is to detect the zero-crossings of the second directional derivative of the smoothed image $f * \beta^n_{2^j}(x, y)$ along the gradient orientation

$$\frac{\partial^2\left(f * \beta^n_{2^j}(x,y)\right)}{\partial n^2} = \left(\frac{\partial}{\partial x}\cos\theta + \frac{\partial}{\partial y}\sin\theta\right)^2 \left(f * \beta^n_{2^j}(x,y)\right). \quad (42)$$

We can still derive a subdivision algorithm to compute the three local partial derivative components or wavelet transforms:

$$W^1_{2^j} f(x,y) = \frac{\partial^2\left(f * \beta^n_{2^j}(x,y)\right)}{\partial x^2} = f * \psi^{n,1}(x,y), \quad (43)$$

$$W^2_{2^j} f(x,y) = \frac{\partial^2\left(f * \beta^n_{2^j}(x,y)\right)}{\partial y^2} = f * \psi^{n,2}(x,y), \quad (44)$$

$$W^3_{2^j} f(x,y) = \frac{\partial^2\left(f * \beta^n_{2^j}(x,y)\right)}{\partial x \partial y} = f * \psi^{n,3}(x,y), \quad (45)$$

where the three-directional wavelet components are defined in the Fourier domain as

$$\begin{aligned}
\hat{\psi}^{n,1}(\omega_x, \omega_y) &= G^{(2)}(\omega_x)\hat{\beta}^n(\omega_x, \omega_y),\\
\hat{\psi}^{n,2}(\omega_x, \omega_y) &= G^{(2)}(\omega_y)\hat{\beta}^n(\omega_x, \omega_y),\\
\hat{\psi}^{n,3}(\omega_x, \omega_y) &= G^{(1)}(\omega_x)G^{(1)}(\omega_y)\hat{\beta}^n(\omega_x, \omega_y)
\end{aligned} \quad (46)$$

where $G^{(1)}$ and $G^{(2)}$ are the transfer functions of the first- and second-order difference operator. From these definitions, we can obtain a recursive algorithm for the computation of the three local partial-derivative components:

$$\begin{cases}
S_{2^j} f &= S_{2^{j-1}} f * (h, h)_{\uparrow 2^{j-1}}\\
W^1_{2^j} f &= S_{2^{j-1}} f * \left(g^{(2)}, d\right)_{\uparrow 2^{j-1}}\\
W^2_{2^j} f &= S_{2^{j-1}} f * \left(d, g^{(2)}\right)_{\uparrow 2^{j-1}}\\
W^3_{2^j} f &= S_{2^{j-1}} f * \left(g^{(1)}, g^{(1)}\right)_{\uparrow 2^{j-1}}
\end{cases} \quad (47)$$

where $I * (h, g)_{\uparrow 2^{j-1}}$ represents the separable convolution of the rows and columns of the image with the one-dimensional filters $[h]_{\uparrow 2^{j-1}}$ and $[g]_{\uparrow 2^{j-1}}$, respectively. The symbol d denotes the Dirac filter whose impulse is one at the origin and zero elsewhere. We can also reconstruct the image from these dyadic wavelet transforms using the following formula

$$\begin{aligned}
S_{2^{j-1}} f = {} & W^1_{2^j} f * \left(\tilde{g}^{(2)}, u\right)_{\uparrow 2^{j-1}} + W^2_{2^j} f * \left(u, \tilde{g}^{(2)}\right)_{\uparrow 2^{j-1}} +\\
& W^3_{2^j} f * \left(\tilde{g}^{(1)}, \tilde{g}^{(1)}\right)_{\uparrow 2^{j-1}} + S_{2^j} f * (h, h)_{\uparrow 2^{j-1}}
\end{aligned} \quad (48)$$

where

$$u(j) = \frac{1}{2^{2n+2}}\binom{2n+2}{j}, \quad 0 \le j \le 2n+2$$

is the FIR of the transfer function $U(\omega) = H^2(\omega)$. The reconstruction formula (48) follows from the following perfect reconstruction identity:

$$\begin{aligned}
& H^2(\omega_x)H^2(\omega_y) + G^{(2)}(\omega_x)\tilde{G}^{(2)}(\omega_x)U(\omega_y) +\\
& G^{(2)}(\omega_y)\tilde{G}^{(2)}(\omega_y)U(\omega_x) + G^{(1)}(\omega_x)\tilde{G}^{(1)}(\omega_x)G^{(1)}(\omega_y)\tilde{G}^{(1)}(\omega_y) = 1
\end{aligned}$$

If we define the three corresponding reconstruction wavelets

$$\begin{aligned}
\hat{\chi}^{n,1}(\omega_x, \omega_y) &= \tilde{G}^{(2)}(\omega_x)H^2(\omega_y)\hat{\beta}^n(\omega_x, \omega_y),\\
\hat{\chi}^{n,2}(\omega_x, \omega_y) &= \tilde{G}^{(2)}(\omega_y)H^2(\omega_x)\hat{\beta}^n(\omega_x, \omega_y),\\
\hat{\chi}^{n,3}(\omega_x, \omega_y) &= \tilde{G}^{(1)}(\omega_x)\tilde{G}^{(1)}(\omega_y)\hat{\beta}^n(\omega_x, \omega_y)
\end{aligned}$$

it can be shown [34] that an image $f(x, y)$ can be represented as

$$\begin{aligned}
& f(x,y) =\\
& \sum_{j=-\infty}^{\infty}\left(W^1_{2^j} f * \chi^{n,1}(x,y) + W^2_{2^j} f * \chi^{n,2}(x,y) + W^3_{2^j} f * \chi^{n,3}(x,y)\right)
\end{aligned} \quad (49)$$

One can also notice that in the above decomposition and reconstruction formula all the filters are binomial which require only the *addition operation*. For illustration, Fig. 6 shows the above decomposition and reconstruction results of a square image at the scales 1, 2, 4. Like the compact wavelet decomposition [40], the above algorithms also decompose an image into horizontal, vertical, and diagonal components. However, this transform has explicit physical meaning and is shift-invariant. This can be useful for certain pattern recognition tasks.

3.2.3 Image Representations by Its Isotropic and Multiorientational Derivative Components

One can obtain a more compact isotropic wavelet representation of an image that is complete and efficient using a radial B-spline as the smoothing kernel in two dimensions. This representation is important because it indicates that an image can be recovered from its multiscale LoG-like components. The radial B-spline $\phi(x, y)$ is a nonseparable function of two variables defined by its Fourier transform

$$\hat{\phi}(\omega_x, \omega_y) = \hat{\beta}^n(\rho) \quad (50)$$

where the radius

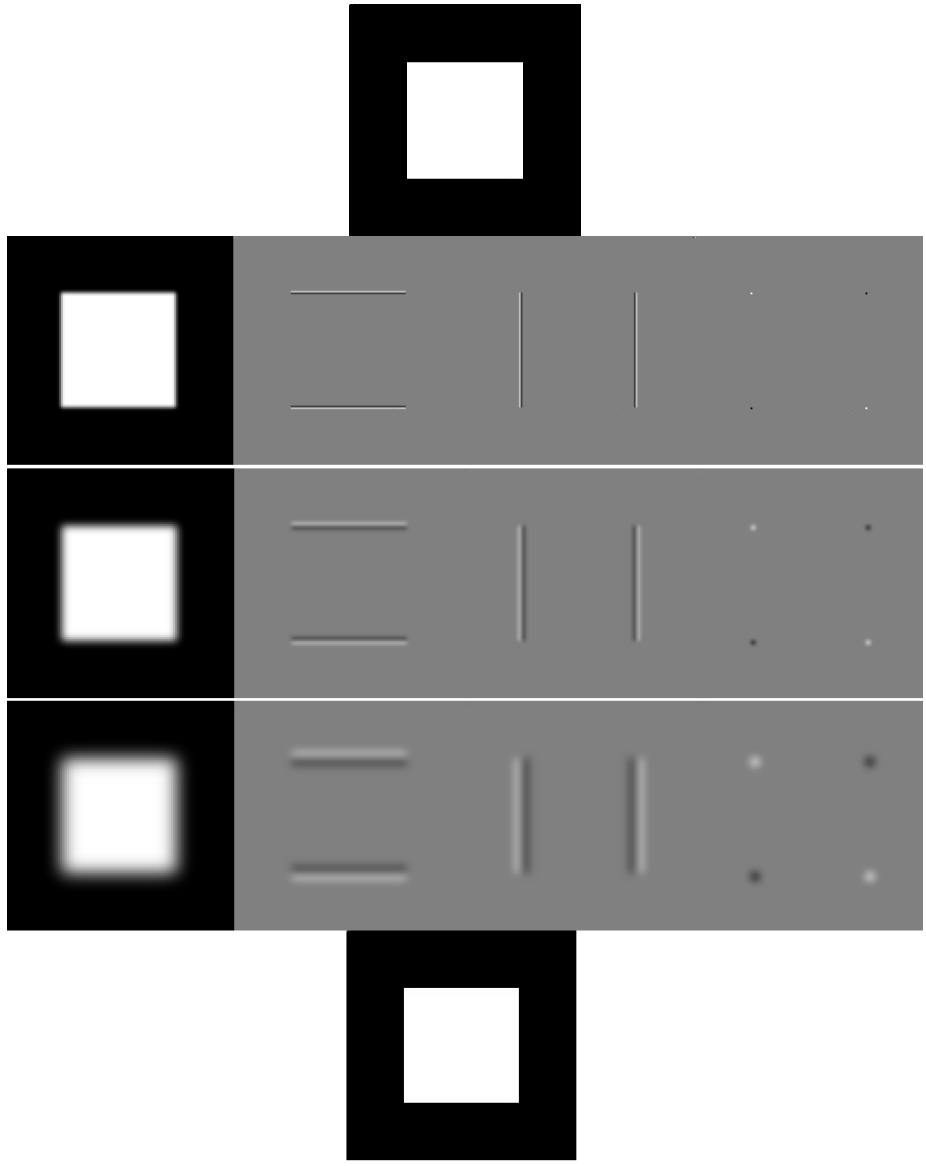

Fig. 6. Representation of a square image by its second order of directional derivative components at dyadic scales. In the top row is the simulated square image and in the bottom row is the reconstructed image. Between them are the directional decompositions along the vertical and diagonal orientations at dyadic scales one, two, and four, respectively.

$$\rho = \min\left(\frac{\sqrt{\omega_x^2 + \omega_y^2}}{2}, \frac{\pi}{2}\right)$$

and the wavelet is defined by

$$\hat{\psi}(\omega_x, \omega_y) = \rho^2 \hat{\phi}(\rho). \tag{51}$$

One may notice that such a wavelet is isotropic and LoG-like, which resembles the human visual system. With these definitions, we still have a filter bank implementation of the decomposition and reconstruction. We omit the details and just give the decomposition formulas,

$$\begin{cases} S_{2^j} f = S_{2^{j-1}} f * h^r_{\uparrow 2^{j-1}} \\ W_{2^j} f = S_{2^{j-1}} f * g^r_{\uparrow 2^{j-1}} \end{cases} \tag{52}$$

where h^r, g^r are the 2D nonseparable radial filters corresponding to h and g, respectively. In this decomposition, two components are obtained at each resolution. By designing the filter $\tilde{g}^r$ from the relation (40), the corresponding 2D nonseparable radial filter $\tilde{g}^r$ can be computed numerically via its Fourier transform $\tilde{G}^r$. Then the reconstruction formula similar to (41) can be obtained. Also, it is easy to check using the same arguments as in the 1D case that an image can be represented as [34]:

$$f(x, y) = \sum_j W_{2^j} f * \chi_{2^j}(x, y) \tag{53}$$

where χ_{2^j} is the 2D reconstruction wavelet defined by $\hat{\chi}(\rho) = \tilde{G}^r(\rho)\hat{\phi}(\rho)$.

One can build a wavelet representation having as many orientation tunings as desired by using nonseparable wavelet bases. A generalized Pythagorean theorem has been proved to decompose an image into a finite number of equally spaced angles [34]:

$$\sum_{k=0}^{n-1} H_k(\theta)\overline{H}_k(\theta) = 1,$$

where

$$H_k(\theta) = j\sqrt{\frac{(2m)!!}{(2m-1)!!\,n}}\cos^m\left(\theta - \frac{k}{n}\right),\ (0 \le k \le n-1).$$

If we multiply the above isotropic wavelet (51) by the angular part $H_k(\theta)$, $0 \le k \le n-1$, $n \in \mathbb{Z}$, we can extract the orientational information in the dyadic scale-space tuned to n orientations [34]:

$$\hat{\psi}_k(\rho,\theta) = \hat{\psi}(\rho)H_k(\theta),\quad 0 \le k \le n-1. \tag{54}$$

Such wavelets can be called orientation tuned LoG-like filters. An image can be represented by its multiscale and multiorientational components,

$$f(x,y) = \sum_{k=0}^{n-1}\sum_{j=-\infty}^{\infty}\left(W_{2^j}^k f * \chi_{2^j}^k\right)(x,y), \tag{55}$$

where $\chi_{2^j}^k$ is the oriented wavelet for reconstruction. Similarly, the pyramid-like filter bank implementation of such a representation can be obtained as follows,

$$\begin{cases} S_{2^j} f = S_{2^{j-1}} f * [h]_{\uparrow 2^{j-1}} \\ W_{2^j}^k f = S_{2^{j-1}} f * [g_k]_{\uparrow 2^{j-1}}, \quad 0 \le k \le n-1 \end{cases}. \tag{56}$$

Therefore, through such an approach we can analyze the directional information of an image feature at a certain angle in dyadic scale space. In Fig. 7, we show a multiscale orientational decomposition and reconstruction, where the orientation number is chosen as three.

3.2.4 Some Comments on the Application of Dyadic Scale-Space Representation

The dyadic scale-space frame representation from B-spline gives rise to many applications, since it provides an invertible, translation-invariant, and pyramid-like compact representation of a signal. One example is the fingerprint-based compression [41] by combining with other techniques. Many image features such as ridges, corners, blobs, and junctions are usually characterized by local differential descriptors [3]. It is usually enough to consider their behaviors only at dyadic scales. The proposed algorithms provide efficient ways and are easy for hardware implementation. For example, in multiscale shape representation, usually the computation of the curvature function is treated in continuous scales [50]. In fact, sometimes it is enough to consider its behavior only at the dyadic scales [49]. We have used the above algorithms to efficiently compute the geometric descriptors for multiscale shape analysis [35].

3.3 Compact Scale-Space Representation

While the dyadic scale-space frame approach provides a more compact representation, it is still over-complete for signal representation. For image compression applications, compact representation is preferred. In order to give a complete picture, we mention briefly the discrete wavelet transform. While the scale-space technique has existed for a long time, it was the orthogonal multiresolution representation proposed by Mallat [40] that makes the mathematical structure of the image more explicit. This is an extension or refinement of traditional scale-space theory. This approach restricts the scale to dyadic and samples the time variable. The starting point is to orthogonalize the *B*-spline basis, and then decompose the signal approximated at a fine scale-space $S_{2^{i+1}}^n$ into a coarser scale space $S_{2^i}^n$ by imposing the orthogonal condition

$$S_{2^{i+1}}^n = S_{2^i}^n \oplus W_{2^i}^n. \tag{57}$$

The detail irregular information of the signal is contained in the subspace $W_{2^i}^n$. This defines an orthogonal multiscale representation. After the *B*-spline basis is converted into an orthogonal basis, the two-scale relation still exists which results in an efficient pyramidal algorithm. The perfect reconstruction condition (40) still exists. However, additional conditions on the filters H, G, $\tilde{G}$ are imposed to ensure the orthogonality. There are several ways to achieve a compact multiresolution by imposing the biorthogonal instead of the orthogonal condition (57). All these compact multiresolutions are related to *B*-splines. A detailed study can be found in [30], [36]. We remark that these wavelet filters can be factored into B-spline filters and hence can be implemented more efficiently [36].

From the above analysis, it is easy to see that the dyadic scale-space frame representation lies between the continuous scale-space and the compact representation. Which kind of representation to select depends on the problem at hand. For example, in multiscale feature extraction, one may compute the differential operation either at the continuous scales or only dyadic scales. Therefore, the continuous or dyadic scale-space frame representation is more useful. However, for compression applications, the compact multiresolution is the favorite.

3.4 Relations Between the Existing Scale-Space Algorithms in Computer Vision

Before the appearance of wavelets, the *B*-spline technique has been widely used in computer vision. Examples include Wells [5], Burt [6], [7], and Ferrari et al. [8], [9]. We shall show that under certain circumstances, they are either equivalent or the special cases of the general algorithm given in Section 3.

3.4.1 Relation to Ferrari et al.'s Method

Ferrari et al. [8], [9] have proposed *B*-spline functions to realize the 2D image filtering recursively. Their idea is to use *B*-splines as the filter kernel:

$$g(m,n) = \sum_k\sum_l f(k,l)h(m-k,n-l), \tag{58}$$

$$h(m,n) = \sum_{k=1}^{p}\sum_{l=1}^{q}\hat{h}(kM,lN)\beta^s(m-kM)\beta^t(m-kM), \tag{59}$$

where $\hat{h}(kM,lN)$ are the interpolation coefficients at the knots (kM, lN), which can be computed using the usual

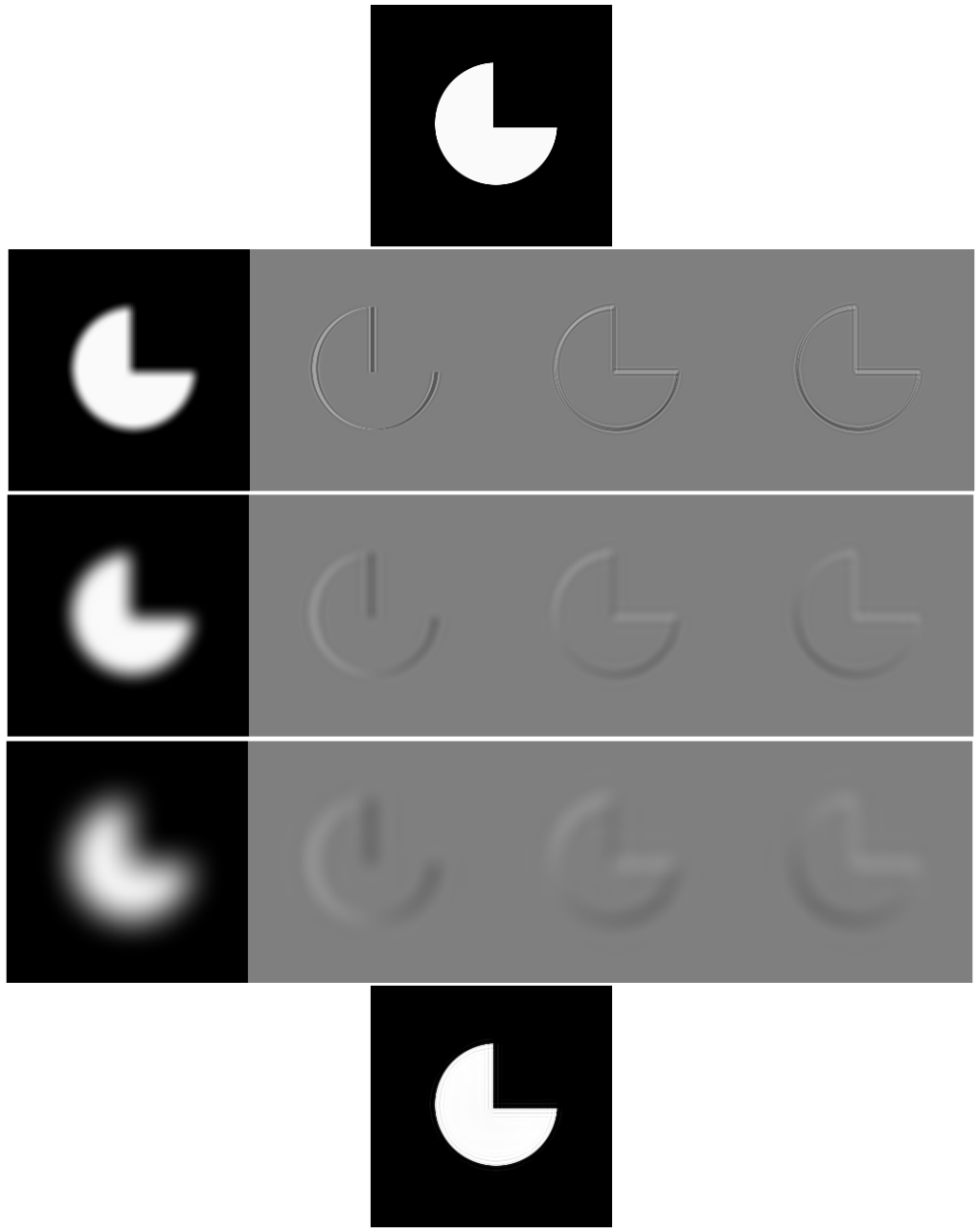

Fig. 7. Multiorientational decomposition and reconstruction at dyadic scales. Here, the orientation number is taken as three. In the top row is the simulated image and in the bottom row is the reconstructed image. Between them are the multiorientational decompositions at dyadic scales one, two, and four, respectively. Using the proposed algorithm, an image can be decomposed into any orientation.

method. Then using the properties of discrete *B*-spline, they derive the recursive 2D image filtering. In this way, the computational load can be greatly reduced. Therefore, this approach is a special case of our proposed algorithm for continuous scale-space filtering.

3.4.2 Relation to Wells' Method

In [5], Wells proposed an approach for efficient synthesis of Gaussian filters by cascaded uniform filters. It is easy to show that his method is equivalent to using cascaded zeroth discrete *B*-spline to approximate the Gaussian kernel. By this approach, the cascaded convolution with a zeroth-degree *B*-spline (uniform filters) can be realized by running the average sum technique as discussed above. Obviously, his method is a special case of our recursive algorithm.

3.4.3 Relation to Burt's Laplacian Pyramid Algorithm

Burt [6], [7] has introduced the following low-pass filter for the generation of Gaussian or Laplacian pyramids

$$w(0) = a, \quad w(1) = w(-1) = 1/4, \quad w(2) = w(-2) = 1/4 - a/2. \tag{60}$$

If the parameter a is taken as $a = 3/8$, then $w(j)$ can be rewritten as

$$w(j) = \frac{1}{4}\binom{4}{j+2}.$$

This is equivalent to the special case of $n = 3$ in (35). In this case, the filter is also equivalent to the operator used for the generation of the dual cubic spline pyramid representation as discussed in [29].

4 Edge Patterns in the *B*-Spline Based Scale-Space

The study of edge patterns in scale-space is very important for many applications. Much work has been done on the study of edge patterns in the Gaussian based scale-space. In this section, we want to investigate the edge behaviors in the *B*-spline based scale-space.

It is well-known that there exists a similar "uncertainty principle" between good edge localization and noise removal. At finer scale, better localization can be achieved with the cost of noise pollution, and vice versa. Many researchers have studied the localization of operators based on Gaussian kernel in scale-space. Berzins [24] studied the accuracy of Laplacian operator, Shah et al. [23] considered the localization of pulse and staircase edge models, Clark [21] investigated the phantom edges in a scale-space. For corner detection, Asada and Brady [50], Chin [51] have also considered the behavior of edge models. We found that their derivations are all based on Gaussian kernel. It is necessary to study the behavior of various edge models in *B*-spline based scale-space which give some a priori knowledge of various patterns in an image.

Here, we present a more general proof with the assumption that the primitive θ in the definition $\psi^1(x) = \frac{d}{dx}\theta(x)$ is symmetric and with compact support $[-w/2, w/2]$ and its derivatives have the shape as in Fig. 2. In practice, the support of the Gaussian kernel is usually truncated to a finite interval. Obviously, the truncated Gaussian and *B*-spline kernel are included in this assumption. We shall show that these edge models have the same behaviors as that derived from the traditional Gaussian kernel. First, we adopt the following accurate definition of an edge [21].

Definition 1. *A point x_0 is called an authentic edge of a signal $f(x)$, if $|W^1 f(s, x_0)|$ is maxima, or,*

$$W^1 f(s, x_0) \cdot \frac{\partial^2}{\partial x^2} W^1 f(s, x_0) < 0.$$

Otherwise, it will be a phantom edge.

We shall use $W^1 f$ and $W^2 f$ to denote two types of wavelet transform where the wavelets are the first and second derivatives of θ, respectively. Clark [21] analyzed these two types of detection. Generally, zero-crossing detection is equivalent to the extrema detection. However, extrema detection includes both maxima and minima detection. Only the edge point detected by local maxima belongs to authentic edge and the edge point detected by local minima corresponds to phantom edge. It is shown that zero-crossing edge detection algorithms can produce edges which do not correspond to significant image intensity changes. Such edges are called phantom or spurious.

Now, as an example we study the behavior of staircase edge model in scale-space. The staircase edge model can be represented as

$$f(x) = A_1 u(x) + A_2 u(x - d), \quad x \in \mathbb{R} \tag{61}$$

where A_1, A_2 represent the amplitudes of the edge, d is the distance between the two abrupt changes at $x = 0$ and $x = d$, $u(x)$ is the step function,

$$u(x) = \begin{cases} 1 & \text{if} \quad x \geq 0 \\ 0 & \text{if} \quad x < 0 \end{cases} \tag{62}$$

whose derivative in the distributional sense is $\delta(x)$. Hence, from (19),

$$\begin{aligned} W^1 f(s, x) &= A_1 \int_0^{+\infty} \psi_s^1(x)dx + A_2 \int_d^{+\infty} \psi_s^1(x)dx \\ &= A_1 \theta\left(\frac{x}{s}\right) + A_2 \theta\left(\frac{x-d}{s}\right) \end{aligned} \tag{63}$$

i.e., the wavelet transform is just the sum of two dilated smoothing functions. At the location $x = 0$,

$$\begin{aligned} &\frac{\partial}{\partial x} W^1 f(s, x)\Big|_{x=0} = \frac{A_1}{s}\theta'(0) + \frac{A_2}{s}\theta'(0)\left(-\frac{d}{s}\right) = \\ &\frac{A_2}{s}\theta'\left(-\frac{d}{s}\right) = \begin{cases} > 0 & \frac{d}{s} < \frac{w}{2} \\ = 0 & \frac{d}{s} \geq \frac{w}{2} \end{cases} \end{aligned} \tag{64}$$

Therefore, only at a small scale $s < \frac{2d}{w}$, the location of edge at $x = 0$ can be detected exactly. Similarly, at the location $x = d$,

$$\begin{aligned} &\frac{\partial}{\partial x} W^1 f(s, x)\Big|_{x=d} = \frac{A_1}{s}\theta'\left(\frac{d}{s}\right) + \frac{A_2}{s}\theta'(0) = \\ &\frac{A_1}{s}\theta'\left(\frac{d}{s}\right) = \begin{cases} < 0 & \frac{d}{s} < \frac{w}{2} \\ = 0 & \frac{d}{s} \geq \frac{w}{2} \end{cases} \end{aligned} \tag{65}$$

i.e., the location of edge at $x = d$ can only be detected exactly at a small scale $s < \frac{2d}{w}$. For large scale, it will spread like a cone in scale-space and will be influenced by another cone at $x = 0$. As a consequence, the edge location will be misdetected due to the superposition of two diffused cones. One may deduce that there will be another point x_0 such that $\frac{\partial}{\partial x} W^1 f(s, x)\Big|_{x=x_0} = 0$ due to different parities in the values of $\frac{\partial}{\partial x} W^1 f(s, x)$ at the location of $x = 0$ and $x = d$. However, it is easy to find such a point corresponding to a local minimum, which means it is a phantom edge point. This cannot be distinguished from others by zero-crossing detection. Fig. 8 illustrates the behavior of this type of edge in the scale-space. That was why local maxima is preferred for edge detection in [41], [42].

Similarly, for zero-crossing detection of $W^2 f(s, x)$, we can draw the same conclusion. In the above analysis, we only consider one type of edge model. Other types of edges such as the step, pulse, ramp, roof can be treated in a similar way [34]. Also, our derivation is based on a more general assumption on the kernels which include both the truncated Gaussian and the *B*-spline.

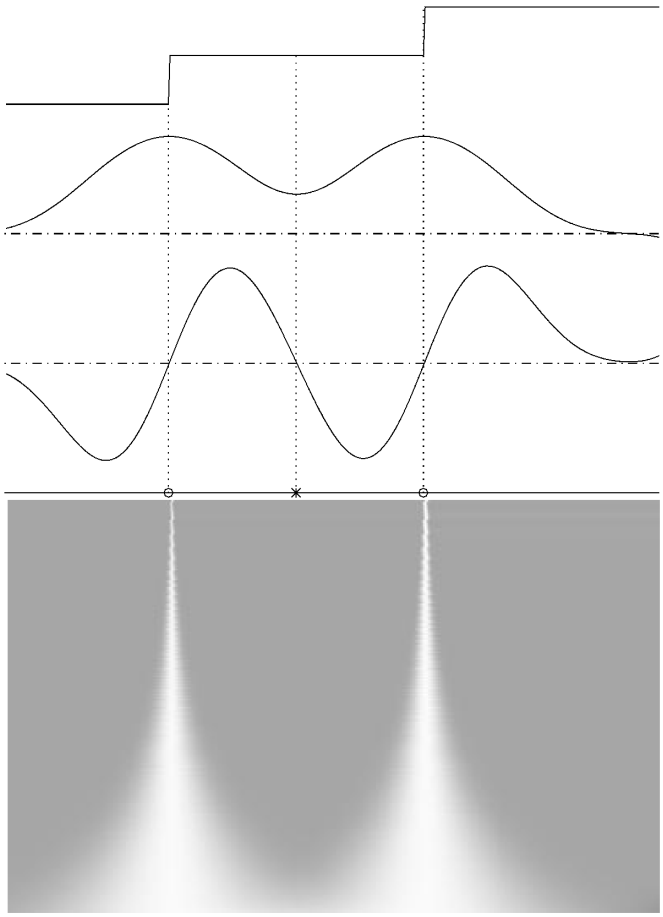

Fig. 8. The scale-space behavior of the staircase edge model. The top row is the staircase edge model. The second and third rows are wavelet transforms using the wavelets in Fig. 2. The symbol * denotes the phantom edge, and o denotes the authentic edges. The two authentic edge points can be localized using maxima detection, and they cannot be distinguished from the phantom edge by the zero-crossing detection. The bottom row is the continuous scale-space map of the staircase edge using the first wavelet by continuously changing the scale along a log axis.

5 Discussions on the Properties of the *B*-Spline-Based Scale-Space

We now discuss the advantages and properties of the *B*-spline based scale-space.

- **Efficiency**: It is a basic requirement that any algorithm should be able to capture and process the meaningful information contained in the signal as fast as possible. Obviously, the major weakness of the traditional Gaussian-based scale-space is the lack of efficient algorithms. On the contrary, *B*-spline techniques facilitate computational efficiency. The computational complexity is scale independent. Moreover, in contrast with the scale-space based on the Gaussian kernel, the *B*-spline representation of a signal is determined directly on an initial discrete set of points, avoiding problems caused by discretization in continuous scale-space. *B*-splines also have been used as smoothing windows for efficient computation of Gabor transform to extract frequency information [31].
- **Parallelism**: Data parallelism is common in computer vision which arises from the nature of an image. As Koendrink [1] pointed out, it seems clear enough that visual perception treats images on several levels of resolution simultaneously and that this fact must be important for the study of perception. In this paper, efficient parallel structure of an image is exhibited using *B*-spline techniques. It may provide a good interpretation of the human visual system which can process the hierarchical information simultaneously. *B*-splines provide a flexible way to process the multiscale information using either coarse-to-fine strategy or in parallel. This is also easy for hardware implementation.
- **Completeness and invertibility**: We usually use the zero-crossings or the local extrema as meaningful description of a signal. It is clearly important, therefore, to characterize in what sense the information in an image or a signal is captured by these primal sketches uniquely. For a Gaussian-based scale-space, the completeness property is guaranteed by the fingerprint theorem [13]: The map of the zero crossing across scales determines the signal uniquely for almost all signals in the absence of noise. Such results have theoretical interest in that they answer the question of what information is conveyed by the zero and level crossings of multiscale Gaussian-filtered signals. Poggio and Yuille's proof is heavily dependent on the Gaussian kernel and they conjectured that under certain conditions, Gaussian kernel is necessary for fingerprints theorem to be true. However, later Wu and Xie [17] gave a negative answer and presented a more general proof, which states that the fingerprint theorem holds for any symmetry kernel. Therefore, **the fingerprint theorem is also true in the case of *B*-splines for continuous scale-space representation**.

  Differential operators have also been widely used for multiscale geometric description of images, but it has not been clear that such representations are invertible. As shown in the paper, using *B*-splines, efficient frame algorithms can be designed to express an image from its local derivatives at dyadic scales.
- **Compactness**: For compression application, we require a representation to be as compact as possible so that an image can be represented by the corresponding primitives using less storage. In [13], Poggio and Yuille conjectured that the fingerprints are redundant and the appropriate constraints derived from the process underlying signal generation should be used to characterize how to collapse the fingerprints into a more compact representation. In the paper, the more compact dyadic scale-space representations are proposed. We can use such representation for compression applications by combining with other techniques.
- **Causality**: Since edge points are important features, it is natural to require that no new features are created as the scale increases. A multiscale feature-detection method that does not introduce new features as the scale increases is said to possess the property of causality. Causality is in fact equivalent to the maximum principle in the theory of parabolic differential

equations [18]. The Gaussian scale-space is governed by the heat-diffusion equation and therefore possesses the causality property. Such continuous causality property of the Gaussian kernel is not shared by the *B*-spline. However in the discrete sense, Aissen et al. [53] proved that for a discrete scale-space kernel h, the number of local extrema or zero-crossings in $f_{out} = h * f_{in}$ does not exceed the number of local extrema or zero-crossings in f_{in} if and only if its generating function

$$H(z) = \sum_{n=-\infty}^{\infty} h(n) z^n$$

is of the form

$$H(z) = cz^k e^{\left(q_{-1}z^{-1}+q_1 z\right)} \prod_{i=1}^{\infty} \frac{\left(1+\alpha_i z\right)\left(1+\delta_i z^{-1}\right)}{\left(1-\beta_i z\right)\left(1-\gamma_i z^{-1}\right)}, \quad (66)$$

where

$$c > 0,\ q_{-1},\ q_1,\ \alpha_i,\ \beta_i,\ \gamma_i,\ \delta_i \geq 0,\ \beta_i,\ \gamma_i < 1, \quad (67)$$

and

$$\sum_{i=1}^{\infty} \left(\alpha_i + \beta_i + \gamma_i + \delta_i\right) < \infty. \quad (68)$$

It is easy to verify that the discrete *B*-spline kernel in (25) satisfies such a condition. Therefore, **the causality property still holds for discrete *B*-spline filtering in the discrete sense**. The number of local extrema or zero-crossings of the derivative of the discrete signal does not increase after running average sum. This justifies the use of the discrete smoothing kernel in practice.

- **Orientation**: Orientation analysis is an important task in early vision and image processing, for example, in texture analysis [44]. The Laplacian multiresolution [6] does not introduce any spatial orientation selectivity into the decomposition process. Daugman [44] showed that these impulse responses can be approximated by Gaussian windows modulated with a wave. It is meaningful to combine both the orientation analysis and the scale feature [43]. In this paper, an efficient pyramid-like algorithm is designed using the *B*-spline technique to analyze and synthesize an image from its multiorientational information at any number of angles in the dyadic scale-space. Note that the usual wavelet transforms can decompose an image in only three orientations.

There are other advantages of the *B*-spline kernel. In the time-frequency analysis, the Gaussian kernel is the optimal function that minimizes the uncertainty principle. The cubic *B*-spline is already a good approximation to the Gaussian function [28], see also Fig. 1. As the order approaches infinity, the *B*-splines converge to the Gaussian in both the time and frequency domain. Moreover, a *B*-spline resembles the response of receptive field [22] and is also suitable for modeling the human visual system. Edge detection is an ill-posed problem. From the view point of the regularization theory, cubic spline is proved optimal. The connection between the regularized edge detection and the smoothing spline problem proposed by Schoenberg, Reinsh in statistics is noted by Poggio et al. [20]. It was shown that the cubic *B*-spline rather than the Gaussian kernel is optimal for edge detection.

B-splines are the shortest basis functions that provide a stable multiresolution analysis of a signal [36], [33]. This explains why many wavelet models of a vision [40], [28], [45], [46], [47], [37] are derived from *B*-splines [33], [36]. For the derivative operations, the *B*-spline approach is very intrinsic which elucidates the relationship between derivative and difference which are usually characterized by the two-scale difference relations. *B*-splines play an important role to bridge the traditional scale-space theory, dyadic scale-space frame, and compact multiresolution representation.

6 Conclusions

This paper describes a *B*-spline based visual model. For a long time, the Gaussian kernel has been commonly used in computer vision. In this paper, a general framework for scale-space representation using *B*-splines is presented. In particular, the design of two types of scale-space representations is given in detail. A fast algorithm for continuous scale-space filtering is proposed. In the case of dyadic scale, some efficient frame algorithms are designed to express the image from its local differential descriptors. The intrinsic relationship with the compact wavelet models is also indicated. Several algorithms are proved to be special cases of our proposed algorithm.

To our knowledge, the scale-space property based on *B*-splines has not been fully studied before. We examine the property of *B*-spline based scale-space in parallel with the Gaussian kernel. Our results indicate that *B*-splines possess almost the same properties as the Gaussian kernel. Moreover, the *B*-spline kernels outperform the Gaussian in many aspects, notably, computational efficiency.

Appendix A: Derivation of Fast Implementation of Continuous Wavelet Transform at Rational Scales

We use the m-scale relation (15) to derive the filter bank implementation of scale-space filtering at rational scales. From (19), (20), (17),

$$\begin{aligned} Wf\left(\frac{m_1}{m_2}, t\right) &= \left(\psi_{\frac{m_1}{m_2}} * f\right)(t) \\ &= \frac{m_2}{m_1}\left(\sum_k g(k)\beta^{n_2}\left(\frac{m_2}{m_1}t - k\right)\right) * \left(\sum_l c(l)\beta^{n_1}(t-l)\right) \\ &= \frac{m_2}{m_1}\sum_k \sum_l g(k)c(l)\beta^{n_2}\left(\frac{m_2}{m_1}t - k\right) * \beta^{n_1}(t-l) \end{aligned} \quad (69)$$

Using the m-refinable relation (15),

$$\beta^{n_2}\left(\frac{m_2}{m_1}t-k\right)=m_1\sum_j B_{m_1}^{n_2}(j)\beta^{n_2}\left(m_2t-m_1k-j\right),$$
$$\beta^{n_1}(t-l)=m_2\sum_i B_{m_2}^{n_1}(i)\beta^{n_1}\left(m_2(t-l)-i\right)$$

Hence,

$$\beta^{n_2}\left(\frac{m_2}{m_1}t-k\right)*\beta^{n_1}(t-l)=$$
$$\left(m_1\sum_j B_{m_1}^{n_2}(j)\beta^{n_2}\left(m_2t-m_1k-j\right)\right)*$$
$$\left(m_2\sum_i B_{m_2}^{n_1}(i)\beta^{n_1}\left(m_2(t-l)-i\right)\right)=$$
$$m_1m_2\sum_j\sum_i B_{m_1}^{n_2}(j)B_{m_2}^{n_1}(i)$$
$$\left(\beta^{n_2}\left(m_2t-m_1k-j\right)*\beta^{n_1}\left(m_2t-m_2l-i\right)\right)=$$
$$m_1\sum_j\sum_i B_{m_1}^{n_2}(j)B_{m_2}^{n_1}(i)\beta^{n_1+n_2+1}\left(m_2t-m_1k-m_2l-j-i\right) \quad (70)$$

where the following property of *B*-spline is used [25]:

$$\beta^{n_2}\left(m_2t-k\right)*\beta^{n_1}\left(m_2t-l\right)=\frac{1}{m_2}\beta^{n_1+n_2+1}\left(m_2t-k-l\right). \quad (71)$$

Substituting (70) into (69) gives

$$Wf\left(\frac{m_1}{m_2},t\right)=m_2\sum_k\sum_l g(k)c(l)\sum_j\sum_i B_{m_1}^{n_2}(j)B_{m_2}^{n_1}(i)\beta^{n_1+n_2+1}$$
$$\left(m_2t-m_1k-m_2l-j-i\right)$$
$$=m_2\sum_k\sum_l g(k)c(l)\sum_j B_{m_1}^{n_2}(j)\left(\beta^{n_1+n_2+1}*B_{m_2}^{n_1}\right)$$
$$\left(m_2t-m_1k-m_2l-j\right)$$
$$=m_2\sum_k\sum_l g(k)c(l)\left(\beta^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}\right)\left(m_2t-m_1k-m_2l\right)$$
$$=m_2\sum_k g(k)\left(\beta^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}*c_{\uparrow m_2}\right)\left(m_2t-m_1k\right)$$
$$=m_2\left(\beta^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}*c_{\uparrow m_2}*g_{\uparrow m_1}\right)\left(m_2t\right)$$
$$=m_2\left(\beta^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}*c_{\uparrow m_2}*g_{\uparrow m_1}\right)_{\downarrow m_2}(t)$$

If we take $t=r\in\mathbb{Z}$, then the above formula can be written as,

$$Wf\left(\frac{m_1}{m_2},r\right)=m_2\left(b^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}*c_{\uparrow m_2}*g_{\uparrow m_1}\right)_{\downarrow m_2}(r). \quad (72)$$

If we take $t=\frac{r}{m_2}, r\in\mathbb{Z}$, we get an interpolation formula, and the size of the transformed data is m_2 times the size of the original sampling signal data:

$$Wf\left(\frac{m_1}{m_2},\frac{r}{m_2}\right)=m_2\left(b^{n_1+n_2+1}*B_{m_2}^{n_1}*B_{m_1}^{n_2}*c_{\uparrow m_2}*g_{\uparrow m_1}\right)(r). \quad (73)$$

APPENDIX B: DERIVATION OF THE RECONSTRUCTION FILTER RESPONSES

For the first-order difference $G(z)=z-1$, the perfect reconstruction condition (40) gives

$$\tilde{G}(z)=\frac{1-H^2(z)}{G(z)}=\frac{1-\left[\left(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\right)^{n+1}\right]^2}{\left(z^{\frac{1}{2}}-z^{-\frac{1}{2}}\right)z^{\frac{1}{2}}}$$
$$=-\frac{1}{4}\left(1-z^{-1}\right)\sum_{j=0}^{n+1}\left(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\right)^j \quad (74)$$

and the corresponding FIR is

$$\tilde{g}_l=\begin{cases}\frac{1}{2^{2n+2}} & l=-(n+1)\\ -\sum_{j=-l}^{n}\frac{1}{2^{2j+2}}\binom{2j}{j+l}+\sum_{j=-l}^{n+1}\frac{1}{2^{2j}}\binom{2(j-1)}{j+l} & -n\le l\le -1\\ -\sum_{j=l}^{n}\frac{1}{2^{2j+2}}\binom{2j}{j+l}+\sum_{j=l}^{n+1}\frac{1}{2^{2j+4}}\binom{2(j+1)}{j+l+2} & 0\le l\le n-1\\ -\frac{1}{2^{2n+2}} & l=n\\ 0 & \text{otherwise}\end{cases} \quad (75)$$

For the second-order difference $G(z)=z^2-2z+1$, (40) gives

$$\tilde{G}(z)=\frac{1-H^2(z)}{G(z)}=\frac{1-\left[\left(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\right)^{n+1}\right]^2}{z\left(z^{\frac{1}{2}}-z^{-\frac{1}{2}}\right)^2}$$
$$=-\frac{1}{4z}\sum_{j=0}^{n+1}\left(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\right)^j \quad (76)$$

and the corresponding FIR is

$$\tilde{g}_l=\begin{cases}-\frac{1}{4}\sum_{j=|l|}^{n}\frac{1}{2^{2j}}\binom{2j}{j+|l|} & |l|\le n\\ 0 & \text{otherwise}\end{cases} \quad (77)$$

ACKNOWLEDGMENTS

The authors wish to thank the referees for their comments, which greatly improve the presentation of the paper. The first author wishes to express his thanks to Prof. Wu for providing the reference [17].

This work is supported by the Wavelets Strategic Research Programme funded by the National Science and Technology Board and Ministry of Education of Singapore under Grant RP960601 / A.

REFERENCES

[1] J.J. Koenderink, "The Structures of Images," *Biological Cybernetics*, vol. 50, pp. 360-370, 1984.

[2] A. Rosenfeld and M. Thurston, "Edge and Curve Detection for Visual Scene Analysis," *IEEE Trans. Computers*, vol. 20, no. 5, pp. 562-569, 1971.

[3] Bart M. ter Haar Romeny, ed., *Geometry-Driven Diffusion in Computer Vision*. The Hague, The Netherlands: Kluwer Academic Publishers, 1994.

[4] M. Gökmen and A.K. Jain, "$\lambda\tau$-Space Representation of Images and Generalized Edge Detector," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 545-563, June 1997.

[5] M. Wells, "Efficient Synthesis of Gaussian Filters by Cascaded Uniform Filters," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 2, pp. 234-239, 1986.

[6] P.J. Burt and E. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Trans. Communications*, vol. 31, pp. 482-540, 1983.

[7] P.J. Burt, "Fast Hierarchical Correlations With Gaussian-Like Kernels," *Computer Vision, Graphics, and Image Processing*, vol. 16, pp. 20-21, 1981.

[8] L.A. Ferrari, P.V. Sankar, J. Sklansky, and S. Leeman, "Efficient Two-Dimensional Filters Using B-Spline Functions," *Computer Vision, Graphics, and Image Processing*, vol. 35, no. 2, pp. 152-169, 1986.

[9] L.A. Ferrari, P.V. Sankar, S. Shinnaka, and J. Sklansky, "Recursive Algorithms for Implementing Digital Image Filters," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 3, pp. 461-466, 1987.

[10] D. Marr and E. Hildreth, "Theory of Edge Detection," *Proc. Royal Soc. London*, vol. B207, pp. 187-217, 1980.

[11] A.P. Witkin, "Scale-Space Filtering," *Proc. Seventh Joint. Conf. Artificial Intelligence*, pp. 1,019-1,023, Karlsruhe, Germany, 1983.

[12] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, pp. 679-698, 1986.

[13] A.L. Yuille and T. Poggio, "Fingerprints Theorems for Zero-Crossings," *J. Optical Soc. Am.*, vol. 2A, no. 5, pp. 683-692, 1985.

[14] A.L. Yuille and T. Poggio, "Scaling Theorems for Zero-Crossings," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 15-25, Jan. 1986.

[15] J. Babaud, A.P. Witkin, M. Baudin, and R.O. Duda, "Uniqueness of Gaussian Kernel for Scale-Space Filtering," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 26-33, Jan. 1986.

[16] L.D. Wu and Z.H. Xie, "Scaling Theorem for Zero-Crossings," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, Jan. 1990.

[17] L.D. Wu and Z.H. Xie, "On Fingerprint Theorems," *Proc. Ninth Int'l Conf. Pattern Recognition*, pp. 1,216-1,221, Rome, 1988.

[18] R.A. Hummel and A.R. Moniot, "Reconstruction From Zero-Crossings in Scale-Space," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 12, pp. 2,111-2,130, 1989.

[19] V. Anh, J.Y. Shi, and H.T. Tsui, "Scaling Theorems for Zero-Crossings of Bandlimited Signals," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 3, pp. 309-320, Mar. 1990.

[20] T.A. Poggio, V. Torre, and C. Koch, "Computational Vision and Regularization Theory," *Nature*, vol. 317, pp. 314-319, 1985.

[21] I.J. Clark, "Authenticating Edges Produced by Zero-Crossing Algorithms," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 1, pp. 43-57, Jan. 1989.

[22] R.A. Young, "The Gaussian Derivative Model for Machine Vision: Visual Cortex Simulation," *J. Optical Soc. Am.*, July 1987.

[23] M. Shah, A. Sood, and R. Jain, "Pulse and Staircase Edge Models," *Computer Vision, Graphics, and Image Processing*, vol. 34, pp. 321-343, 1986.

[24] V. Berzins, "Accuracy of Laplacian Edge Detection," *Computer Vision, Graphics, and Image Processing*, vol. 27, pp. 195-210, 1984.

[25] M. Unser, A. Aldroubi, and M. Eden, "B-Spline Signal Processing: Part I—Theory," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 821-833, 1993.

[26] M. Unser, A. Aldroubi, and M. Eden, "B-Spline Signal Processing: Part II—Efficient Design and Applications," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 834-847, 1993.

[27] M. Unser, A. Aldroubi, and S.J. Schiff, "Fast Implementation of Continuous Wavelet Transforms With Integer Scale," *IEEE Trans. Signal Processing*, vol. 42, no. 12, pp. 3,519-3,523, 1994.

[28] M. Unser, A. Aldroubi, and M. Eden, "On the Asymptotic Convergence of B-Spline Wavelets to Gabor Functions," *IEEE Trans. Information Theory*, vol. 38, no. 2, pp. 864-872, 1992.

[29] M. Unser, A. Aldroubi, and M. Eden, "The Polynomial Spline Pyramid," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 364-378, Apr. 1993.

[30] M. Unser, A. Aldroubi, and M. Eden, "A Family of Polynomial Spline Wavelet Transforms," *Signal Processing*, vol. 30, no. 2, pp. 141-162, 1993.

[31] M. Unser, "Fast Gabor-Like Windowed Fourier and Continuous Wavelet Transform," *IEEE Signal Processing Letters*, vol. 1, no. 5, pp. 76-79, 1995.

[32] M. Unser, "A Practical Guide to the Implementation of the Wavelet Transform," A. Aldroubi and M. Unser, eds., *Wavelets in Medicine and Biology*, pp. 37-73. Boca Raton, Fla.: CRC Press, 1996.

[33] M. Unser, "Ten Good Reasons for Using Spline Wavelets," *Proc. SPIE*, vol. 3,169, *Wavelet Applications in Signal and Image Processing*, vol. 5, pp. 422-431, 1997.

[34] Y.-P. Wang, "On B-Spline Wavelet Theory and Its Applications in Computer Vision," PhD thesis, Image Processing Center, School of Electronics and Infomation Eng., Xi'an Jiaotong Univ., China, 1996.

[35] Y.-P. Wang, S.L. Lee, and K. Toraichi, "Multiscale Curvature Based Shape Representation Using B-Spline Wavelets," unpublished manuscript, 1997. Available at *http://wavelets.math.nus.edu.sg/~wyp/download_papers/shape2.ps.gz*.

[36] S.L. Lee, A. Sharma, and H.H. Tan, "Spline Interpolation and Wavelets Construction," *Applied and Computational Harmonic Analysis*, vol. 5, no. 3, pp. 249-276, July 1998.

[37] Y.W. Koh, S.L. Lee, and H.H. Tan, "Periodic Orthogonal Splines and Wavelets," *Applied and Computational Harmonic Analysis*, vol. 2, pp. 201-218, 1995.

[38] W. Lawton, S.L. Lee, and Z. Shen, "Characterization of Compactly Supported Refinable Splines," *Advances in Computational Math.*, vol. 3, pp. 137-145, 1995.

[39] K. Toraichi, M. Kamada, S. Itahashi, and R. Mori, "Window Functions Represented by B-Spline Functions," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 1, pp. 145-147, 1989.

[40] S. Mallat, "A Theory for Multiresolution Signal Decomposition: Wavelet Representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674-693, July 1989.

[41] S. Mallat and S. Zhong, "Characterization of Signals From Multiscale Edges," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 7, pp. 710-732, July 1992.

[42] S. Mallat and W.L. Huang, "Singularity Detection and Processing With Wavelets," *IEEE Trans. Information Theory*, vol. 32, no. 2, pp. 617-643, Mar. 1992.

[43] S. Mallat, "Wavelets for a Vision," *Proc. IEEE*, vol. 84, no. 4, pp. 604-614, 1996.

[44] J.G. Daugman, "Complete Discrete 2D Gabor Transform by Neural Networks for Image Analysis and Compression," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 36, pp. 107-114, 1988.

[45] A. Cohen, I. Daubechies, and J.C. Feauveau, "Biorthogonal Bases of Compactly Supported Wavelets," *Comm. Pure Applied Math.*, vol. 45, pp. 485-560, 1992.

[46] C.K. Chui and J.Z. Wang, "On Compactly Supported Spline Wavelets and a Duality Principle," *Trans. Am. Math. Soc.*, vol. 330, no. 2, pp. 903-915, 1992.

[47] C.K. Chui and E. Quak, "Wavelets on a Bounded Interval," D. Brasess and L.L. Schumaker, eds., *Numerical Methods in Approximation Theory*, vol. 9, pp. 53-65. Basel: Birhauser Verlag, 1992.

[48] M.J. Shensa, "The Discrete Wavelet Transform: Wedding the à Trous Algorithms and Mallat Algorithms," *IEEE Trans. Signal Processing*, vol. 40, no. 10, pp. 2,464-2,482, 1992.

[49] J.-S. Lee, Y.-N. Sun, and C.-H. Chen, "Multiscale Corner Detection by Using Wavelet Transform," *IEEE Trans. Image Processing*, vol. 4, no. 1, pp. 100-104, 1995.

[50] A. Asada and M. Brady, "The Curvature Primal Sketch," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 2-3, Jan. 1986.

[51] A. Rattarangsi and R.T. Chin, "Scale-Based Detection of Corners of Planar Curves," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 4, pp. 430-449, Apr. 1992.

[52] H. Olkkonen, "Discrete Binomial Splines," *Graphical Models and Image Processing*, vol. 57, no. 2, pp. 101-106, 1995.

[53] M. Aissen, I.J. Schoenberg, and A. Whitney, "On the Generating Functions of Totally Positive Sequences I," *J. d'Anal. Math.*, vol. 2, pp. 93-103, 1952.

Yu-Ping Wang obtained his BSc in applied mathematics in 1990 from Tianjin University, China. Dr. Wang received his MSc in computational mathematics and PhD in communication and electronic systems at the Center for Applied Mathematics and at the Image Processing Center from Xi'an Jiaotong University, China, in 1993 and 1996, respectively. He has been a research fellow with the Wavelets Strategic Research Programme at the National University of Singapore since 1996. His current research interests include computer vision, signal analysis, and various applications of wavelets, partial differential equations, fractals, and splines.

S.L. Lee received his BSc (Hons) and MSc degrees from the University of Malaya, Kuala Lumpur, in 1969 and 1971, respectively, and his PhD in mathematics from the University of Alberta, Canada, in 1974. His research interests include spline approximation, computer-aided geometric design, and wavelets. He is a professor of mathematics at the National University of Singapore and the principal investigator of the Wavelets Strategic Research Programme.