Journal of Mathematical Imaging and Vision, 3, 349–376 (1993).

Discrete Derivative Approximations with Scale-Space Properties: A Basis for Low-Level Feature Extraction

TONY LINDEBERG
Computational Vision and Active Perception Laboratory, Department of Numerical Analysis and Computing Science, Royal Institute of Technology, S-100 44 Stockholm, Sweden

Abstract. This article shows how discrete derivative approximations can be defined so that *scale-space properties hold exactly also in the discrete domain*. Starting from a set of natural requirements on the first processing stages of a visual system, *the visual front end*, it gives an axiomatic derivation of how a multiscale representation of derivative approximations can be constructed from a discrete signal, so that it possesses an *algebraic structure similar* to that possessed by the derivatives of the traditional scale-space representation in the continuous domain. A family of kernels is derived that constitute *discrete analogues* to the continuous Gaussian derivatives.

The representation has theoretical advantages over other discretizations of the scale-space theory in the sense that operators that commute before discretization *commute after discretization*. Some computational implications of this are that derivative approximations can be computed *directly* from smoothed data and that this will give *exactly* the same result as convolution with the corresponding derivative approximation kernel. Moreover, a number of *normalization* conditions are automatically satisfied.

The proposed methodology leads to a scheme of computations of multiscale low-level feature extraction that is conceptually very simple and consists of four basic steps: (i) *large support* convolution smoothing, (ii) *small support* difference computations, (iii) *point operations* for computing differential geometric entities, and (iv) *nearest-neighbour operations* for feature detection.

Applications demonstrate how the proposed scheme can be used for edge detection and junction detection based on derivatives up to order three.

Key words. scale-space, visual front end, smoothing, Gaussian filtering, Gaussian derivative, discrete approximation, edge detection, junction detection, multiscale representation, computer vision, digital signal processing

1 Introduction

A common problem in computer vision concerns how to compute derivative approximations from discrete data. This problem arises, for example, when to compute image descriptors, such as features or differential invariants, from image data and when to relate image properties to phenomena in the outside world. It is, however, well known that derivative estimation is not a well-posed problem. That derivative estimators are known to enhance the noise is often taken as an argument that some sort of smoothing is necessary.

Ultimately, the task of defining derivatives from real-world data boils down to a fundamental and *inherent measurement problem*, namely, the well-known fact that objects in the world and features in images, in contrast to ideal mathematical entities, such as "point" or "line," exist as meaningful entities only over certain finite ranges of scale. This intrinsic property is the basic reason why a *multiscale representation* is crucially important when describing image structure. A methodology proposed by Witkin [34] and Koenderink [15] to obtain such a multiscale representation is by embedding the signal into a one-parameter family of derived signals, the *scale-space*, where the parameter t, called the scale parameter,[1] describes the current level of scale.

Let us briefly review some aspects of this pro-

cedure as it is formulated for two-dimensional continuous signals. Given a signal $f : \mathbb{R}^2 \to \mathbb{R}$, the scale-space representation $L : \mathbb{R}^2 \times \mathbb{R}_+ \to \mathbb{R}$ is defined[2] by $L(\cdot, \cdot; 0) = f(\cdot, \cdot)$ and for $t > 0$ by convolution with the Gaussian kernel $g : \mathbb{R}^2 \times \mathbb{R}_+ \backslash \{0\} \to \mathbb{R}$:

$$L(x, y; t) = (g(\cdot, \cdot; t) * f(\cdot, \cdot))(x, y; t), \quad (1)$$

where

$$g(x, y; t) = \frac{1}{2\pi t} \exp\left(-\frac{x^2 + y^2}{2t}\right). \quad (2)$$

Equivalently, this family can be generated by the solution to the diffusion equation

$$\partial_t L = \frac{1}{2}\nabla^2 L = \frac{1}{2}(\partial_{xx} + \partial_{yy})L, \quad (3)$$

with initial condition $L(\cdot, \cdot; 0) = f(\cdot, \cdot)$. From this representation spatial derivatives of the smoothed intensity function L can be defined at different levels of scale

$$L_{x^i y^j}(x, y; t) = (\partial_{x^i y^j} L)(x, y; t). \quad (4)$$

Since the derivative operator commutes with the smoothing operator, the smoothed derivatives obtained in this way satisfy

$$\partial_{x^i y^j}(g * f) = (\partial_{x^i y^j} g) * f = g * (\partial_{x^i y^j} f), \quad (5)$$

implying that there are, in principle, three equivalent[3] ways to compute them: (i) by differentiating the smoothed signal, (ii) by convolving the signal with the differentiated smoothing kernel, or (iii) by smoothing the differentiated signal. Moreover, the spatial derivatives satisfy the diffusion equation

$$\partial_t L_{x^i y^j} = \frac{1}{2}\nabla^2 L_{x^i y^j} \quad (6)$$

and inherit the cascade smoothing property of Gaussian smoothing

$$L_{x^i y^j}(\cdot, \cdot; t_2) = g(\cdot, \cdot; t_2 - t_1) * L_{x^i y^j}(\cdot, \cdot; t_1) \quad (t_2 > t_1 \geq 0) \quad (7)$$

associated with the semigroup property of the Gaussian kernel

$$g(\cdot, \cdot; t_2) = g(\cdot, \cdot; t_2 - t_1) * g(\cdot, \cdot; t_1) \quad (t_2 > t_1 \geq 0). \quad (8)$$

This type of multiscale representation based on smoothed derivatives, or *Gaussian derivatives*, has been proposed by Koenderink and van Doorn [16], [18] as a possible model for the local processing in the receptive fields in a visual system. By combining the output from these operators at any specific scale, smoothed differential geometric descriptors can be defined at that scale. If such descriptors are defined at *all* scales, the result is a multiscale differential geometric representation of the signal. This type of framework is useful for a variety of early vision tasks.

A problem that arises in this context is how these operations should be discretized when to be implemented in a machine vision system. The theory just described applies to continuous signals, whereas realistic signals obtained from standard cameras are *discrete*. Although a standard discretization of the continuous equations could be expected to give a behaviour that in some sense is relatively close to the behaviour in the continuous case, especially at coarse scales where the grid effects can be expected to be smaller, it is not guaranteed that the original scale-space conditions, however formulated, will be preserved after the discretization. Another important question is how sensitive these operations will be to noise, particularly when derivatives of high order are to be computed from noisy measured data.[4]

In this paper a *discrete analogue of the multiscale Gaussian derivative representation* is developed. The case with discrete signals defined on an infinite and uniformly sampled square grid is treated and a set of discrete operators is proposed, which in a certain sense represents the canonical way to discretize the above-stated continuous expressions. By replacing (i) the above continuous signal by a discrete signal, (ii) the convolution with the continuous Gaussian kernel g by discrete convolution with a kernel T called the discrete analogue of the Gaussian kernel, and (iii) the derivative operators $\partial_{x^i y^j}$ with a certain set of difference operators $\delta_{x^i y^j}$, it is shown how a multiscale representation of discrete derivative approximations can be defined, so that discrete analogues of (1)–(8) *hold exactly after discretization.*

The proposed representation has theoretical advantages over traditional discretizations based on different versions of the sampled or integrated Gaussian kernel and over discretizations carried out in the frequency domain, in the sense that *operators that commute before the discretization, also commute after the discretization.* An important computational implication of this is that *the derivative approximations can be computed directly from the smoothed grey-level values at different scales*, without any need for redoing the smoothing part of the operation, which is usually the computationally most expensive part when smoothed derivative approximations are computed. Another positive side effect is that a large number of *normalization conditions* concerning the sums of the filter coefficients are transferred to the discrete domain.

As a further support for the presented methodology, experimental results are provided in which these operations are used for a few different visual tasks. A straightforward edge-detection scheme is described. It is similar to Canny's method [6] but does not need any direct estimates of the gradient direction. Instead, zero crossings are detected in a certain polynomial expression in terms of derivatives up to order two, and tests are made on the sign of another polynomial expression in terms of derivatives up to order three (to eliminate false edges). Qualitively, the results obtained are similar to those of Canny, although the proposed scheme is given by a conceptually much simpler framework and, in addition, has the advantage that subpixel accuracy is obtained automatically. Then, it is illustrated how a *junction detector* can be straightforwardly implemented by detecting extremum regions in another polynomial expression in terms of derivatives up to order two.

The paper is organized as follows: Section 2 reviews scale-space theory for continuous and discrete signals. This constitutes the background material for the theoretical analysis of scale-space properties of different multiscale derivative approximations carried out in section 3. Section 4 describes some straightforward computational implications of the discrete theory presented, with comparisons to other possible discretizations of the continuous scale-space theory. Graphical illustrations of the resulting derivative approximation kernels are given in section 5, and section 6 gives experimental results of using the proposed representation for low-level feature extraction. Finally, section 7 gives a brief summary of the main results.

2 Background: Axiomatic Scale-Space Formulations

For a reader not familiar with the scale-space literature, the task of designing a multiscale signal representation may at first glance be regarded as somewhat arbitrary. Would it suffice to carry out just *any* type of smoothing operation? This is, however, not the case. It is crucially important in the construction of a scale-space representation that the transformation from a fine scale to a coarse scale really can be regarded as a simplification, so that fine-scale features disappear *monotonically* with increasing scale. If new artificial structures could be created at coarser scales not corresponding to important regions in the finer-scale representations of the signal, then it would be impossible to determine whether a feature at a coarse scale corresponded to a simplification of some coarse-scale structure from the original image or was just an accidental phenomenon, say, an amplification of the noise, *created by the smoothing method—not the data*. Therefore it is of utmost importance that artifacts should not be introduced by the smoothing transformation in going from a finer to a coarser scale.

This property has been formalized in different ways by different authors, all of them leading to a similar result, and there are now several theoretical results indicating that within the class of linear transformations, the scale-space formulation in terms of the diffusion equation describes the canonical way of constructing a multiscale image representation.

2.1 Continuous Signals

Koenderink [15] introduced the notion of *causality*, which means basically that new level curves must not be created when the scale parameter

is increased. In other words, it should always be possible to trace a grey-level value existing at a certain level of scale to a similar grey-level at any finer level of scale. He showed that these criteria, when combined with *homogeneity* and *isotropy* constraints (which essentially mean that all spatial points and all scale levels should be handled in a similar manner) by necessity and sufficiency lead to a formulation in terms of the diffusion equation, in both one and two dimensions. A similar result regarding the zero crossings of the Laplacian, based on slightly different assumptions, was given by Yuille and Poggio [35].

Babaud et al. [2] gave a particular proof in the one-dimensional case and showed that natural constraints on a smoothing kernel necessarily implied that the smoothing kernel had to be a Gaussian. Lindeberg [20] showed that a variation-diminishing property of *not introducing new local extrema* (or, equivalently, of not introducing new zero crossings) in the smoothed signal with increasing scale, combined with a *semigroup structure* on the family of convolution kernels and natural symmetry and normalization constraints, also, by necessity, leads to the Gaussian kernel. Recently, Florack et al. [11] have shown that the uniqueness of the Gaussian kernel for scale-space representation can be derived under weaker conditions by combining the semigroup structure of a convolution operation with a *uniform scaling property* over scales.

In this context it should be remarked that since the spatial derivatives computed from the scale-space representation satisfy the diffusion equation, it follows that they will exhibit the same scale-space properties over scales as does the smoothed grey-level signal. See also Koenderink and van Doorn [18], who showed that scale invariance applied to operators derived from the scale-space representation leads to the derivatives of the Gaussian kernels.

2.2 Discrete Signals

Concerning the discretization of this operation, Lindeberg [20], [21] has shown that the natural way to construct a *scale-space for discrete signals* is by convolution with a kernel termed *the discrete analogue of the Gaussian kernel* or, equivalently, by solving a certain semidiscretized version of the diffusion equation. In the one-dimensional case, the smoothing transformation is uniquely determined if it is assumed that the *number of local extrema* in the smoothed signal should not increase with scale and that the scale-space is to be generated by convolution with a *semigroup* of normalized and symmetric kernels having a *continuous* scale parameter. Given a discrete signal $f : \mathbb{Z} \to \mathbb{R}$, the scale-space representation $L : \mathbb{Z} \times \mathbb{R}_+ \to \mathbb{R}$ is defined by

$$L(x; t) = (T(\cdot; t) * f(\cdot))(x; t), \tag{9}$$

where $T : \mathbb{Z} \times \mathbb{R}_+ \to \mathbb{R}$ denotes the discrete analogue of the Gaussian kernel

$$T(n; t) = e^{-t} I_n(t) \tag{10}$$

and I_n are the modified Bessel functions of integer order, see, e.g., Abramowitz and Stegun [1]. This scale-space family can equivalently be defined from the solution of the semidiscretized version of the diffusion equation

$$\begin{aligned}(\partial_t L)(x; t) &= \frac{1}{2}(\nabla_3^2 L)(x; t) \\ &= \frac{1}{2}(L(x-1; t) \\ &\quad -L(x; t) + L(x+1; t)),\end{aligned} \tag{11}$$

with initial condition $L(\cdot; 0) = f(\cdot)$. Unfortunately, however, this formulation cannot be directly extended to higher dimensions, since in two (and higher) dimensions there are no non-trivial kernels never increasing the number of local extrema. Neither can the scale-invariance argument by Florack et al. [11] be applied in the discrete case; a perfectly scale-invariant operator cannot be defined on a discrete grid that has a certain preferred scale given by the distance between adjacent grid points.

For discrete signals of higher dimension a fruitful requirement turns out to be as follows: *if at some level a point is a local maximum (minimum), then its value must not increase (decrease) when the scale parameter increases*; see [20], [24]. In the continuous case this condition, which is similar to the maximum principle for parabolic differential equations, is equivalent to the causality requirement used by Koenderink [15] for

deriving the necessity of the continuous scale-space.

By assuming that the scale-space is generated by convolution with a *semigroup* of symmetric convolution kernels, which are continuous with respect to their *continuous scale parameter,* it can be shown that the above nonenhancement requirement of local extrema heavily restricts the class of smoothing transformations. Given a two-dimensional signal $f : \mathbb{Z}^2 \to \mathbb{R}$, the scale-space representation $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ must for some fixed $\gamma \in [0, 1]$ satisfy the semidiscretized version of the diffusion equation

$$\partial_t L = \frac{1}{2}\nabla_\gamma^2 L = \frac{1}{2}\left((1-\gamma)\nabla_5^2 L + \gamma\nabla_\times^2 L\right), \quad (12)$$

with initial condition $L(\cdot, \cdot; 0) = f$. Here ∇_5^2 and $\nabla_\times^2$ denote two common discrete approximations to the Laplacian operator, the five-point operator and cross operator, defined by (where the (temporary) notation $f_{-1,1}$ stands for $f(x-1, y+1)$, etc.).

$$(\nabla_5^2 f)_{0,0} = f_{-1,0} + f_{+1,0} + f_{0,-1} + f_{0,+1} - 4f_{0,0}, \quad (13)$$

$$(\nabla_{\times^2}^2 f)_{0,0} = \frac{1}{2}(f_{-1,-1} + f_{-1,+1} + f_{+1,-1} + f_{+1,+1} - 4f_{0,0}), \quad (14)$$

The parameter γ in undetermined by this theory. In references [20], [24] indications are given that it should not exceed the value $\gamma_{\max} = \frac{1}{2}$. Setting $\gamma = \gamma_{\text{symm}} = \frac{1}{3}$ gives a smoothing kernel with the highest rotational invariance or, rather, the least degree of rotational asymmetry in the Fourier domain. The corresponding smoothing kernel is separable if and only if $\gamma = \gamma_{\text{sep}} = 0$. Then, the scale-space family is obtained by applying the one-dimensional discrete analogue of the Gaussian kernel (10) along each dimension:

$$L(x, y; t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} T(m; t)T(n; t) \times f(x-m, y-n). \quad (15)$$

Let $T_\gamma : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ denote the convolution kernel describing the solution of (12) for any given value of γ. Then, T_γ obeys the cascade smoothing and semigroup properties

$$L(\cdot, \cdot; t_2) = T_\gamma(\cdot, \cdot; t_2 - t_1) * L(\cdot, \cdot; t_1), \quad (16)$$

$$T_\gamma(\cdot, \cdot; t_2) = T_\gamma(\cdot, \cdot; t_2 - t_1) * T_\gamma(\cdot, \cdot; t_1), \quad (17)$$

where $t_2 > t_1 \geq 0$. Similar results hold in one dimension as well as higher dimensions [24].

3. Derivative Approximations with Scale-Space Properties

3.1 Preliminaries

If we are to construct discrete analogues of derivatives at multiple scales and they are to possess scale-space properties in a discrete sense, what properties are then desirable? Let us start with the following very simple but quite useful observation,[5] whose proof follows directly from the commutative property of convolution transformations.

OBSERVATION 1 (Discrete convolution commutes with discrete scale-space smoothing). Given a discrete signal $f : \mathbb{Z}^2 \to \mathbb{R}$ in l_1 and a fixed value of $\gamma \in [0, 1]$, let $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be the discrete scale-space representation of f generated from (12), and let $\mathcal{D}$ be a linear and shift-invariant operator from l_1 to l_1 corresponding to discrete convolution with a kernel with finite support. Then, $\mathcal{D}$ commutes with a scale-space smoothing operator.

Concerning multiscale derivative approximations, this result has the following implications.

PROPOSITION 2 (Discrete derivative approximations: sufficiency (I)). Given a discrete signal $f : \mathbb{Z}^2 \to \mathbb{R}$ in l_1 and a fixed value of γ in $[0, 1]$, let $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be the discrete scale-space representation of f generated from

$$\partial_t L = \frac{1}{2}\left((1-\gamma)\nabla_5^2 L + \gamma\nabla_\times^2 L\right) = \frac{1}{2}\nabla_\gamma^2 L, \quad (18)$$

with initial condition $L(\cdot, \cdot; 0) = f(\cdot, \cdot)$. If a smoothed derivative $L_{x^i y^j}$ is defined as the result of applying a linear and shift-invariant operator $\mathcal{D}_{x^i y^j}$ to the smoothed signal L, i.e.,

$$L_{x^i y^j} = \mathcal{D}_{x^i y^j} L, \quad (19)$$

where $\mathcal{D}_{x^i y^j}$ corresponds to discrete convolution with a kernel of finite support, then the derivative approximation operator $\mathcal{D}_{x^i y^j}$ commutes with the scale-space smoothing operator

$$\begin{aligned}\mathcal{D}_{x^i y^j} L &= \mathcal{D}_{x^i y^j}(T * f) \\ &= (\mathcal{D}_{x^i y^j} T) * f \\ &= T * (\mathcal{D}_{x^i y^j} f).\end{aligned} \tag{20}$$

Moreover, the discrete derivative approximation obeys the cascade smoothing property

$$\begin{aligned}(\mathcal{D}_{x^i y^j} L)(\cdot, \cdot; t_2) = T(\cdot, \cdot; t_2 - t_1) * \\ (\mathcal{D}_{x^i y^j} L)(\cdot, \cdot; t_1) \quad (t_2 > t_1 \geq 0)\end{aligned} \tag{21}$$

and satisfies the semidiscretized version of the diffusion equation

$$\partial_t(\mathcal{D}_{x^i y^j} L) = \frac{1}{2} \nabla^2_\gamma (\mathcal{D}_{x^i y^j} L). \tag{22}$$

In particular, the discrete derivative approximation fulfills the following nonenhancement property of local extrema: if $(x_0, y_0) \in \mathbb{Z}^2$ is a local extremum point for the mapping $(x, y) \mapsto L_{x^i y^j}(x, y; t_0)$, then the derivative of $L_{x^i y^j}$ with respect to t in this point satisfies

$$\partial_t L_{x^i y^j}(x_0; t_0) \leq 0$$
if (x_0, y_0) is a local maximum point, (23)

$$\partial_t L_{x^i y^j}(x_0; t_0) \geq 0$$
if (x_0, y_0) is a local minimum point, (24)

i.e., it satisfies the scale-space conditions in the discrete case.

Proof. The validity of (20) follows directly from Observation 1, as does (21) if Observation 1 is combined with the cascade smoothing property (16) of the discrete scale-space. The validity of (22) can be derived by using $\mathcal{D}_{x^i y^j} L = T * (\mathcal{D}_{x^i y^j} f)$ from (20). Finally, (23) and (24) are direct consequences of the fact that because of (22) it holds that $\mathcal{D}_{x^i y^j} L$ is a scale-space representation of $\mathcal{D}_{x^i y^j} f$; see [20] or [24] for direct proofs.

In other words, *if a discrete derivative approximation is defined as the result of applying a convolution operator to the smoothed signal, then it will possess all the scale-space properties listed in section 1*, i.e., equations (1)–(8). Obviously, the derivative approximation should also be selected such that it in a numerical sense approximates the continuous derivative. A natural minimum requirement to pose is that the discrete operator $\mathcal{D}_{x^i y^j}$ should constitute a *consistent*[6] approximation of the continuous derivative operator.

3.2 Stating an Axiomatic Formulation

Given that the task is to state an axiomatic formulation of the first stages of visual processing, *the visual front end*, a list of desired properties can be made long, e.g., *linearity, translational invariance, mirror symmetry, semigroup, causality, continuity, differentiability, normalization* to one, nice *scaling* behaviour, *locality*, rapidly *decreasing* for large x and t, existence of an *infinitesimal generator*, and *invariance* with respect to certain *grey-level transformations*. Such a list will, however, be redundant, as is this one.

In this treatment a (minimal) subset of these properties will be taken as axioms. It will be assumed that the derivative approximation scheme should be generated by convolution with a one-parameter family of kernels possessing a cascade smoothing property and that the scale-space family should satisfy a nonenhancement property of local extrema. When these scale-space conditions are formulated, it turns out to be natural to express the nonenhancement property in terms of a sign condition on the derivative of the scale-space representation with respect to the scale parameter. To ensure differentiability in this step, a series of successive definitions and propositions will be stated that lead to the desired result.

By necessity, the details will be somewhat technical. The hasty reader may without loss of continuity proceed directly to Corollary 11, where a summary is given.

3.3 Necessity

The scale-space for discrete derivative approximations can be derived by postulating the following structure on the derivative approximation operators, which is similar to, but not equal to, the structure postulated on the smoothing

operation in the derivation of the traditional (zero-order) discrete scale-space representation in [20]; see also appendix A.1.

DEFINITION 3 (pre-scale-space family of derivative approximation kernels). A one-parameter family of kernels $D_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ is said to be a pre-scale-space family of (i, j)-derivative approximation kernels if

(i) $D_{x^i y^j}(\cdot, \cdot; 0)$ is a definite support kernel corresponding to a consistent discrete approximation of the derivative operator $\partial_{x^i y^j}$ and
(ii) $D_{x^i y^j}$ satisfies the cascade smoothing property

$$D_{x^i y^j}(\cdot, \cdot; t_2) = T(\cdot, \cdot; t_2 - t_1) * D_{x^i y^j}(\cdot, \cdot; t_1) \quad (t_2 \geq t_1 \geq 0) \quad (25)$$

for some family of kernels $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ in l_1 that, in turn, obeys

(a) the symmetry properties $T(-x, y; t) = T(x, y; t)$ and $T(y, x; t) = T(x, y; t)$ for all $(x, y) \in \mathbb{Z}^2$ and
(b) the continuity requirement $\|T(\cdot, \cdot; t) - \delta(\cdot, \cdot)\|_1 \to 0$ when $t \downarrow 0$.

DEFINITION 4 (Pre-scale-space representation of derivative approximations). Let $f : \mathbb{Z}^2 \to \mathbb{R}$ be a discrete signal in l_1, and let $D_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be a pre-scale-space family of (i, j)-derivative approximation kernels. Then, the one-parameter family of signals $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ given by

$$L_{x^i y^j}(x, y; t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} D_{x^i y^j}(m, n; t) f(x-m, y-n) \quad (26)$$

is said to be the pre-scale-space representation of (i, j)-derivative approximations of f generated by $D_{x^i y^j}$.

The linear type of smoothing is a consequence of the principle that the first stages of visual processing, the *visual front end*, should be as *uncommitted* as possible and should make no actual irreversible decisions. More technically, the linearity can also be motivated by requiring the discrete derivative approximations to obey linearity properties similar to those of the continuous derivatives.

The convolution type of smoothing and the symmetry requirements on T correspond to the assumption that in the absence of any information about what the image can be expected to contain, all spatial points should be treated in the same way, i.e., the smoothing should be *spatially shift invariant* and *spatially isotropic*.

The cascade form of smoothing and the continuity with respect to the continuous scale parameter reflect the properties that any coarse-scale representation should be computable from any fine-scale representation and that all scale levels should be treated similarly. In other words, there should be *no preferred scale*. It can be shown that these requirements imply that $L_{x^i y^j}$ is *differentiable* with respect to the scale parameter.

LEMMA 5 (Differentiability of derivative approximations). Let $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be a pre-scale-space representation of (i, j)-derivative approximations to a signal $f : \mathbb{Z}^2 \to \mathbb{R}$ in l_1. Then $L_{x^i y^j}$ satisfies the differential equation

$$\partial_t L_{x^i y^j} = \mathcal{A} L_{x^i y^j} \quad (27)$$

for some linear and shift-invariant operator $\mathcal{A}$.

Proof. Because of the cascade smoothing property of $D_{x^i y^j}$ we have that

$$T(\cdot, \cdot; t_2) * D_{x^i y^j}(\cdot, \cdot; 0) * f' = T(\cdot, \cdot; t_2 - t_1) * T(\cdot, \cdot; t_1) * D_{x^i y^j}(\cdot, \cdot; 0) * f' \quad (28)$$

and

$$(f' * D_{x^i y^j}(\cdot, \cdot; 0)) * (T(\cdot, \cdot; t_2) - T(\cdot, \cdot; t_2 - t_1) * T(\cdot, \cdot; t_1)) = 0 \quad (29)$$

hold for any $f' : \mathbb{Z}^2 \to \mathbb{R}$ and any $t_2 \geq t_1 \geq 0$. Hence $T(\cdot, \cdot; t_2) - T(\cdot, \cdot; t_2 - t_1) * T(\cdot, \cdot; t_1)$ will always be in the null space of $D_{x^i y^j}(\cdot, \cdot; 0)$ and we can with respect to the effect on $L_{x^i y^j}$ of $D_{x^i y^j}$ without loss of generality assume that T obeys the semigroup property $T(\cdot, \cdot; t_2) = T(\cdot, \cdot; t_2 -$

$t_1) * T(\cdot, \cdot; t_1)$. This means that $L_{x^i y^j}$ is a *pre-scale-space representation* of $D_{x^i y^j}(\cdot, \cdot; 0) * f$ (see Definition 13 in appendix A.1). According to Lemma 14 in appendix A.1, it follows that $L_{x^i y^j}$ satisfies (27).

This property makes it possible to formulate the nonenhancement property for local extrema in terms of derivatives of the scale-space representation with respect to the scale parameter. In every local maximum point the grey-level value is required not to increase, and in every local minimum point the value is required not to decrease.

DEFINITION 6 (Pre-scale-space properties). A differentiable one-parameter family of signals $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ is said to possess pre-scale-space properties or, equivalently, not to enhance local extrema, if for every value of the scale parameter $t_0 \in \mathbb{R}_+$ it holds that if $(x_0, y_0) \in \mathbb{Z}^2$ is a local extremum[7] point for the mapping $(x, y) \mapsto L_{x^i y^j}(x, y; t_0)$, then the derivative of $L_{x^i y^j}$ with respect to t in this point satisfies

$$(\partial_t L_{x^i y^j})(x_0, y_0; t_0) \leq 0 \text{ if } (x_0, y_0) \text{ is a (weak) local maximum point,} \quad (30)$$

$$(\partial_t L_{x^i y^j})(x_0, y_0; t_0) \geq 0 \text{ if } (x_0, y_0) \text{ is a (weak) local minimum point.} \quad (31)$$

Given these properties, a pre-scale-space family of derivative approximation kernels is regarded as a scale-space family of derivative approximation kernels if it satisfies the pre-scale-space properties just described for *any* input signal.

DEFINITION 7 (Scale-space family of derivative approximation kernels). A one-parameter family of pre-scale-space (i, j)-derivative approximation kernels $D_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ is said to be a scale-space family of (i, j)-derivative approximation kernels if for any signal $f : \mathbb{Z}^2 \to \mathbb{R} \in l_1$ the pre-scale-space representation of (i, j)-derivative approximations to f generated by $D_{x^i y^j}$ obeys the nonenhancement property stated in Definition 6, i.e., if for any signal $f \in l_1$ local extrema in $L_{x^i y^j}$ are never enhanced.

DEFINITION 8 (Scale-space representation of derivative approximations). Let $f : \mathbb{Z}^2 \to \mathbb{R}$ be a discrete signal in l_1, and let $D_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be a family of scale-space (i, j)-derivative approximation kernels. Then, the pre-scale-space representation of (i, j)-derivative approximations $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ of f is said to be a scale-space representation of (i, j)-derivative approximations to f.

From these definitions it can be shown that the structure of the scale-space representation is determined up to two arbitrary constants and that $L_{x^i y^j}$ must satisfy a semidiscretized version of the diffusion equation.

THEOREM 9 (Discrete derivative approximations: necessity). A scale-space representation of (i, j)-derivative approximations $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ to a signal $f : \mathbb{Z}^2 \to \mathbb{R}$ satisfies the differential equation

$$\partial_t L_{x^i y^j} = \alpha \nabla_5^2 L_{x^i y^j} + \beta \nabla_\times^2 L_{x^i y^j}, \quad (32)$$

with initial condition $L_{x^i y^j}(\cdot, \cdot; 0) = D_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot)$ for some constants $\alpha \geq 0$ and $\beta \geq 0$ and some finite support kernel $D_{x^i y^j}$.

Proof. See appendix A.2.

THEOREM 10 (Discrete derivative approximations: sufficiency (II)). Let $f : \mathbb{Z}^2 \to \mathbb{R}$ be a discrete signal in l_1, and let $L_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be the representation generated by the solution to the differential equation

$$\partial_t L_{x^i y^j} = \alpha \nabla_5^2 L_{x^i y^j} + \beta \nabla_\times^2 L_{x^i y^j} \quad (33)$$

with initial condition $L_{x^i y^j}(\cdot, \cdot; 0) = D_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot)$ for some fixed $\alpha \geq 0$ and $\beta \geq 0$ and some finite support kernel $D_{x^i y^j}(\cdot, \cdot; 0)$, corresponding to a consistent approximation to the derivative operator $\partial_{x^i y^j}$. Then, $L_{x^i y^j}$ is a scale-space representation of (i, j)-derivative approximations to f.

Proof. See appendix A.3.

By reparametrizing $\alpha = C(1 - \gamma)$ and $\beta = C\gamma$ (where $\gamma \in [0, 1]$) and without loss of generality (linearly) transforming the scale parameter t

such that $C = 1/2$, it follows that the necessity and sufficiency results can be summarized in the following way.

COROLLARY 11 (Derivative approximations preserving scale-space properties). Within the class of linear transformations of convolution type that obey the cascade smoothing property, a multiscale representation of discrete derivative approximations $L_{x^iy^j}$ of a signal f, satisfying

$$L_{x^iy^j}(\cdot, \cdot; 0) = \mathcal{D}_{x^iy^j} * f \tag{34}$$

for some finite support convolution operator $\mathcal{D}_{x^iy^j}$, possesses scale-space properties in the discrete sense if and only if it is defined as the result of applying $\mathcal{D}_{x^iy^j}$, to the scale-space representation of f at any scale, i.e., if and only if $L_{x^iy^j}$ is defined as

$$L_{x^iy^j} = \mathcal{D}_{x^iy^j} L = \mathcal{D}_{x^iy^j}(T_\gamma * f), \tag{35}$$

where T_γ denotes the discrete analogue of the Gaussian kernel defined as the kernel describing the solution to (12) for a fixed $\gamma \in [0, 1]$. Equivalently, the derivative approximation possesses discrete scale-space properties if and only if it, for some fixed $\gamma \in [0, 1]$ and for some linear transformation of the scale parameter ($t = \alpha' t'$, where $\alpha' > 0$), satisfies the semidiscretized version of the diffusion equation

$$\partial_t L_{x^iy^j} = \frac{1}{2}\nabla^2_\gamma L_{x^iy^j} \tag{36}$$

with initial condition
$L_{x^iy^j}(\cdot, \cdot; 0) = D_{x^iy^j}(\cdot, \cdot; 0) * f$.

The result has been expressed in a general form to indicate that similar results hold in the one-dimensional case as well as in higher dimensions. (For example, in the one-dimensional case the operator ∇^2_γ is replaced by $\nabla^2_3 L$). Now what remains is to *define*[8] how such derivative approximations are to be computed within the given class of operations.

3.4 One-Dimensional Signals

In the one-dimensional case it is natural to define the discrete correspondence to the derivative operator ∂_x as the first-order difference operator δ_x. This gives

$$\begin{aligned}(\mathcal{D}_x L)(x; t) &= (\delta_x L)(x; t) \\ &= \frac{1}{2}(L(x + 1; t) - L(x - 1; t))\end{aligned} \tag{37}$$

and the striking similarity between the discrete and continuous relations,

$$\begin{aligned}(\delta_x T)(x; t) &= -\frac{x}{t}T(x; t), \\ (\partial_x G)(x; t) &= -\frac{x}{t}G(x; t).\end{aligned} \tag{38}$$

Similarly, it is natural to define the discrete correspondence of the second-order derivative operator ∂_{xx} as the second-order difference operator δ_{xx} given by

$$\begin{aligned}(\mathcal{D}_{xx} L)(x; t) &= (\delta_{xx} L)(x; t) \\ &= L(x + 1; t) - 2L(x; t) \\ &\quad + L(x - 1; t).\end{aligned} \tag{39}$$

From the diffusion equation it follows that the following relations are satisfied:

$$\begin{aligned}(\delta_{xx} T)(x; t) &= 2(\partial_t T)(x; t), \\ (\partial_{xx} g)(x; t) &= 2(\partial_t g)(x; t).\end{aligned} \tag{40}$$

Note, however, the clear problem in this discrete case[9]:

$$\delta_x \delta_x \neq \delta_{xx}. \tag{41}$$

Difference operators of higher order can be defined in an analogous manner:

$$\begin{aligned}\mathcal{D}_{x^{2n}} &= \delta_{x^{2n}} = (\delta_{xx})^n, \\ \mathcal{D}_{x^{2n+1}} &= \delta_{x^{2n+1}} = \delta_x \delta_{x^{2n}},\end{aligned} \tag{42}$$

which means that the derivative approximations of different order are related by

$$L_{x^{n+2}} = \delta_{xx} L_{x^n}, \quad L_{x^{2n+1}} = \delta_x L_{x^{2n}}. \tag{43}$$

3.5 Two-Dimensional Signals

For two-dimensional signals it is natural to let the definitions of the derivative approximations depend on the value of γ.

3.5.1 Separable case. In the separable case $\gamma_{\text{sep}} = 0$ it is natural to inherit the definitions from the one-dimensional case:

$$\mathcal{D}_{x^i y^j} = \delta_{x^i y^j} = \delta_{x^i} \delta_{y^j}, \tag{44}$$

where the operator δ_{y^j} should be interpreted as a difference operator in the y-direction that is similar to δ_{x^i} in the x-direction. This gives

$$\nabla^2_\gamma = \nabla^2_5 = \delta_{xx} + \delta_{yy}. \tag{45}$$

If T^x, $T^y : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ are defined as the two-dimensional kernels corresponding to convolution with the one-dimensional discrete analogue of the Gaussian kernel $T : \mathbb{Z} \times \mathbb{R}_+ \to \mathbb{R}$ along the x- and y-directions, respectively, then the effect of the derivative approximation method can be written

$$\begin{aligned} L_{x^i y^j} &= \delta_{x^i y^j} (T_{\gamma=0} * f) \\ &= \delta_{x^i} \delta_{y^j} (T^x * T^y * f) \\ &= (\delta_{x^i} T^x) * (\delta_{y^j} T^y) * f, \end{aligned} \tag{46}$$

which implies that a (two-dimensional) derivative approximation $L_{x^i y^j}$ of order $i + j$ exactly corresponds to the result of applying a (one-dimensional) derivative approximation kernel $\delta_{x^i} T^x$ of order i along the x-direction and a (one-dimensional) derivative approximation kernel $\delta_{y^j} T^y$ of order j along the y-direction.

3.5.2 Rotationally symmetric case. The case $\gamma_{\text{symm}} = \frac{1}{3}$ corresponds to approximating the continuous Laplacian with the discrete nine-point operator[10] ∇^2_9 having the computational molecule

$$\nabla^2_9 = \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 2/3 & -10/3 & 2/3 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}. \tag{47}$$

If it is assumed that the second-derivative-approximation operators should be symmetric and satisfy[11] $\widetilde{\delta}_{xx} + \widetilde{\delta}_{yy} = \nabla^2_9$, it is natural to assume that, for some values a and b, $\widetilde{\delta}_{xx}$ and $\widetilde{\delta}_{yy}$ can be represented by 3×3 computational molecules of the form

$$\widetilde{\delta}_{xx} = \begin{pmatrix} a & -2a & a \\ b & -2b & a \\ a & -2a & a \end{pmatrix},$$

$$\widetilde{\delta}_{yy} = \begin{pmatrix} a & b & a \\ -2a & -2b & -2a \\ a & b & a \end{pmatrix}. \tag{48}$$

The condition $\widetilde{\delta}_{xx} + \widetilde{\delta}_{yy} = \nabla^2_9$ then gives

$$\widetilde{\delta}_{xx} = \begin{pmatrix} 1/12 & -1/6 & 1/12 \\ 5/6 & -5/3 & 5/6 \\ 1/12 & -1/6 & 1/12 \end{pmatrix},$$

$$\widetilde{\delta}_{yy} = \begin{pmatrix} 1/12 & 5/6 & 1/12 \\ -1/6 & -5/3 & -1/6 \\ 1/12 & 5/6 & 1/12 \end{pmatrix}. \tag{49}$$

We leave the question of how to define the other operators, $\mathcal{D}_x$, $\mathcal{D}_y$, and $\mathcal{D}_{xy}$, open.

3.6 Other Possible Choices

Concerning the choice of these operators, it should be remarked that these (in principle, arbitrary) selections[12] were based on the relations to standard discrete operators used in numerical analysis; see, e.g., Dahlquist et al. [8]. Other design criteria may lead to other operators; see, e.g., Haralick [13], Meer and Weiss [29], and Vieville and Faugeras [33]. Nevertheless, the *algebraic* scale-space properties are preserved whatever linear operators are used.

4 Computational Implications

4.1 Derivative Approximations Directly from Smoothed Grey-Level Data

An immediate consequence of the proposed scale-space representation of discrete derivative approximations is that the derivative approximations can be computed *directly* from the smoothed grey-level values at different scales and that this will (up to numerical truncation and rounding errors) give *exactly* the same result as convolving the signal with the discrete analogue to the Gaussian-derivative kernel. This has a clear advantage in terms of computational efficiency, since the derivative approximation operators $\mathcal{D}_{x^i y^j}$ usually have a small support region and contain a small number of nonzero filter coefficients. Hence, there is absolutely no need

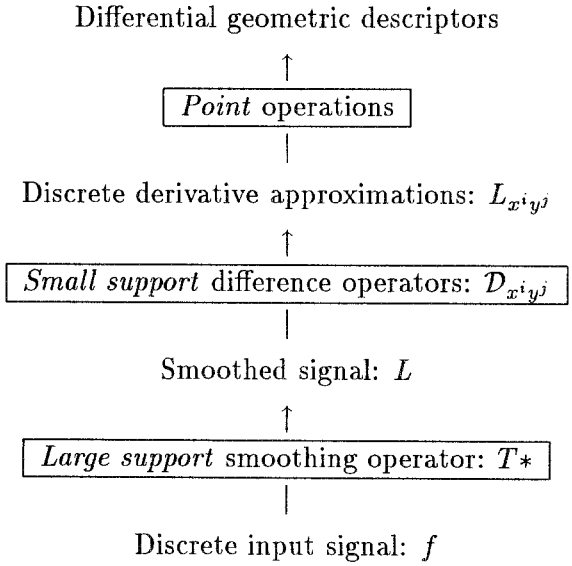

Fig. 1. Schematic overview of the different types of computations required for computing multiscale derivative approximations and discrete approximations to differential geometric descriptors by using the proposed framework.

for *redoing* the smoothing part of the transformation, as is the case if *several* derivative approximations are computed by convolution with smoothed derivative filters, e.g., some discrete approximations to the Gaussian derivatives.[13] This issue is of particular importance in computing multiscale differential geometric descriptors of high derivation order; see figure 1.

4.2 Normalization of the Filter Coefficients

This subsection lists a number of continuous relations and their discrete correspondences. When the scale parameter tends to zero, the continuous and discrete Gaussian kernels tend to the continuous and discrete delta functions, δ_{cont} and δ_{disc}, respectively:

$$\lim_{t \downarrow 0} T(x; t) = \delta_{\text{disc}}(x),$$
$$\lim_{t \downarrow 0} g(x; t) = \delta_{\text{cont}}(x). \tag{50}$$

Concerning the normalization of the filter coefficients, it holds that

$$\sum_{n=-\infty}^{\infty} T(n; t) = 1, \quad \int_{\xi=-\infty}^{\infty} g(\xi; t) d\xi = 1, \tag{51}$$

which means that in the discrete case the sum of the smoothing coefficients is exactly one. Similarly, the sum of the filter coefficients in a derivative approximation kernel is exactly zero, i.e.,

$$\sum_{n=-\infty}^{\infty} (\delta_{x^n} T)(n; t) = 0,$$
$$\int_{\xi=-\infty}^{\infty} (\partial_{x^n} g)(\xi; t) d\xi = 0, \tag{52}$$

for any integer $n \geq 1$. A trivial consequence of this is that the sum of the filter coefficients in the discrete Laplacian of the Gaussian is exactly zero and there is no need for modifying the filter coefficients, as has been the case in previous implementations of this operator.

In some situations it is useful to normalize the kernels used for derivative computations so that the integral of positive parts remains constant over scales. Such kernels have been used in edge detection, by, for example, Korn [19] and Zhang and Bergholm [36], for automatic scale selection in feature detection by Lindeberg [26], and in shape from texture by Lindeberg and Gårding [27]. Then, the following relations are useful:

$$\int_{\xi=0}^{\infty} (\partial_x g)(\xi; t) d\xi = -g(0; t) = \frac{1}{\sqrt{2\pi t}},$$
$$\sum_{n=0}^{\infty} (\delta_{x^-} T)(n; t) = -T(0; t). \tag{53}$$

In practice, to give an effect equivalent to normalizing the kernels such that the sum of the positive values is always equal to one, it is, of course, sufficient to divide $\delta_{x^-} L$ and $\delta_{y^-} L$ by $T(0; t)$. Note that with increasing t this correction factor asymptotically approaches the corresponding normalization factor in the continuous case, $1/g(0; t) = \sqrt{2\pi t}$, whereas at $t = 0$ the discrete normalization factor is exactly one, in contrast to the continuous normalization factor, which is then zero; see also section 4.4.

4.3 Comparisons with Other Discretizations

As a comparison with other possible approaches, observe that if continuously equivalent expressions are discretized in a straightforward way, say, by approximating the convolution integral

by using the rectangle rule of integration and by approximating the derivative operator $\partial_{x^i y^j}$ with the difference operator $\delta_{x^i y^j}$, then the discretized results will depend on which actual expression in selected. Consider, for example, the three equivalent expressions in (5), where

(i) the discretization of the left expression corresponds to discrete convolution with the *sampled Gaussian* kernel followed by the application of a *difference* operator,

(ii) the discretization of the central expression corresponds to discrete convolution with the *sampled derivative* of the Gaussian kernel, and

(iii) the discretization of the right expression corresponds to the application of the central *difference* operator to the signal followed by discrete convolution with the *sampled Gaussian* kernel.

It is clear that the equivalence is violated: (i) and (iii) describe equivalent operations, whereas (i) and (ii) do not. In the particular case of the Laplacian of the Gaussian, $\nabla^2(g * f)$, it is well known that this kernel is not separable. In performing the computations in the spatial domain, the fact that g satisfies the diffusion equation, $\nabla^2(g * f) = 2\partial_t(g * f)$, is sometimes used for reducing the computational work by computing $g * f$ at two adjacent scales, forming the difference, and then dividing by the scale difference. In the literature this method is usually referred to as the difference-of-Gaussians (DOG) approach; see, e.g., Marr and Hildreth [28]. Note, however, that when the scale difference tends to zero, the result of this operation is not guaranteed to converge to the actual result of, say, convolving the original signal with the sampled Laplacian of the Gaussian – not even if the calculations (of the spatially sampled data) are represented with infinite precision in the grey-level domain.

For the proposed discrete framework, on the other hand, the discrete analogues of these entities are *exactly equivalent*; see (21) and (22). The main reason why "the discrete scale-space representations" generated from different versions of the sampled Gaussian kernel do not possess discrete scale-space properties is that when this discretization method is used, the discrete correspondences to operators, which commute before the discretization, do not *commute after the discretization*.

4.4 Discrete Modeling of Feature Detectors

The proposed discrete kernels are also suitable for discrete modeling of feature detectors. As an example of this, consider the *diffuse step edges* studied by Zhang and Bergholm [36]. In the continuous case, the intensity profile perpendicular to such a (straight and unit height) edge may be modeled by

$$\phi(x; t_0)(x) = \int_{\xi=-\infty}^{x} g(\xi; t_0) d\xi, \qquad (54)$$

where t_0 describes the degree of diffuseness. In a scale-space representation, L_{t_0}, of $\phi(\cdot; t_0)$ the variation over scales of the gradient magnitude, computed at the origin and normalized as described in section 4.2, is given by

$$(\partial_x L_{t_0})(0; t) \sim \frac{g(0; t_0 + t)}{g(0; t)}. \qquad (55)$$

If a corresponding *discrete diffuse step edge* is defined by

$$\phi(x; t_0)(x) = \sum_{n=-\infty}^{x} T(n; t_0) \qquad (56)$$

and if the gradient is approximated by the backward-difference operator δ_{x^-}, then the analytic expression for the corresponding discrete analogue of the gradient magnitude will be algebraically similar to that in the continuous case,

$$(\delta_{x^-} L_{t_0})(0; t) \sim \frac{T(0; t_0 + t)}{T(0; t)}. \qquad (57)$$

Moreover, it is easy to show that when t and t_0 increase, $(\delta_{x^-} L_{t_0})(0; t)$ approaches $\sqrt{t}/\sqrt{t_0 + t}$, which agrees with the continuous expression for $(\partial_x L_{t_0})(0; t)$ obtained from (55); see also [36].

5 Kernel Graphs

Figure 2 illustrates the differences and similarities between the proposed discrete kernels and

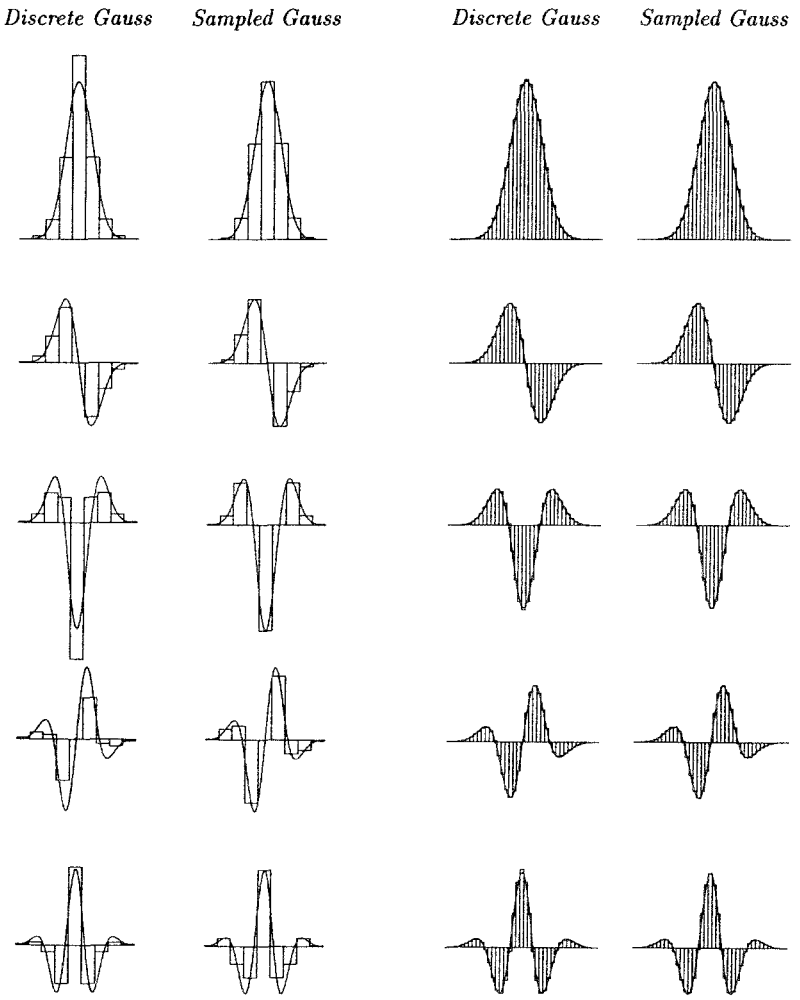

Fig. 2. Comparisons of the discrete analogue of the Gaussian kernel and the sampled Gaussian kernel at scale levels $t = 1.0$ and $t = 16.0$. The columns show from left to right the discrete Gaussian $t = 1.0$, the sampled Gaussian $t = 1.0$, the discrete Gaussian $t = 16.0$, and the sampled Gaussian $t = 16.0$. The derivative/difference order increases from top to bottom; the upper row shows the raw smoothing kernel; the first-, second-, third-, and fourth-order derivative/difference kernels follow. The block diagrams indicate the discrete kernels, and the smooth curve indicates the continuous Gaussian.

the derivatives of the continuous Gaussian kernel. Shown are the graphs of

$$T_{x^n} = \delta_{x^n} T \quad \text{and} \quad g_{x^n} = \partial_{x^n} g \qquad (58)$$

for a few orders of derivatives/differences and at two different scales. These kernels describe the equivalent effect of computing smoothed derivatives in the one-dimensional case as well as the separable two-dimensional case; see (46). For comparison, the equivalent discrete kernels corresponding to sampling the derivatives of the Gaussian kernel are displayed. It can be seen that the difference is largest at fine scales and that it decreases as the kernels approach each other at coarser scales.

Figure 3 shows the corresponding two-dimensional kernels represented by grey-level values, together with examples of equivalent directional derivatives corresponding to pointwise linear combinations of the components of $L_{x^i y^j}$, calculated by using the well-known expression for the nth-order directional derivative ∂_α^n of a function L in any direction α,

$$\partial_\alpha^n L = (\cos\alpha\, \partial_x + \sin\alpha\, \partial_y)^n L. \qquad (59)$$

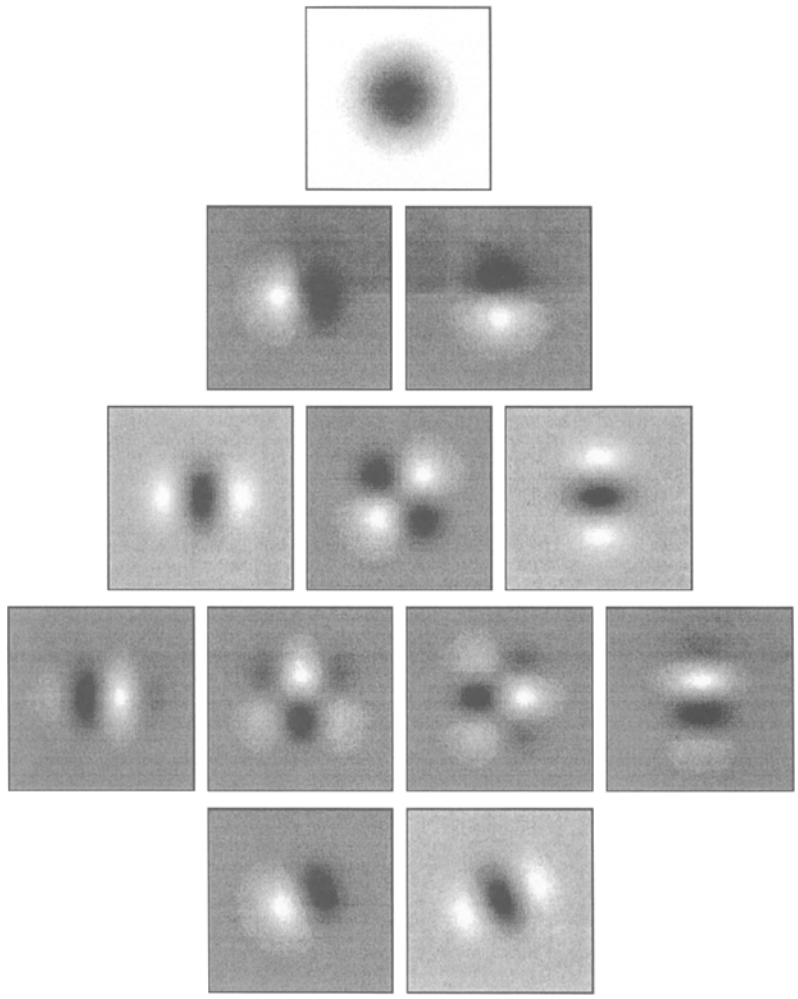

Fig. 3. Grey-level illustrations of the *equivalent* two-dimensional discrete derivative approximation kernels up to order three (in the separable case corresponding to $\gamma = 0$): (row 1) zero-order smoothing kernel T (inverted); (row 2) first-order-derivative approximation kernels $\delta_x T$ and $\delta_y T$; (row 3) second-order-derivative approximation kernels $\delta_{xx} T$, $\delta_{xy} T$, $\delta_{yy} T$; (row 4) third-order-derivative approximation kernels $\delta_{xxx} T$, $\delta_{xxy} T$, $\delta_{xyy} T$, $\delta_{yyy} T$; (row 5) first- and second-order directional-derivative approximation kernels in the direction 22.5° computed from (59). The scale level is $t = 64.0$, and the image size is 127 × 127 pixels.

In the terminology of Freeman and Adelson [12] and Perona [32], these kernels are trivially "steerable" (as is the directional derivative of any continuously differentiable function).

Figure 4 gives an illustration of what might happen if the sampling problems at fine scales are not properly treated. It shows a situation in which the slope at the origin of the sampled (fifth-order) derivative of the Gaussian kernel is reversed. If this kernel is used for derivative estimation, then even the sign of the derivative can be wrong. For the corresponding discrete kernel, however, the qualitative behaviour is correct.

6 Experimental Results: Low-Level Feature Extraction

To demonstrate how the proposed framework can be used in low-level feature extraction, this section will briefly describe how a subpixel edge

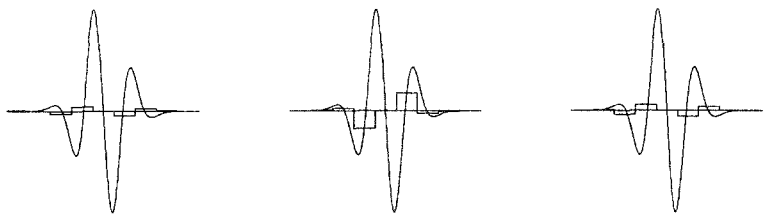

Fig. 4. Different discrete approximations of the fifth-order derivative of the Gaussian at a fine scale ($t = 0.46$): (left) fifth-order difference of the discrete analogue of the Gaussian kernel, (middle) sampled fifth-order derivative of the Gaussian kernel, and (right) fifth-order difference of the sampled Gaussian kernel. Observe that the slope at the origin of the kernel in the middle curve differs from the slopes of the left and right ones. This means that if this filter is used for derivative approximations, then, in general, even the sign of the derivative estimate may be wrong.

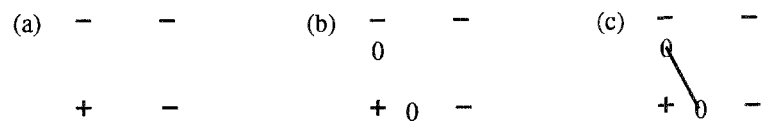

Fig. 5. The subpixel edge interpolation procedure in a simple situation, a four-point cell in which $\tilde{L}_{\tilde{v}\tilde{v}}$ is positive at one point and negative at all the other ones, whereas $\tilde{L}_{\tilde{v}\tilde{v}\tilde{v}}$ is negative at all points: (a) sign pattern for $L_{\tilde{v}\tilde{v}}$, (b) estimated locations of zero crossings from linear interpolation, and (c) zero crossings connected by a straight line segment (+ denotes a pixel for which $\tilde{L}_{\tilde{v}\tilde{v}} > 0$ and − denotes a pixel for which $\tilde{L}_{\tilde{v}\tilde{v}} < 0$).

detector can be easily formulated in terms of zero crossings of a certain polynomial expression in these derivative approximations and how a junction detector can be expressed in terms of maxima of another such expression. (In a related work [27], use of these discrete derivative approximations as a basis for shape from texture is developed.)

6.1 Edge Detection Based on Local Directional Derivatives

A natural way to define edges from a continuous grey-level image $L : \mathbb{R}^2 \to \mathbb{R}$ is as the set of points for which the gradient magnitude assumes a maximum in the gradient direction. This method is usually referred to as "non-maximum suppression"; see, e.g., Canny [6] or Korn [19].

6.1.1 Differential geometric definition of edges in the continuous case. To give a differential definition of this concept, introduce a curvilinear coordinate system (u, v) such that at every point the v-direction is parallel to the gradient direction of L and at every point the u-direction is perpendicular. Moreover, at any point $P = (x, y) \in \mathbb{R}^2$ let $\partial_{\tilde{v}}$ denote the directional derivative operator in the gradient direction of L at P and $\partial_{\tilde{u}}$ denote the directional derivative in the perpendicular direction. Then, at any $P \in \mathbb{R}^2$ the gradient magnitude is equal to $\partial_{\tilde{v}} L$, denoted $L_{\tilde{v}}$, at that point. If it is assumed that the second- and third-order directional derivatives of L in the v-direction are not simultaneously zero, the condition for P_0 to be a gradient maximum in the gradient direction may be stated as

$$\begin{cases} L_{\tilde{v}\tilde{v}} = 0, \\ L_{\tilde{v}\tilde{v}\tilde{v}} < 0. \end{cases} \tag{60}$$

By expressing the directional derivatives in terms of derivatives with respect to the Cartesian coordinates (x, y), $\partial_{\tilde{u}} = \sin\phi\partial_x - \cos\phi\partial_y$, $\partial_{\tilde{v}} = \cos\phi\partial_x + \sin\phi\partial_y$, where $(\cos\phi, \sin\phi)$ is the normalized gradient direction of L at P_0, and by taking the numerators of the resulting expressions, this condition can be expressed as

$$\begin{aligned} \tilde{L}_{\tilde{v}\tilde{v}} &= L_{\tilde{v}}^2 L_{\tilde{v}\tilde{v}} \\ &= L_x^2 L_{xx} + 2L_x L_y L_{xy} + L_y^2 L_{yy} = 0, \end{aligned} \tag{61}$$

$$\begin{aligned} \tilde{L}_{\tilde{v}\tilde{v}\tilde{v}} &= L_{\tilde{v}}^3 L_{\tilde{v}\tilde{v}\tilde{v}} \\ &= L_x^3 L_{xxx} + 3L_x^2 L_y L_{xxy} \\ &\quad + 3L_x L_y^2 L_{xyy} + L_y^3 L_{yyy} \\ &< 0, \end{aligned} \tag{62}$$

where $L_{\tilde{v}} = (L_x^2 + L_y^2)^{1/2}$. If L is reinterpreted as the scale-space representation of a signal f, it follows that the edges in f at any scale t can be defined as the points on the zero-crossing curves of the numerator of $\tilde{L}_{\tilde{v}\tilde{v}}$ for which $\tilde{L}_{\tilde{v}\tilde{v}\tilde{v}}$ is strictly negative. Note that with this formulation there is no need for any explicit estimate of the gradient direction. Moreover, there are no specific assumptions about the shape of the intensity profile perpendicular to the edge.

6.1.2 Discrete approximation and interpolation scheme. Given discrete data, we propose that the derivatives $L_{x^i y^j}$ can be estimated by using the discrete derivative approximation in subsection 6.1.1. From these, in turn, discrete approximations to $\tilde{L}_{\tilde{v}\tilde{v}}$ and $\tilde{L}_{\tilde{v}\tilde{v}\tilde{v}}$ can be computed as *pointwise* polynomials.

Of course, there do not exist any exact discrete correspondences to zero-crossing curves in the discrete case. Nevertheless, a *subpixel* edge detector can be defined by interpolating for zero crossings in the discrete data. A natural way of implementing this is by, for every four-cell in the image, $\{(x, y), (x + 1, y), (x, y + 1), (x + 1, y + 1)\}$, performing a two-step linear interpolation. The idea is basically as follows. For any pair of (four-) adjacent points having opposite signs of $\tilde{L}_{\tilde{v}\tilde{v}}$, introduce a zero-crossing point on the line between, with the location set by linear interpolation. Then, connect any pair of such points within the same four-cell by a line segment; see figure 5(a). Performing this operation on all four-cells in an image gives edge segments that can be easily linked[14] into polygons by an edge tracker and, by definition, will be continuous in space.

6.1.3 Experimental Results. Figure 6 displays an example of applying this edge-detection scheme to an image of a table scene at a number of different scales, while figure 7 shows a simple comparison with a traditional implementation of the Canny–Deriche edge detector [9]. Of course, it is not easy to make a fair comparison of the two methods, since the Canny–Deriche method is pixel oriented and uses a different smoothing filter. Moreover, the usefulness of the output is ultimately determined by the algorithms that use the output from this operation as input and can hardly be measured in isolation.

Finally, it should be remarked that the main purpose of the experiment is not to show that this edge detection method is entirely new; rather, it is to demonstrate, firstly, that useful results can be obtained by using the proposed derivative approximations[15] up to order three and, secondly, that by using the scheme in figure 1, results qualitatively comparable to a state-of-the-art detector can be obtained by very simple means.

6.2 Junction Detection

An entity commonly used for junction detection is the curvature of level curves in intensity data; see, e.g., Kitchen and Rosenfeld [14] or Koenderink and Richards [17]. In terms of derivatives of the intensity function with respect to the (x, y)-, and (u, v)-coordinates, respectively, it can be expressed as

$$\begin{aligned} \kappa &= \frac{L_y^2 L_{xx} - 2L_x L_y L_{xy} + L_x^2 L_{yy}}{(L_x^2 + L_y^2)^{3/2}} \\ &= \frac{L_{\tilde{u}\tilde{u}}}{L_{\tilde{v}}}. \end{aligned} \tag{63}$$

To give a stronger response near edges, this entity is usually multiplied by the gradient magnitude raised to some power, n. A natural choice is $n = 3$. This leads to polynomial expression. Moreover, the resulting operator becomes skew invariant; see Blom [3].

$$\begin{aligned} \tilde{\kappa} &= L_{\tilde{v}}^3 \kappa = L_{\tilde{v}}^2 L_{\tilde{u}\tilde{u}} \\ &= L_y^2 L_{xx} - 2L_x L_y L_{xy} + L_x^2 L_{yy}. \end{aligned} \tag{64}$$

Figure 6 displays the result of computing this *rescaled level curve curvature* at a number of different scales and then taking the absolute value in every point. To enhance the maxima in the output, a certain type of blob detection, *grey-level blob* detection [21], [22], has been applied

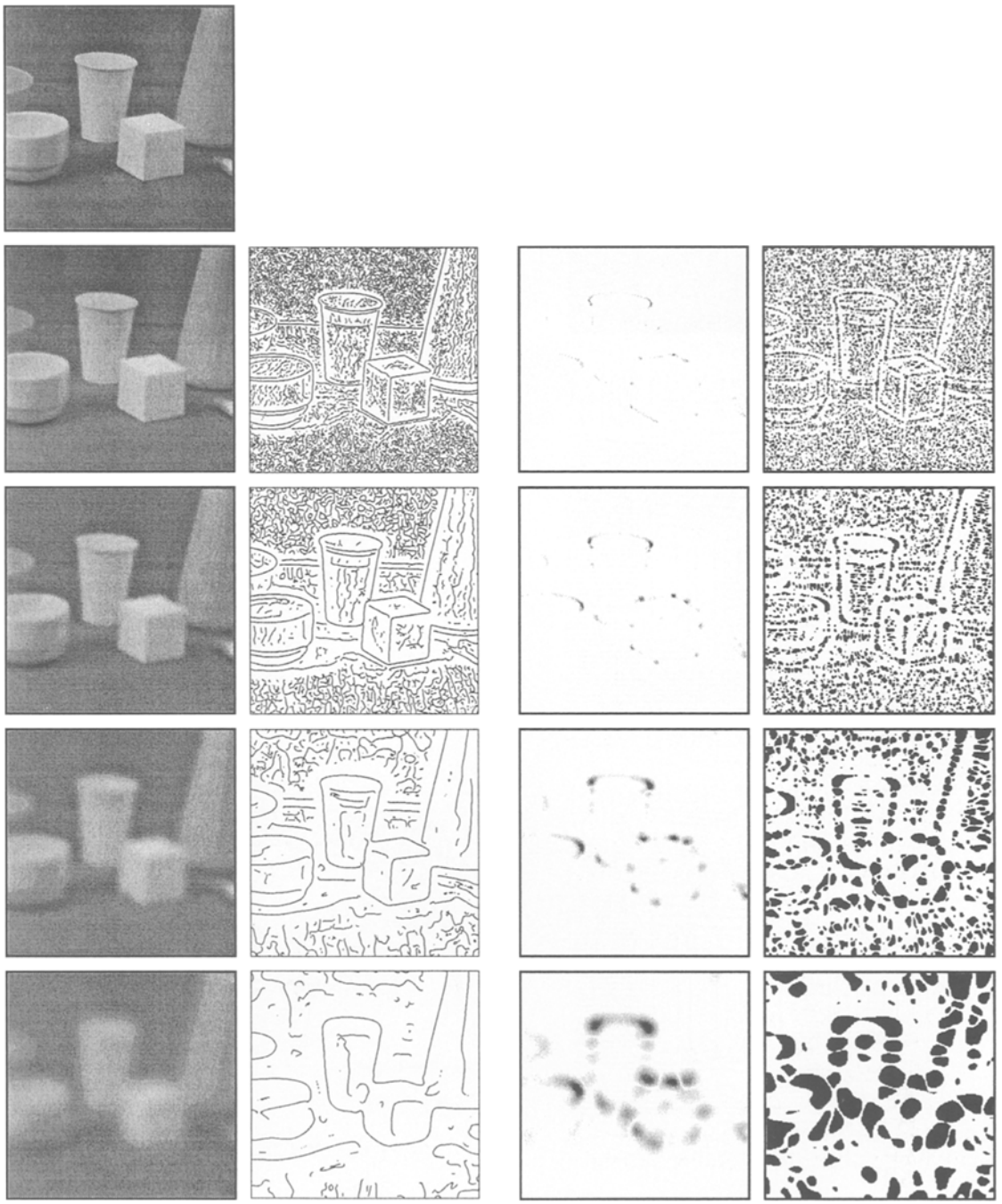

Fig. 6. Edges and junctions detected by applying the presented subpixel edge-detection scheme and by computing (the absolute value of) the rescaled level curve curvature at scales $t = 1, 4, 16$, and 64, respectively (from top to bottom). The columns show from (left to right) the smoothed grey-level values, the detected edges, the rescaled level curve curvature, and the grey-level blobs extracted from the curvature data. (No thresholding has been performed.)

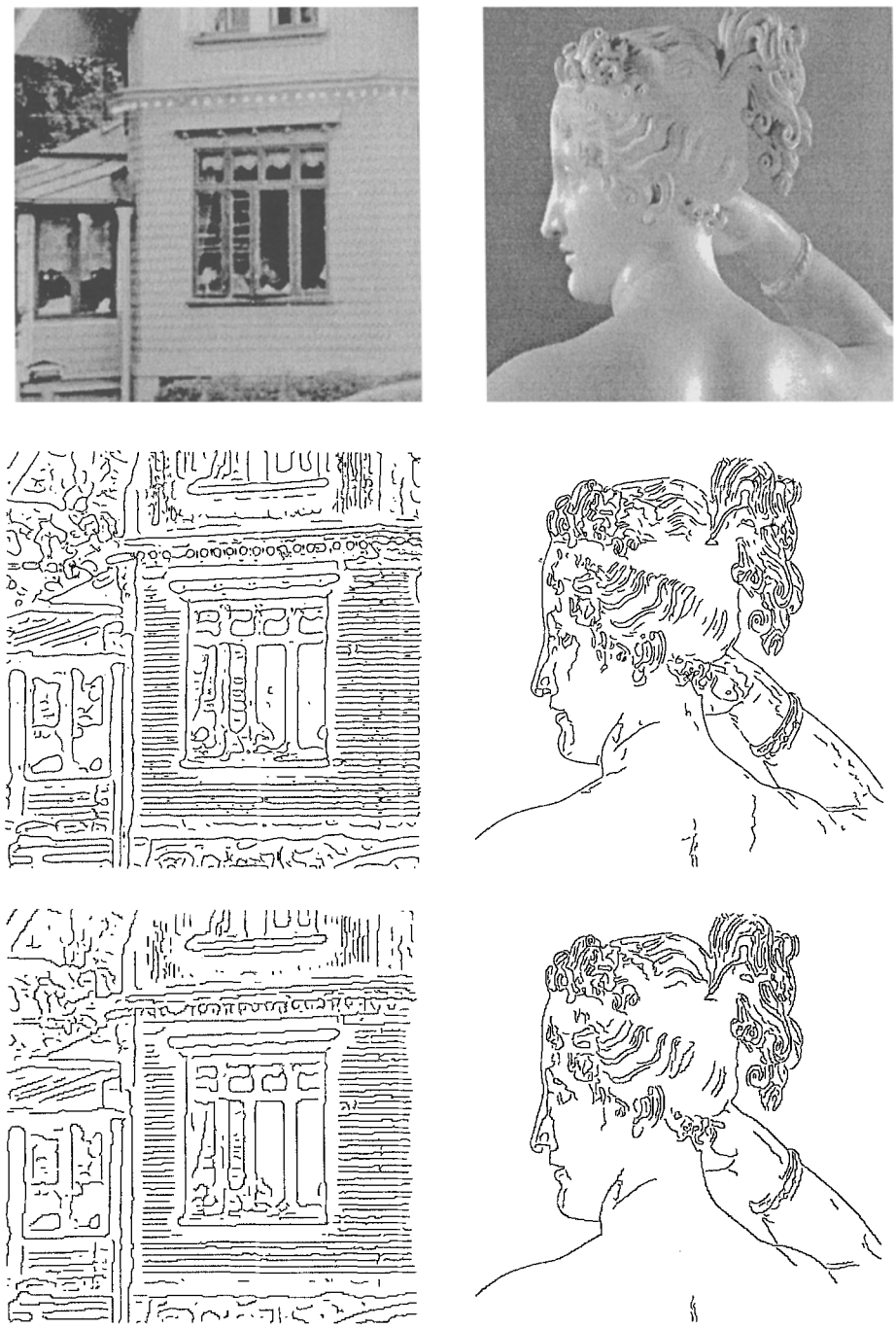

Fig. 7. A comparison of the subpixel edge-detection scheme based on discrete derivative approximations (middle row) and Canny–Deriche edge detection (bottom row). The scale values used for smoothing in the left column were $t = 4.0$ (middle left) and $\alpha = 0.7$ (bottom left), and the corresponding values in the right column were $t = 1.4$ (middle right) and $\alpha = 1.2$ (bottom right). In the left column no thresholding was performed on the gradient magnitude or the length of the edge segments, whereas in the right column hysteresis thresholds used on the gradient magnitude (low = 3.0, high = 5.0) and a threshold on the length of the edge segments (5.0) were selected manually. The left image size is 256 × 256 pixels, and the right image size is 512 × 512 pixels.

to the curvature data. Basically, each grey-level blob corresponds to one local extremum and vice versa. Observe that at fine scales mainly blobs due to noise are detected, whereas at coarser scales the operator gives a stronger response in regions that correspond to meaningful junctions in the scene.

6.3 Feature Detection from Singularities in Scale-Space

The preceding two applications exemplify how low-level feature detectors can be formulated in terms of singularities of differential entities. By this, one more step has been added to the flow of computations illustrated in figure 1, namely, *singularity detection*, which in these cases is equivalent to the detection of zero crossings and/or local extrema, operations corresponding to *nearest-neighbour processing*; see figure 8.

A main reason why this formulation in terms of singularities is important is that these singularities do not depend on the actual numerical values of the differential geometric entities but do depend on their *relative* relations. Therefore they will be less sensitive to the effect of scale-space smoothing, which is well known to *decrease the amplitude* of the variations in a signal and its derivatives. In fact, the preceding differential entities will be *invariant* to a number of primitive transformations of either the original or the smoothed grey-level signal: *translations, rotations*, and *uniform rescalings* in space as well as *affine intensity* transformations.[16]

To give a precise formulation of this, let $L_{\tilde{u}^m\tilde{v}^n} = L_{\tilde{u}^\alpha}$ denote a mixed directional derivative of order $|\alpha| = m + n$, where $\alpha = (m, n)$, and let $\mathcal{D}$ be a (possibly nonlinear) homogeneous differential expression of the form

$$\mathcal{D}L = \sum_{i=1}^{I} c_i \prod_{j=1}^{J} L_{\tilde{u}^{\alpha_{ij}}}, \tag{65}$$

where $|\alpha_{ij}| > 0$ for all $i = [1 \cdots I]$ and $j = [1 \cdots J]$, and $\sum_{j=1}^{J} |\alpha_{ij}| = N$ for all $i \in [1 \cdots I]$. Moreover, let $\mathcal{S}_\mathcal{D}L$ denote the *singularity set* of this operator, i.e., $\mathcal{S}_\mathcal{D}L = \{(x; t) : \mathcal{D}L(x; t) = 0\}$, and let $\mathcal{G}$ be the Gaussian smoothing operator, i.e., $L = \mathcal{G}f$. Under these transformations of the spatial domain (represented by $x \in \mathbb{R}^2$) and the intensity domain (represented by either the unsmoothed f or the smoothed L) the singularity sets transform as shown in table 1. In other words, feature detectors formulated in terms of differential singularities by definition commute with a number of elementary transformations of the spatial and intensity domains, and it does not matter whether the transformation is performed before or after the smoothing step.

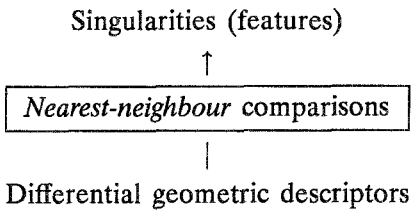

Fig. 8. The low-level feature extractors have been expressed in terms of singularities of different differential geometric entities. This corresponds to the addition of one more processing step to the flow of computations illustrated in figure 1, namely, singularity detection. This operation can (at least in these cases) be equivalently expressed in terms of detection of zero crossings (and/or local extrema) corresponding to comparisons of nearest-neighbour pixels.

In the preceding, R is a rotation matrix, Δx is a vector ($\in \mathbb{R}^2$), and a, b, and s are scalar constants. The definitions of the transformed singularity sets are as follows:

$$\mathcal{TS}_\mathcal{D}L = \{(x; t) : \mathcal{D}L(x + \Delta x; t) = 0\},$$
$$\mathcal{RS}_\mathcal{D}L = \{(x; t) : \mathcal{D}L(Rx; t) =\},$$
$$\mathcal{US}_\mathcal{D}L = \{(x; t) : \mathcal{D}L(sx; s^2t) = 0\}.$$

Moreover, such singularities can be easily *related* (and linked) *across scales* in a well-defined manner; the implicit function theorem can be used for defining paths across scales and for deriving closed-form expressions for the drift velocity of feature points with respect to scale-space smoothing; see [23] and [25] for details. For example, for a curved edge given by nonmaximum suppression, i.e., $L_{\tilde{v}\tilde{v}} = 0$, the drift velocity in the normal direction of the curve, $(\alpha_{\tilde{u}}, \alpha_{\tilde{v}}) = (L_{\tilde{v}}^2 L_{\tilde{u}\tilde{v}\tilde{v}} + 2L_{\tilde{v}} L_{\tilde{u}\tilde{v}} L_{\tilde{u}\tilde{u}},\ L_{\tilde{v}}^2 L_{\tilde{v}\tilde{v}\tilde{v}} + 2L_{\tilde{v}} L_{\tilde{u}\tilde{v}}^2)$, is

$$(\partial_t u, \partial_t v) = -\frac{L_{\tilde{v}}(L_{\tilde{u}\tilde{u}\tilde{v}\tilde{v}} + L_{\tilde{v}\tilde{v}\tilde{v}\tilde{v}} + 2L_{\tilde{u}\tilde{v}}(L_{\tilde{u}\tilde{u}\tilde{u}} + L_{\tilde{u}\tilde{v}\tilde{v}})}{2((L_{\tilde{v}} L_{\tilde{u}\tilde{v}\tilde{v}} + 2L_{\tilde{u}\tilde{v}} L_{\tilde{u}\tilde{u}})^2 + (L_{\tilde{v}} L_{\tilde{v}\tilde{v}\tilde{v}} + 2L_{\tilde{u}\tilde{v}}^2)^2)}$$

Table 1

Transformation	Definition	Invariance
Translation	$(TL)(x; t) = L(x + \Delta x; t)$	$S_D \mathcal{G} T f = S_D T \mathcal{G} f = T S_D \mathcal{G} f$
Rotation	$(\mathcal{R}L)(x; t) = L(Rx; t)$	$S_D \mathcal{G} \mathcal{R} f = S_D \mathcal{R} \mathcal{G} f = \mathcal{R} S_D \mathcal{G} f$
Uniform scaling	$(UL)(x; t) = L(sx; t)$	$S_D \mathcal{G} U f = S_D U \mathcal{G} f = U S_D \mathcal{G} f$
Affine intensity	$(\mathcal{A}L)(x; t) = aL(x; t) + b$	$S_D \mathcal{G} \mathcal{A} f = S_D \mathcal{A} \mathcal{G} f = S_D \mathcal{G} f$

$$\times \left(\frac{\alpha_{\tilde{u}}}{L_{\tilde{v}}}, \frac{\alpha_{\tilde{v}}}{L_{\tilde{v}}} \right), \tag{66}$$

which reflects the nontrivial effect of smoothing in the general case. For a straight edge, where all partial derivatives in the u-direction are zero, this expression reduces to

$$(\partial_t u, \partial_t v) = -\frac{1}{2} \frac{L_{\tilde{v}\tilde{v}\tilde{v}}}{L_{\tilde{v}\tilde{v}}} (0, 1). \tag{67}$$

6.4 Selective Mechanisms

In the treatment so far, the major aim has been to demonstrate what information can be obtained from the computed data without introducing any commitments in the processing. Therefore no attempts have been made to suppress irrelevant edges or junction candidates by thresholding or by other means. Nevertheless, when the output from these processing modules is used as input to other ones, there is an obvious need for some *selective mechanisms* for deciding what structures in the image should be regarded as significant and what scales are appropriate for treating those.

The top row in figure 9 shows the effect of introducing a low (manually selected) threshold on gradient magnitude in the edge-detection step. Of course, if desired, hysteresis thresholding can be easily introduced into the scheme in a similar manner. The bottom row in this figure illustrates a more refined way of automatically selecting a sparse subset of junction candidates for further processing; it shows the result of applying a *multiscale* blob detection method, the scale-space primal sketch [21], [22], to the absolute value of the curvature data computed from (64). Obviously, the output from this *second-stage multiscale analysis*, performed after the (nonlinear) computation of the differential descriptor generates output results that are much more useful for further processing than those generated by the single-scale extraction of grey-level blobs illustrated in figure 6; see also [4] and [27].

7 Summary and Discussion

The main subject of this paper has been a description of a canonical way to discretize the primary components in scale-space theory—the convolution smoothing, the diffusion equation, and the smoothed derivatives—such that the scale-space properties hold exactly in the discrete domain.

A theoretical advantage of the proposed discrete theory is that several algebraic scale-space properties in the continuous case transfer directly to the discrete domain and operators that commute in the continuous case commute (exactly) also after the discretization. Examples of this are the nonenhancement property of local extrema and the semigroup/cascade smoothing property of the smoothing operator.

A computational advantage with the proposed discrete analogues of the Gaussian-derivative kernels is that there is no need for redoing the smoothing part of the transformation when several derivative approximations are computed. Exactly the same results are obtained by smoothing the signal with the (large support) discrete analogue of the Gaussian kernel (once) and then computing the derivative approximations by applying different (small support) difference operators to the output from the smoothing operation.

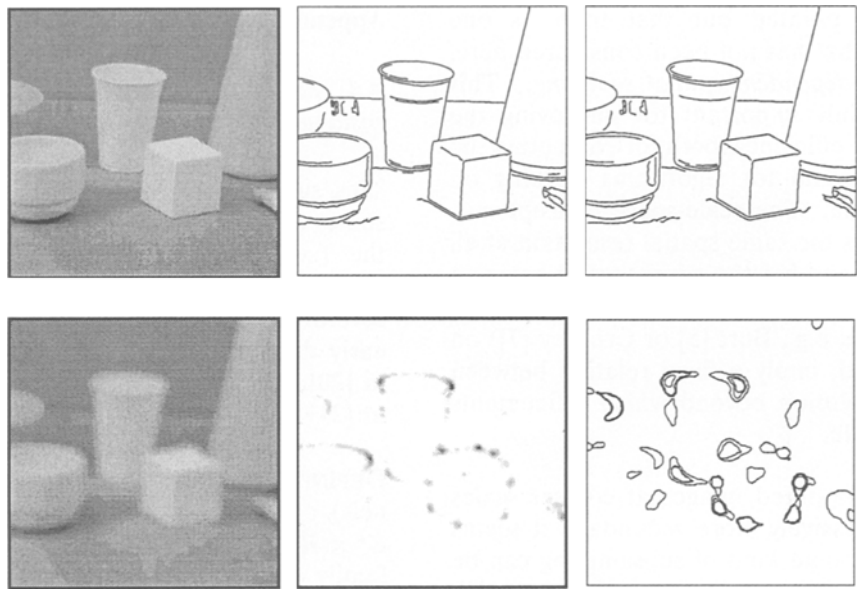

Fig. 9. Edges detected from the table scene image by using a low (manually selected) hysteresis threshold on gradient magnitude: (top left) grey-level image, (top middle) edges detected at scale $t = 2$ (low = 1.0, high = 3.0), and (top right) edges detected at scale $t = 4$ (low =1.0, high =2.0). Suppression of junction candidates without thresholding on the operator response: (bottom left) image smoothed to scale $t = 8$ (bottom middle) absolute value of the rescaled level curve curvature computed at that scale, and (bottom right) boundaries of the 40 most significant curvature blobs extracted by applying a multiscale blob detection method (the scale-space primal sketch [21], [22]) to the curvature data.

The specific difference operators $\delta_{x^i y^j}$ used in this presentation have been selected so that they in a numerical sense constitute consistent discrete approximations to the continuous derivative operators. This means that the discrete approximations will approach the continuous results when the grid effects get smaller, i.e., when the grid spacing becomes small compared to a characteristic length in the data. Hence, with increasing scale the output from the proposed discrete operators can be expected to approach the corresponding continuous theory results. Any specific convergence result, however, depend on what assumptions are posed on the continuous signal and the sampling method.

The proposed framework has been derived from a set of theoretical properties postulated on the first stages of visual processing. In practice, it leads to a conceptually very simple scheme for computing low-level features from raw (discrete) image data; the scheme lends itself to direct implementation in terms of natural operations in visual front end: (i) *large support linear* smoothing, (ii) *small support linear* derivative approximations, (iii) *pointwise nonlinear* combinations of derivatives into differential geometric descriptors, and (iv) *nearest-neighbour comparisons*. Experiments demonstrate that this scheme gives useful results in edge detection and junction detection using derivatives of order up to three. At the end of section 6.4 it was also indicated how useful results in junction detection and shape from texture can be obtained, by adding a *second-stage scale-space smoothing* step to the output from step (iii) above.

Finally, it should be emphasized that although the treatment here, because of simplicity of presentation, has been concerned with one-dimensional and two-dimensional signals, the methodology is general and can be applied in arbitrary dimensions; see appendix A.5 for a condensed review of the main result that holds in higher dimensions.

7.1 Further Work

Concerning possible generalizations of this work,

it should be pointed out that there is one main subject that has not been considered here, namely, *scale-dependent spatial sampling*. This issue is certainly important for improving the computational efficiency, both for computing the representation and for algorithms working on the output data. The scale-space concept outlined here uses the same spatial resolution at all levels of scale and has the advantage of having a *continuous scale parameter*. The pyramid representations (see, e.g., Burt [5] or Crowley [7]) on the other hand, imply a fixed relation between scale and resolution beyond which refinements are not possible.

Since the smoothed images at coarser scales become progressively more redundant, it seems plausible that some kind of subsampling can be done at the coarser scales without too much loss of information. It would be interesting to analyse how much information is lost by such an operation and to what extent one can introduce a subsampling operator into this representation while still maintaining the theoretical properties associated with having a continuous scale parameter and without introducing any severe discontinuities along the scale direction, which would be a potential source of numerical difficulties for algorithms working on the output from the representation. For example, it is a much harder algorithmic problem to relate structures across scales in a pyramid than in a scale-space having a continuous scale parameter; see, e.g., [21], [22].

Another important problem is how different types of *a priori* knowledge can be incorporated into the analysis. The work presented here concerns *linear* and spatially-shift-invariant isotropic smoothing, based on the argument that in situations in which no information is available, the visual front-end processing should be as *uncommitted* as possible. On the other hand, once any (initial) knowledge is available, a variety of different possibilities open up. A natural generalization to consider is *nonlinear diffusion*, although further work may be needed to develop the notion of anisotropic smoothing as introduced into computer vision by Perona and Malik [31] and Nordström [30].

Appendix A

A.1 Summary of Main Results from Two-Dimensional Discrete Scale-Space Theory

This subsection states, for reference purposes, some basic definitions and key results from the (zero-order) scale-space theory for two-dimensional discrete signals, which are used in several of the proofs presented in this paper. An early description of this material can be found in [20], and a more rigorous treatment is given in [21] and [24].

DEFINITION 12 (Pre-scale-space family of kernels). A one-parameter family of kernels $T : Z^2 \times R_+ \to R$ is said to be a pre-scale-space family of kernels if it satisfies

(i) $T(\cdot, \cdot; 0) = \delta(\cdot, \cdot)$,
(ii) the semigroup property $T(\cdot, \cdot; s) * T(\cdot, \cdot; t) = T(\cdot, \cdot; s + t)$,
(iii) the symmetry properties[17] $T(-x, y; t) = T(x, y; t)$ and $T(y, x; t) = T(x, y; t)$ for all $(x, y) \in Z^2$, and
(iv) the continuity requirement $\|T(\cdot, \cdot; t) - \delta(\cdot, \cdot)\|_1 \to 0$ when $t \downarrow 0$.

DEFINITION 13 (Pre-scale-space representation). Let $f : \mathbb{Z}^2 \to \mathbb{R}$ be a discrete signal, and let $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be a pre-scale-space family of kernels. Then the one-parameter family of signals $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ given by

$$L(x, y; t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} T(m, n; t) \times f(x - m, y - n) \qquad (68)$$

is said to be the pre-scale-space representation of f generated by T.

LEMMA 14 (a pre-scale-space representation is differentiable). Let $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be the pre-scale-space representation of a signal $f : \mathbb{Z}^2 \to \mathbb{R}$ in l_1. Then, L satisfies the differential equation,

$$\partial_t L = \mathcal{A} L \qquad (69)$$

for some linear and shift-invariant operator $\mathcal{A}$.

Proof. See Lemma 3.1 in [21] (or part I of the proof of Theorem 7 in [20]).

DEFINITION 15 (scale-space family of kernels). A one-parameter family of pre-scale-space kernels $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ is said to be a scale-space family of kernels if for any signal $f : \mathbb{Z}^2 \to \mathbb{R} \in l_1$ the pre-scale-space representation of f generated by T obeys the nonenhancement property stated in Definition 6, i.e., if for any signal f local extrema in L are never enhanced.

DEFINITION 16 (scale-space representation). A pre-scale-space representation $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ of a signal $f : \mathbb{Z}^2 \to \mathbb{R}$ generated by a family of kernels $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$, which are scale-space kernels, is said to be a scale-space representation of f.

THEOREM 17 (scale-space for two-dimensional discrete signals: necessity). A scale-space representation $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ of a signal $f : \mathbb{Z}^2 \to \mathbb{R}$ satisfies the differential equation

$$\partial_t L = \alpha \nabla_5^2 L + \beta \nabla_\times^2 L \tag{70}$$

with initial condition $L(\cdot, \cdot; 0) = f(\cdot, \cdot)$ for some constants $\alpha \geq 0$ and $\beta \geq 0$.

Proof. See Theorem 3.2 in [21] or (part II of the proof of Theorem 7 in [20]).

THEOREM 18 (scale-space for two-dimensional discrete signals: sufficiency). Let $f : \mathbb{Z}^2 \to \mathbb{R}$ be a discrete signal in l_1, and let $L : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ be the representation generated by the solution to differential equation

$$\partial_t L = \alpha \nabla_5^2 L + \beta \nabla_\times^2 L, \tag{71}$$

with initial condition $L(\cdot, \cdot; 0) = f(\cdot, \cdot)$ for some fixed $\alpha \geq 0$ and $\beta \geq 0$. Then L is a scale-space representation of f.

Proof. See Theorem 3.3 in [21] (or Theorem 8 in [20]).

A.2 Proof of Theorem 9

The proof will be similar to the proof of Theorem 17, which was based on a number of different test signals that were used for successively restricting the class of possible smoothing transformations. Since, however, it cannot be guaranteed that those signals will be in the range of $D_{x^i y^j}$, the proof has to be slightly modified.

To start with, observe that as a result of Lemma 5 it holds that $L_{x^i y^j}$ obeys a linear differential equation. Because of the shift invariance, $\mathcal{A}L_{x^i y^j}$ can be written

$$(\mathcal{A}L_{x^i y^j})(x, y; t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} a_{m,n} L_{x^i y^j}(x - m, y - n; t) \tag{72}$$

for some set of filter coefficients $a_{m,n}$. Also, observe that since the null space of $D_{x^i y^j}$ cannot be expected to consist of the zero element only, $\mathcal{A}$ cannot be completely determined but can be determined only modulus the null space of $D_{x^i y^j}$. Therefore (for this proof) introduce a notion of equivalence, so that two operators $\mathcal{A}$ and $\mathcal{A}'$ are treated as equivalent if $\mathcal{A} = \mathcal{A}' + \mathcal{A}_\mathcal{N}$ for some element $\mathcal{A}_\mathcal{N}$ in the null space of $D_{x^i y^j}$. In the first step of the proof it will be shown that $\mathcal{A}$ must be equivalent to an operator having *finite support.*

Let f_0 be a finite support signal such that $D_{x^i y^j} * f_0$ is not identically zero. Then $D_{x^i y^j} * f_0$ has finite support. For some real constant b, consider a signal f_1 given by $f_1(x, y) = f_0(x, y) + b f_0(x - x_0, y - y_0)$, where (x_0, y_0) is selected sufficiently far away from the origin so that the support regions of $(D_{x^i y^j} * f_0)(x, y)$ and $(D_{x^i y^j} * f_0)(x - x_0, y - y_0)$ do not overlap and are separated by a set of (four-connected) zeros. Then there must exist at least one extremum point (x_1, y_1) in the support region of $(D_{x^i y^j} * f_0)(x, y)$. At that point it holds that

$$\begin{aligned}(\partial_t L_{x^i y^j})(x_1, y_1; 0) &= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} a_{m,n} (D_{x^i y^j} * f_0)(x_1 - m, y_1 - n) \\ &\quad + b \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} a_{m,n} \\ &\quad (D_{x^i y^j} * f_0)(x_1 - x_0 - m, y_1 - y_0 - n) \\ &= A + bB. \end{aligned} \tag{73}$$

If the latter sum B is nonzero, then since the first sum A is bounded, it is always possible to select a value of b so that this expression assumes an arbitrary sign. This means that if (x_1, y_1) is a local maximum, then a value of b can be selected so that $(\partial_t L_{x^i y^j})(x_1, y_1)$ is strictly positive, and if (x_1, y_1) is a local minimum, then there exists a value of b so that $(\partial_t L_{x^i y^j})(x_1, y_1)$ is strictly negative. In other words, if B is nonzero, then a value of b can always be found that leads to a violation of the nonenhancement property of local extrema. Now since B must be zero for any selection of f_0 and any (x_1, y_1) such that the support regions of $(D_{x^i y^j} * f_0)(x, y)$ and $(D_{x^i y^j} * f_0)(x - x_1, y - y_1)$ are separated, it follows that $\mathcal{A}$ must be equivalent to an operator $\mathcal{A}'$ corresponding to convolution with a finite support kernel. In other words, without loss of generality it can be assumed that there exists some M such that $a'_{m,n} = 0$ if $|m| > M$ or $|n| > M$. Let R_A be the region $R_A = \{(x, y) \in \mathbb{Z}^2 : (|x| \leq M) \wedge (|y| \leq M)\}$.

Now the extremum-point conditions (23) and (24) will be combined with Definitions 7 and 8 to show that $\mathcal{A}'$ must be equivalent to a *local operator*, i.e., an operator for which $a_{m,n} = 0$ if $|m| > 1$ or $|n| > 1$. This is easily understood by studying the following counterexample. Given some constant c and any point $(x_2, y_2) \in \mathbb{Z}^2$ with $\max(|x_2|, |y_2|) > 1$, let $f_2 : \mathbb{Z}^2 \to \mathbb{R}$ be an input signal so that $L_{x^i y^j}(x, y; 0)$ satisfies

$$L_{x^i y^j}(x, y; 0) = \begin{cases} 1 & \text{if } (x, y) = (0, 0), \\ c & \text{if } (x, y) = (-x_2, -y_2), \\ 0 & \text{for any other } (x, y) \in R_A. \end{cases} \tag{74}$$

Such a signal can always be constructed if $(D_{x^i y^j})(x, y; 0)$ has at least one nonzero coefficient (see appendix A.4). Obviously $(0, 0)$ is a local maximum, which gives

$$(\partial_t L_{x^i y^j})(0, 0; 0) = a_{0,0} + b a_{x_2, y_2}. \tag{75}$$

If a_{x_2, y_2} is nonzero, then we can always select a value of b so that $(\partial_t L_{x^i y^j})(0, 0; 0) > 0$ and the nonenhancement property of local extrema is violated. Hence for any (x_2, y_2) with $\max(|x_2|, |y_2|) > 1$, a_{x_2, y_2} must be zero, i.e, $\mathcal{A}$ must be equivalent to a local operator. Denote that operator by $\mathcal{A}''$.

Since $L_{x^i y^j}$ is assumed to be a scale-space family of derivative approximations, there exists some family of kernels $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ such that $L_{x^i y^j}(\cdot, \cdot; t) = T(\cdot, \cdot; t) * L_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot)$. Because of the *symmetry* requirements on T, $T(-x, y; t) = T(x, y; t)$ and $T(y, x; t) = T(x, y; t)$, it follows that it can be assumed that $\mathcal{A}''$ possesses the same symmetry properties, i.e., that $\mathcal{A}'' L_{x^i y^j}$ can be written

$$\partial_t L_{x^i y^j} = \begin{pmatrix} a & b & a \\ b & c & b \\ a & b & a \end{pmatrix} L_{x^i y^j} \tag{76}$$

for some (other) constants a, b, and c. Finally, let f_3 be a signal such that

$$L_{x^i y^j}(x, y; t) = \begin{cases} 1 & \text{for all } (x, y) \text{ with } |x| \leq 1 \text{ and } |y| \leq 1, \\ 0 & \text{for any other } (x, y) \in R_A. \end{cases} \tag{77}$$

(Such a signal can always be constructed in the way described in appendix A.4.) Since $(0, 0)$ is both a maximum point and a minimum point, it follows by analogy with the proof of Theorem 9, that the *sum of the coefficients must be zero* in $(\partial_t L_{x^i y^j})(0, 0; 0) = 4a + 4b + c$ because of the nonenhancement property of local extrema. Trivially, c must be nonpositive, and $\mathcal{A}''$ can be written

$$\mathcal{A}'' L_{x^i y^j} = \alpha \nabla_5^2 L_{x^i y^j} + \beta \nabla_\times^2 L_{x^i y^j} \tag{78}$$

for some nonnegative constants α and β.

A.3 Proof of Theorem 10

From Theorem 18 we have that $L_{x^i y^j}$ is a scale-space representation of $D_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot)$. Hence $L_{x^i y^j}$ satisfies the nonenhancement property of local extrema for any f.

Since (71) is a linear differential equation with constant coefficients, its solution can be written $L(\cdot, \cdot; t) = T(\cdot, \cdot; t) * f(\cdot, \cdot)$ for some family of kernels $T : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$. By considering the solution of (71) at time t_2 computed from the original signal f and by considering the solution at time $t_2 \to t_1$ by using as input signal the solution at time t_1 computed

from the original signal, it follows that T must obey the semigroup property. Moreover, since the operators ∇_5^2 and $\nabla_\times^2$ are symmetric with respect to coordinate sign changes and reversals of x and y, the family T must satisfy the symmetry constraints $T(-x, y; t) = T(x, y; t)$ and $T(y, x; t) = T(x, y; t)$ for all $(x, y) \in \mathbb{Z}^2$. Furthermore, the differentiability implies continuity at the origin. By replacing f with $D_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot)$ it follows that the solution to (33) can be written

$$L_{x^i y^j}(\cdot, \cdot; 0) = T(\cdot, \cdot; t) * D_{x^i y^j}(\cdot, \cdot; 0) * f(\cdot, \cdot). \quad (79)$$

Then $D_{x^i y^j} : \mathbb{Z}^2 \times \mathbb{R}_+ \to \mathbb{R}$ defined by $D_{x^i y^j}(\cdot, \cdot; t) = T(\cdot, \cdot; t) * D_{x^i y^j}(\cdot, \cdot; 0)$ is a prescale-space family of derivative kernels. Since $L_{x^i y^j}$ satisfies the nonenhancement property for any input signal, it follows that $D_{x^i y^j}$ is a scale-space family of derivative kernels and thus that $L_{x^i y^j}$ is a scale-space family of derivative approximations.

A.4 Generating Input Signals

In this subsection the validity of a trivial statement will be verified, namely, that given an arbitrary kernel K of finite support having at least one nonzero filter coefficient, an arbitrary signal f_{out}, and an arbitrary square region R, it is always possible to construct an input signal f_{in} such that $f_{\text{out}} = K * f_{\text{in}}$ on R. (Of course, since K cannot be expected to be invertible, it can never be guaranteed that f_{out} will be in the range of K. What the result states is that it is always possible to find a *finite* interval so that on *that interval* $K * f_{\text{in}}$ is equal to some function in the range of K.)

In the one-dimensional case this is trivial since f_{in} will be given by a triangular linear system of equations. In the two-dimensional case just a little more care needs be taken: in the filter mask corresponding to K, denote by a the leftmost nonzero filter coefficient in the uppermost row containing nonzero filter coefficients, i.e., let $a = K_{ij}$, where i is the minimum value of μ such that $K_{\mu\nu} \neq 0$ for some ν and j, in turn, is the maximum value of μ such that $K_{i\mu} \neq 0$. Then the values of f_{in} can be determined one by one simply by traversing each row in R, starting from the left in the uppermost row. If the desired value of f_{out} in this point is b, then let f_{in} be equal to b/a in the point corresponding to the center of the filter mask. If the desired value in the next point is c and $K_{i,j-1} = d$, let f_{in} in the next point be equal to $c - bd/a$, etc. Obviously, this procedure also leads to a triangular system of equations.

A.5 Discrete Derivative Approximations to N-Dimensional Signals

Concerning discrete signals $f : \mathbb{Z}^N \to \mathbb{R}$ of arbitrary dimension, it holds that the representation $L : \mathbb{Z}^N \times \mathbb{R}_+ \to \mathbb{R}$ given by separable convolution with the one-dimensional discrete analogue of the Gaussian kernel $T : \mathbb{Z} \times \mathbb{R}_+ \to \mathbb{R}$ along each coordinate direction possesses all the scale-space properties listed in section 1. More generally, it is shown in [24] that a one-parameter family $L : \mathbb{Z}^N \times \mathbb{R}_+ \to \mathbb{R}$ of N-dimensional discrete signals is a scale-space representation[18] if and only if it satisfies the differential equation

$$(\mathcal{A}L)(x; t) = \sum_{\xi \in \mathbb{Z}^N} a_\xi L(x - \xi; t) \quad (80)$$

for some set of filter coefficients $a_\xi \in R$ satisfying

(i) the *locality* condition $a_\xi = 0$ if $\|\xi\| > 1$,
(ii) the *positivity* constraint $a_\xi \geq 0$ if $\xi \neq 0$,
(iii) the *zero sum* condition $\sum_{\xi \in \mathbb{Z}^N} a_\xi = 0$, and
(iv) the *symmetry* requirements $a_{(-\xi_1, \xi_2, \ldots, \xi_N)} = a_{(\xi_1, \xi_2, \ldots, \xi_N)}$ and $a_{P_k^N(\xi_1, \xi_2, \ldots, \xi_N)} = a_{\xi_1, \xi_2, \ldots, \xi_N}$ for all $\xi = (\xi_1, \xi_2, \ldots, \xi_N) \in \mathbb{Z}^N$ and all possible permutations P_k^N of N elements.

By analogy with Proposition 2, the scale-space properties transfer to any discrete derivative approximation defined by linear filtering. This framework provides a complete catalogue of the discrete derivative approximations with scale-space properties in arbitrary dimensions.

Acknowledgments

I would like to thank Jan-Olof Eklundh for con-

tinuous support and encouragement, as well as Fredrik Bergholm, Jonas Gårding, and Harald Winroth for useful comments. The "Paolina" image was kindly provided by Pietro Perona. This work was partially performed under the ESPRIT-BRA project INSIGHT. The support from the Swedish National Board for Industrial and Technical Development, NUTEK, is gratefully acknowledged.

Notes

1. The scale parameter t used in this presentation corresponds to σ^2, where σ is the standard deviation of the Gaussian kernel.
2. IR_+ denotes the set of real nonnegative numbers.
3. It is well known that under rather weak requirements on the input signal f the solution to the diffusion equation will be infinitely differentiable (C^∞) for $t > 0$. However, stronger regularity requirements must be posed on f in order for the last equality to be valid.
4. A rule of thumb sometimes used in this context is that when derivatives of order two and higher are computed from raw image data, then the amplitude of the amplified noise will often be of (at least) the same order of magnitude as the derivative of the signal.
5. The conditions concerning the finite-support convolution kernel and $f \in l_1$ can in fact be weakened. However, the generality of this statement is sufficient for our purpose.
6. A discrete derivative approximation, $\delta^{(h)}_{x^i y^j}$ depending on a step length h is said to be *consistent* if (under reasonable assumptions on the signal $L : \mathrm{IR}^2 \to \mathrm{IR}$) the truncation error tends to zero as the step length tends to zero, i.e., if $\lim_{h \downarrow 0}(\delta^{(h)}_{x^i y^j} L)(x_0, y_0) = (\partial_{x^i y^j} L)(x_0, y_0)$. In our considerations h is omitted from the notation, since the grid spacing is throughout assumed to be equal to one.
7. Here, a *weak* notion of local extremum is used; a two-dimensional point $(x_0, y_0) \in \mathbb{Z}^2$ is said to be a local maximum if its grey-level value is greater than or equal to the grey-level values in all its (eight-)neighbours.
8. Note that nowhere in the proofs has use been made of the fact that $D_{x^i y^j}$ is a derivative approximation operator. Corresponding results hold if $D_{x^i y^j}$ is replaced by an *arbitrary linear operator*.
9. The second-difference operator can, of course, also be defined as $\mathcal{D}_{xx} = \delta_x \delta_x$. Then, however, $\mathcal{D}_{xx} \neq \nabla^2_3$. Another possibility is to use both the forward-difference operator, $(\delta_{x+} L)(x; t) = L(x + 1; t) - L(x; t)$, and the backward-difference operator, $(\delta_{x-} L)(x; t) = L(x; t) - L(x - 1; t)$ and, e.g., to define $\tilde{\mathcal{D}}_x = \delta_{x+}$ and $\tilde{\mathcal{D}}_{xx} = \delta_{x-} \delta_{x+}$. By this, $\tilde{\mathcal{D}}_{xx}$ will correspond to (a translate of) $\tilde{\mathcal{D}}^2_x$. Then, however, the odd-order derivatives are no longer estimated at the grid points and (38) no longer holds. Nevertheless, the commutative algebraic structure with respect to smoothing and derivative approximations is preserved independently of this choice.
10. In numerical analysis this operator is known as the rotationally-most-symmetric 3×3 discrete approximation to the Laplacian operator [8]; see also the result concerning the Fourier transform in section 2.2.
11. This condition is a necessary requirement for $\partial_t L = \frac{1}{2}(\tilde{\delta}_{xx} + \tilde{\delta}_{yy})L$ to hold.
12. Similar operators have also been used in *pyramid* representations; see, e.g., Burt and Adelson [5] and Crowley and Stern [7].
13. The same problem arises also if the computations are performed in the Fourier domain, since at least one inverse fast Fouriers transformation will be needed for each derivative approximation.
14. Some care must, however, be taken with respect to the sign pattern of $\tilde{L}_{\tilde{v}\tilde{v}\tilde{v}}$ and ambiguous situations, for example, when all adjacent points in a four-cell have opposite signs in $\tilde{L}_{\tilde{v}\tilde{v}}$).
15. These were derived solely from theoretical scale-space considerations.
16. See also [10] concerning necessity of results regarding a similar (but not identical) set of transformations.
17. $T(x, -y; t) = T(x, y; t)$ is implied from the two other properties.
18. In the sense that it obeys that N-dimensional analogies of Definition 6, the semigroup property, and the symmetry and continuity conditions in Definition 3.

References

1. M. Abramowitz and I.A. Stegun, *Handbook of Mathematical Functions*, Applied Mathematics Series, vol. 55, National Bureau of Standards, Gaithersburg, MD, 1964.
2. J. Babaud, A.P. Witkin, M. Baudin and R.O. Duda, "Uniqueness of the Gaussian kernel for scale-space filtering," *IEEE Trans. Patt. Anal. Mach. Intell.* vol. 8, pp. 26–33, 1986.
3. J. Blom, "Topological and geometrical aspects of image structure," *Ph.D. thesis*, Department of Medical Physics, University of Utrecht, Utrecht, The Netherlands, 1992.
4. K., Brunnström, T.P. Lindeberg and J.-O. Eklundh, "Active detection and classification of junctions by foveation with a head–eye system guided by the scale-space primal sketch," in *Proc. 2nd European Conference on Computer Vision*, Santa Margherita Ligure, Italy, May 19–22, 1992, pp. 701–709.
5. P.J. Burt and E.H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.* vol. 9, pp. 532–540, 1983.
6. J. Canny, "A computational approach to edge detection," *IEEE Trans. Patt. Anal. Mach. Intell.* vol. 8, pp. 679–698, 1986.
7. J.L. Crowley and R.M. Stern "Fast computation of the difference of low pass transform," *IEEE Trans. Patt. Anal. Mach. Intell.* vol. 6, pp. 212–222, 1984.

8. G. Dahlquist, Å. Björk, and N. Anderson, *Numerical Methods*, Prentice-Hall: London, 1974.
9. R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector," Int. J. Comput. Vis., vol. 5, pp. 167–187, 1987.
10. L.M.J. Florack, B.M. ter Haar Romeny, J.J. Koenderink and M.A. Viergever, "General intensity transformations and second order invariants," in *Proc. 7th Scandinavian Conference on Image Analysis*, Aalborg, Denmark, Aug. 13–16, 1991, pp. 338–345.
11. L.M.J. Florack, B.M. ter Haar Romeny, J.J. Koenderink, M.A. Viergever, "Scale and the differential structure of images," *Image Vis. Comput.* vol. 10, pp. 376–388, 1991.
12. W. Freeman, and E. Adelson, "The design and use of steerable filters for image analysis and wavelet decomposition," in *Proc. 3rd International Conference on Computer Vision*, Osaka, Japan, 1990, pp. 406–415.
13. R.M. Haralick, "Digital step edges from zero-crossings of second directional derivatives," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 6, 1984.
14. L. Kitchen and R. Rosenfeld, "Gray-level corner detection," *Patt. Recogn. Lett.*, vol. 1, pp. 95–102, 1982.
15. J.J. Koenderink, "The structure of images," *Biol. Cybernet.* vol. 50, pp. 363–370, 1984.
16. J.J. Koenderink and A.J. van Doorn, "Representation of local geometry in the visual system," *Biol. Cybernet.* vol. 55, pp. 367–375, 1987.
17. J.J. Koenderink and W. Richards, "Two-dimensional curvature operators," *J. Opt. Soc. Am.*, vol. 5, pp. 1136–1141, 1988.
18. J.J. Koenderink and A.J. van Doorn, "Receptive field families," *Biol. Cybernet.*, vol. 63, pp. 291–297, 1990.
19. A.F. Korn, "Toward a symbolic representation of intensity changes in images," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 10, pp. 610–625, 1988.
20. T. Lindeberg, "Scale-space for discrete signals," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 12, pp. 234–254, 1990.
21. T. Lindeberg, *Discrete scale-space theory and the scale-space primal sketch*, Ph.D. thesis, Royal Institute of Technology, Stockholm; revised and extended version is to appear as book *Scale-Space Theory in Early Vision* in the Kluwer International Series in Engineering and Computer Science.
22. T. Lindeberg and J.O. Eklundh, "On the computation of a scale-space primal sketch," *J. Vis. Commun. Image Repro.* vol. 2, pp. 55–78, 1991.
23. T. Lindeberg "Scale-space behaviour of local extrema and blobs," *J. Math. Imag. Vis.* vol. 1, pp. 65–99, 1992.
24. T. Lindeberg "Scale-spacefor N-dimensional discrete signals," in *Proc. Shape in Picture*, Driebergen, The Netherlands, Sept. 7–11, 1992. Ying et al. (eds.) NATO ASI Series F, Springer-Verlag (in press).
25. T. Lindeberg, "Scale-space behaviour and invariance properties of differential singularities," in *Proc. Shape in Picture*, Driebergen, The Netherlands, Sept. 7–11, 1992. Ying et al. (eds.) NATO ASI Series F, Springer-Verlag (in press).
26. T. Lindeberg, "On scale selection for differential operators," in *Proc. 8th Scandinavian Conference on Image Analysis* Tromsö, Norway, May 1992, pp. 857–866.
27. T. Lindeberg and J. Gårding, "Shape from texture from a multi-scale perspective," in *Proc. 4th International Conference on Computer Vision*, Berlin, May 11–14, 1993, pp. 683–691; An extended version is available as a technical report from the Royal Institute of Technology, Stockholm.
28. D. Marr and E. Hildreth "Theory of edge detection," *Proc. Roy. Soc. London, Ser. B*, vol. 207, pp. 187–217, 1980.
29. P. Meer and I. Weiss, "Smoothed differentiation filters for images," *J. Vis. Commun. Image Repro.*, vol. 3, pp. 58–72, 1992.
30. N. Nordström, "Biased anisotropic diffusion—a unified regularization and diffusion approach to edge detection," *Proc. 2nd Conference on Computer Vision*, Antibes, France, 1990, pp. 18–27.
31. P. Perona and J. Malik, "Scale space and edge detection using anisotropic diffusion," in *Proc. IEEE Workshop on Computer Vision*, Miami, FL, 1987, pp. 16–22.
32. P. Perona, "Steerable-scalable kernels for edge detection and junction analysis," in *Proc. 2nd European Conference on Computer Vision*, Santa Margherita Ligure, Italy, May 19–22, 1992, pp. 3–18.
33. T. Vieville and O.D. Faugeras, "Robust and fast computation of unbiased intensity derivatives in images," in *Proc. 2nd European Conference on Computer Vision*, Santa Margherita Ligure, Italy, May 19–22, 1992, pp. 203–211.
34. A.P. Witkin, "Scale-space filtering," in *Proc. 8th International Joint Conference on Artificial Intelligence, Karlsruhe*, West Germany, Aug. 8–12, 1983, pp. 1019–1022.
35. A. Yuille and T. Poggio, "Scaling theorems for zero-crossings," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 9, pp. 15–25, 1986.
36. W. Zhang and F. Bergholm, "An extension of Marr's signature based edge classification," in *Proc. 7th Scandinavian Conference on Image Analysis*, Aalborg, Denmark, Aug. 13–16, 1991, pp. 435–443.

Tony Lindeberg received his M.Sc. degree in engineering physics and applied mathematics from the Royal Institute of Technology, Stockholm, in 1987. After his studies he was first with the research group in computer fluid dynamics at the Aeronautical Research Institute of Sweden. Then he joined the Department of Numerical Analysis and Computing Science at the Royal Institute of Technology, from which he received his Ph.D. degree in computing science with the dissertation "Discrete Scale-Space Theory and the Scale-Space Primal Sketch" in 1991. He is currently a research associate at the Computational Vision and Active Perception Laboratory at the Royal Institute of Technology. His primary research interests in computer vision are scale-space methods, description of shape, and focus of attention.