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

Tony Lindeberg[*]

Computational Vision and Active Perception Laboratory (CVAP)
Department of Numerical Analysis and Computing Science
Royal Institute of Technology, S-100 44 Stockholm, Sweden
Email: tony@bion.kth.se

Abstract

It is developed 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,* an axiomatic derivation is given of how a multi-scale 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 which constitute *discrete analogues* to the continuous Gaussian derivatives.

The representation has theoretical advantages to other discretizations of the scale-space theory in the sense that operators which 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 conceptually very simple scheme of computations for multi-scale low-level feature extraction, consisting 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 are given demonstrating how the proposed scheme can be used for edge detection and junction detection based on derivatives up to order three.

Keywords: scale-space, visual front end, smoothing, Gaussian filtering, Gaussian derivative, discrete approximation, edge detection, junction detection, multi-scale 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. The property 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, like "point" or "line", only exist as meaningful entities over certain finite ranges of scale. This intrinsic property is the basic reason to why a *multi-scale representation* is of crucial importance when describing the structure of an image. A methodology proposed by Witkin [34] and Koenderink [15] to obtain such a multi-scale representation is by embedding the signal into a one-parameter family of derived signals, the *scale-space*, where the parameter t, called scale parameter[1], describes the current level of scale.

Let us briefly review some aspects of this procedure 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 $g : \mathbb{R}^2 \times \mathbb{R}_+ \setminus \{0\} \to \mathbb{R}$

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

where

$$g(x, y;\ t) = \frac{1}{2\pi t} e^{-\frac{x^2 + y^2}{2t}}. \tag{2}$$

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

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

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

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

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

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

implying that there in principle are 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

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

---

[1] The scale parameter t used in this presentation corresponds to σ^2, where σ is the standard deviation of the Gaussian kernel.

[2] $\mathbb{R}_+$ denotes the set of real non-negative 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 to the last equality to be valid.

and inherit the cascade smoothing property of Gaussian smoothing

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

associated with the semi-group property of the Gaussian kernel

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

This type of multi-scale representation based on smoothed derivatives, or *Gaussian derivatives*, has been proposed by Koenderink and van Doorn (1987) 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 multi-scale differential geometric representation of the signal. This type of framework is useful for a variety early vision tasks.

A problem that arises in this context concerns how these operations should be discretized when to be implemented in a machine vision system. The above theory is expressed for continuous signals, while realistic signals obtained from standard cameras are *discrete*. Although a standard discretization of the continuous equations could be expected to give a behaviour, which 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 concerns how sensitive these operations will be to noise, in particular when derivatives of high order are to be computed from noisy measured data[4].

In this presentation we will develop a *discrete analogue of the multi-scale Gaussian derivative representation*. We will treat the case with discrete signals defined on an infinite and uniformly sampled square grid, and propose a set of discrete operators, 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 $\delta_{x^i y^j}$, we will show how a multi-scale representation of discrete derivative approximations can be defined, so that discrete analogues of (1)-(8) *hold exactly after discretization*.

The representation to be proposed has theoretical advantages compared to traditional discretizations based on different versions of the sampled or integrated Gaussian kernel, and discretizations carried out in the frequency domain, in the sense that *operators that commute before the discretization, commute also 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 re-doing the smoothing part of the operation, which is usually the computationally most expansive part when computing smoothed derivative approximations. 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 presented of using these operations for a few different visual tasks. A straightforward edge detection scheme is described, which is similar to Canny's [6] method, but does not need any direct

---

[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.

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 (in order to eliminate "false edges"). Qualitively the results obtained are similar those of Canny, although the proposed scheme is given by a conceptually much simpler framework, and in addition has the advantage that sub-pixel accuracy is obtained automatically. It is also 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 presentation is organized as follows: Section 2 reviews the existing scale-space theory for continuous and discrete signals. This constitutes the background material for the theoretical analysis scale-space properties of different multi-scale derivative approximations carried out in Section 3. Section 4 describes some straightforward computational implications of the presented discrete theory, with comparisons made to other possible discretizations of the continuous scale-space theory. Graphical illustrations of the resulting derivative approximation kernels are given in Section 5, while 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 multi-scale 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. Of crucial importance when constructing a scale-space representation is 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 if it were just an accidental phenomenon, say an amplification of the noise, *created by the smoothing method — not the data*. Therefore, it is of out-most importance that artifacts are not introduced by the smoothing transformation when 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 to construct a multi-scale image representation.

2.1   Continuous Signals

Koenderink [15] introduced the notion of *causality*, which basically means 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. Combined with *homogeneity* and *isotropy* constraints, which essentially mean that all spatial points and all scale levels should be handled in a similar manner, they showed that these criteria by necessity and sufficiency lead to a formulation in terms of the diffusion equation, both in one and two dimensions. A similar result, although based on slightly different assumptions, was given by Yuille and Poggio [35] regarding the zero-crossings of the Laplacian.

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 *semi-group structure* on the family of convolution kernels and natural symmetry and normalization constraints also, by necessity, lead 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 semi-group 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 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, it has shown by Lindeberg [20, 21] 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 semi-discretized 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 *semi-group* of normalized and symmetric kernels having a *continuous* scale parameter. Given a discrete signal $f : Z \rightarrow \mathbb{R}$ the scale-space representation $L : \mathbb{Z} \times \mathbb{R}_+ \rightarrow \mathbb{R}$ is defined by

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

where $T : \mathbb{Z} \times \mathbb{R}_+ \rightarrow \mathbb{R}$ 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 semi-discretized version of the diffusion equation

$$(\partial_t L)(x; t) = \frac{1}{2}(\nabla_3^2 L)(x; t) = \frac{1}{2}(L(x - 1; t) - L(x; t) + L(x + 1; t)) \tag{11}$$

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, which 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.

Assuming that the scale-space is generated by convolution with a *semi-group* of symmetric convolution kernels, which are continuous with respect to their *continuous scale parameter*, it can be shown that the above non-enhancement 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 semi-discretized version of the diffusion equation

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

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 (below the (temporary) notation $f_{-1,1}$ stands for $f(x - 1, y + 1)$ etc):

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

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

The parameter γ is undetermined by this theory. In [20, 24] indications are given that it should not exceed the value $\gamma_{max} = \frac{1}{2}$. Setting $\gamma = \gamma_{symm} = \frac{1}{3}$ 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 $\gamma = \gamma_{sep} = 0$. Then, the scale-space family is obtained by applying the one-dimensional discrete analogue of the Gaussian kernel (10) along each dimension,

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

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

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

$$T_\gamma(\cdot, \cdot; t_2) = T_\gamma(\cdot, \cdot; t_2 - t_1) * T_\gamma(\cdot, \cdot; t_1). \tag{17}$$

where $t_2 > t_1 \geq 0$. Similar results hold in one 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, which are to possess scale-space properties in a discrete sense, what properties are then desirable? Let us start with the following very simple but anyway 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 the scale-space smoothing operator.

---

[5] The conditions concerning 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.

Concerning multi-scale 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

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

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.,

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

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

$$\mathcal{D}_{x^i y^j} L = \mathcal{D}_{x^i y^j} (T * f) = (\mathcal{D}_{x^i y^j} T) * f = T * (\mathcal{D}_{x^i y^j} f). \qquad (20)$$

Moreover, the discrete derivative approximation obeys the cascade smoothing property

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

and satisfies the semi-discretized version of the diffusion equation

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

In particular, the discrete derivative approximation fulfills the following non-enhancement 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

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

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

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

Proof: The validity of (20) follows directly from Observation 1 as does (21) if the lemma 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 due to (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. $\qquad \square$

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 the introduction*, 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.

---

[6] A discrete derivative approximation, $\delta_{x^i y^j}^{(h)}$ depending on a step length h, is said to be *consistent* if (under reasonable assumptions on the signal $L : \mathbb{R}^2 \to \mathbb{R}$) the truncation error tends to zero as the step length tends to zero, i.e., if $\lim_{h \downarrow 0} (\delta_{x^i y^j}^{(h)} L)(x_0, y_0) = (\partial_{x^i y^j} L)(x_0, y_0)$. In our considerations h is omitted from the notation, since the grid spacing is throughout assumed to be equal to one.

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; *linearity, translational invariance, mirror symmetry, semi-group, 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* etc. Such a list, however, will 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 non-enhancement property of local extrema. When formulating these scale-space conditions, it turns out to be natural to express the non-enhancement property in terms of a sign condition on the derivative of the scale-space representation with respect to the scale parameter. In order to ensure differentiability in this step, a series of successive definitions and propositions will be stated, which will 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

- *$D_{x^i y^j}(\cdot, \cdot;\ 0)$ is a finite support kernel corresponding to a consistent discrete approximation of the derivative operator $\partial_{x^i y^j}$, and*

- *$D_{x^i y^j}$ satisfies the cascade smoothing property*

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

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

  - *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*
  - *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

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

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

7

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 make no actual "irreversible decisions". More technically, the linearity can also be motivated by requiring the discrete derivative approximations to obey similar linearity properties as 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 in a similar manner. 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

$$\partial_t L_{x^i y^j} = \mathcal{A} L_{x^i y^j} \tag{27}$$

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

Proof: Due to the cascade smoothing property of $D_{x^i y^j}$ we have that

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

and

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

hold for any $f' : Z^2 \to R$ 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 semi-group 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). $\qquad\square$

This property makes it possible to formulate the non-enhancement 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

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

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

---

[7] Here, a *weak* notion of local extremum is used; a 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.

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 above pre-scale-space properties 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 non-enhancement 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 approximations 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 semi-discretized 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

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

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 differential equation

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

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 by 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 multi-scale representation of discrete derivative approximations $L_{x^i y^j}$ of a signal f, satisfying

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

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

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

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 semi-discretized version of the diffusion equation

$$\partial_t L_{x^i y^j} = \frac{1}{2} \nabla_\gamma^2 L_{x^i y^j} \tag{36}$$

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

The result has been expressed in a general form as 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 ∇_γ^2 is replaced by $\nabla_3^2 L$). 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

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

and the striking similarity between the discrete and continuous relations,

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

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

$$(\mathcal{D}_{xx} L)(x; t) = (\delta_{xx} L)(x; t) = L(x + 1; t) - 2L(x; t) + L(x - 1; t) \tag{39}$$

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

$$(\delta_{xx} T)(x; t) = 2(\partial_t T)(x; t), \qquad (\partial_{xx} g)(x; t) = 2(\partial_t g)(x; t) \tag{40}$$

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

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

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

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

---

[8] Note that it has nowhere in the proofs been made use 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, $\mathcal{D}_{xx} \neq \nabla_3^2$. 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, 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) $\tilde{\mathcal{D}}_x^2$. 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 independent of this choice.

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 $\gamma_{sep} = 0$, 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 similar difference operator in the y-direction as δ_{x^i} is 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

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

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

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

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

$$\tilde{\delta}_{xx} = \begin{pmatrix} a & -2a & a \\ b & -2b & b \\ a & -2a & a \end{pmatrix}, \quad \tilde{\delta}_{yy} = \begin{pmatrix} a & b & a \\ -2a & -2b & -2a \\ a & b & a \end{pmatrix}. \tag{48}$$

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

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

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

---

[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.
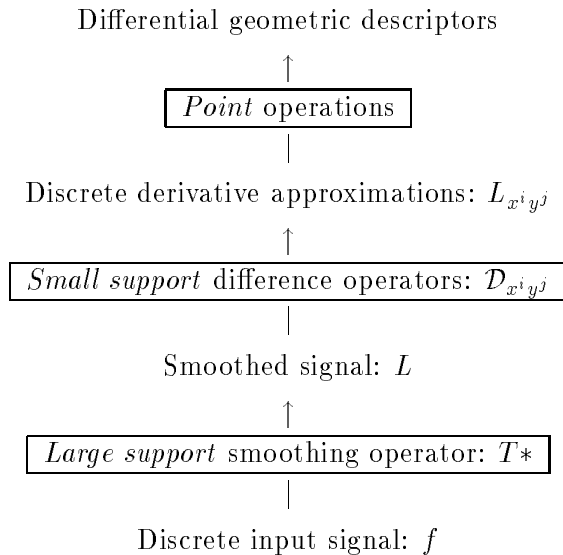
3.6   Other Possible Choices

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, 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 approximations operators, $\mathcal{D}_{x^i y^j}$, usually have a small support region and contain a small number of non-zero filter coefficients. Hence, there is absolutely no need for *re-doing* 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 when computing multi-scale differential geometric descriptors of high derivation order; see Figure 1.

<div align="center">

Differential geometric descriptors

$\uparrow$

| *Point* operations |

|

Discrete derivative approximations: $L_{x^i y^j}$

$\uparrow$

| *Small support* difference operators: $\mathcal{D}_{x^i y^j}$ |

|

Smoothed signal: L

$\uparrow$

| *Large support* smoothing operator: $T*$ |

|

Discrete input signal: f

</div>

Figure 1: Schematic overview of the different types of computations required for computing multi-scale derivative approximations and discrete approximations to differential geometric descriptors using the proposed framework.

---

[12]Similar operators have also been used in *pyramid* representations; see, e.g., Burt [5], and Crowley [7].

[13]The same problem arises also if the computations are performed in the Fourier domain, since at least one inverse FFT transformation will be needed for each derivative approximation.

4.2 Normalization of the Filter Coefficients

Below are listed 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 respectively, δ_{cont} and δ_{disc};

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

Concerning the normalization of the filter coefficients, it holds that

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

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,

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

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;

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

In practice, to give the equivalent effect of 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 $1/g(0;\ t) = \sqrt{2\pi t}$, while at $t = 0$ the discrete normalization factor it 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 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 upon 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

13

(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, while (i) and (ii) do not. Considering the particular case of the Laplacian of the Gaussian, $\nabla^2(g * f)$, it is well-known that this kernel is not separable. When 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 to why the "discrete scale-space representations" generated from different versions of the sampled Gaussian kernel does not possess discrete scale-space properties, is because when using this discretization method, the discrete correspondences to operators, which commute before the discretization, do not *commute after the discretization.*

4.4   Discrete Modelling of Feature Detectors

The proposed discrete kernels are also suitable for discrete modelling 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 modelled by

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

where t_0 describes the degree of diffuseness. In a scale-space representation, L_{t_0}, of E_{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

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

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

$$E_{t_0}(x) = \sum_{n=-\infty}^{x} T(n; \, t_0), \tag{56}$$

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,

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

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].

Discrete Gauss　　　*Sampled Gauss*　　　*Discrete Gauss*　　　*Sampled Gauss*
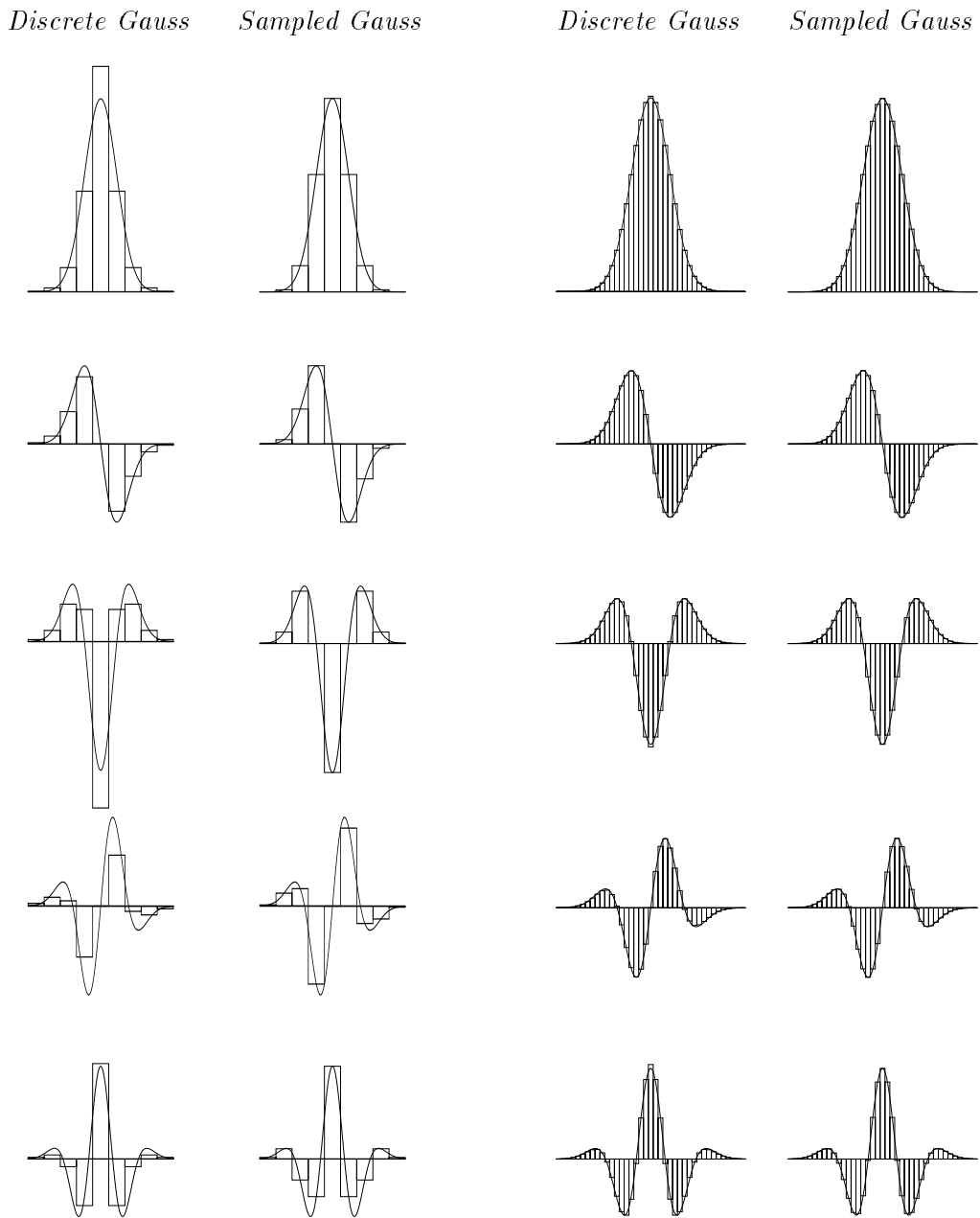
Figure 2: Comparisons between the discrete analogue of the Gaussian kernel and the sampled Gaussian kernel at scale levels $t = 1.0$ and $t = 16.0$. The columns show from left to right; the discrete Gaussian $t = 1.0$, the sampled Gaussian $t = 1.0$, the discrete Gaussian $t = 16.0$, and the sampled Gaussian $t = 16.0$. The derivative/difference order increases from top to bottom; the upper row shows the raw smoothing kernel; then follow the first, second, third and fourth order derivative/difference kernels. The block diagrams indicate the discrete kernels and the smooth curve the continuous Gaussian.
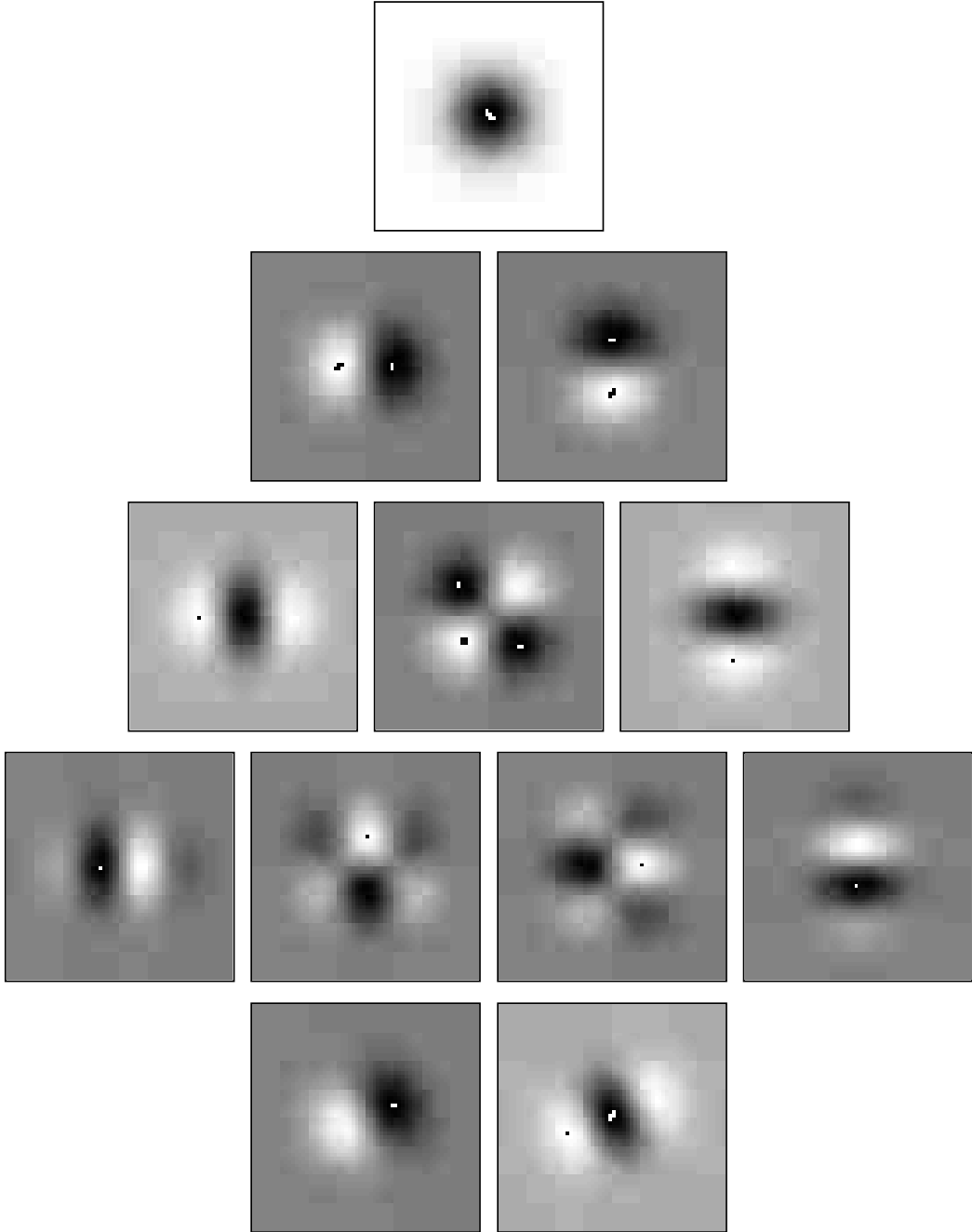
Figure 3: Grey-level illustrations of the *equivalent* two-dimensional discrete derivative approximation kernels up to order three (in the separable case corresponding to $\gamma = 0$). **(row 1)** (a) Zero-order smoothing kernel, T, (inverted). **(row 2)** (b-c) First order derivative approximation kernels, $\delta_x T$ and $\delta_y T$. **(row 3)** (d-f) Second order derivative approximation kernels $\delta_{xx} T$, $\delta_{xy} T$, $\delta_{yy} T$. **(row 4)** (g-j) Third order derivative approximation kernels $\delta_{xxx} T$, $\delta_{xxy} T$, $\delta_{xyy} T$, $\delta_{yyy} T$. **(row 5)** (k-l) First and second order directional derivative approximation kernels in the direction 22.5° computed from (59). (Scale level $t = 64.0$, image size 127×127 pixels).

5 Kernel Graphs

Figure 2 illustrates the differences and similarities between the proposed discrete kernels and the derivatives of the continuous Gaussian kernel. Shown are the graphs of

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

for a few order 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 $L_{x^i y^j}$ using the well-known expression for the nth order directional derivative $\partial_{\bar{\alpha}}^n$ of a function L in any direction α,

$$\partial_{\bar{\alpha}}^n L = (\cos \alpha \, \partial_x + \sin \alpha \, \partial_y)^n L. \tag{59}$$

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 where 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.
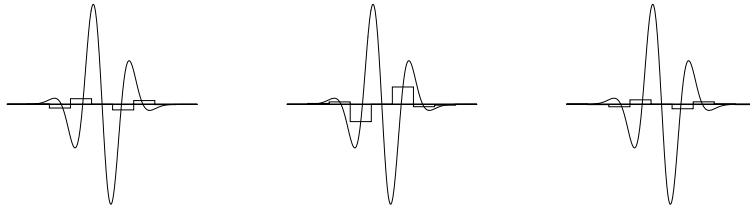
Figure 4: Different discrete approximations of the fifth order derivative of the Gaussian at a fine scale ($t = 0.46$); (a) fifth order difference of the discrete analogue of the Gaussian kernel, (b) sampled fifth order derivative of the Gaussian kernel, and (c) fifth order difference of the sampled Gaussian kernel. Observe that the slope at the origin of the kernel in (b) differs from the slopes of the other 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.

6 Experimental Results: Low-Level Feature Extraction

To demonstrate how the proposed framework can be used in low-level feature extraction, it will in this section be briefly described how a subpixel edge 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], it is developed how these discrete derivative approximations can be used as a basis for shape from texture).

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 $\partial_{\bar{v}}$ denote the directional derivative operator in the gradient direction of L at P and $\partial_{\bar{u}}$ the directional derivative in the perpendicular direction. Then, at any $P \in \mathbb{R}^2$ the gradient magnitude is equal to $\partial_{\bar{v}} L$, denoted $L_{\bar{v}}$, at that point. Assuming 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;

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

By expressing the directional derivatives in terms of derivatives with respect to the Cartesian coordinates (x, y), $\partial_{\bar{u}} = \sin \phi \partial_x - \cos \phi \partial_y$, $\partial_{\bar{v}} = \cos \phi \partial_x + \sin \phi \partial_y$, 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

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

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

where $L_{\bar{v}} = (L_x^2 + L_y^2)^{1/2}$. By reinterpreting L 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 $\tilde{L}_{\bar{v}\bar{v}}$ for which $\tilde{L}_{\bar{v}\bar{v}\bar{v}}$ 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 above discrete derivative approximations. From these in turn, discrete approximations to $\tilde{L}_{\bar{v}\bar{v}}$ and $\tilde{L}_{\bar{v}\bar{v}\bar{v}}$ can be computed as *pointwise* polynomials.
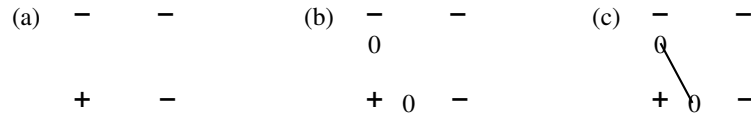
Figure 5: The subpixel edge interpolation procedure in a simple situation; a four-point cell in which $\tilde{L}_{\bar{v}\bar{v}}$ is positive in one point and negative in all the other ones, while $\tilde{L}_{\bar{v}\bar{v}\bar{v}}$ is negative in all points. (a) Sign pattern for $\tilde{L}_{\bar{v}\bar{v}}$. (b) Estimated locations of zero-crossings from linear interpolation. (c) Zero-crossings connected by a straight line segment. ('+' denotes a pixel for which $\tilde{L}_{\bar{v}\bar{v}} > 0$ and '-' a pixel for which $\tilde{L}_{\bar{v}\bar{v}} < 0$).

Of course, there do not exist any exact discrete correspondences to zero-crossing curves in the discrete case. Nevertheless, a *sub-pixel* 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 sign of $\tilde{L}_{\bar{v}\bar{v}}$, 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 between 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 intention with experiment is not to argue 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

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

As 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 a polynomial expression. Moreover, the resulting operator becomes skew invariant; see Blom [3].

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

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. In order to enhance the maxima in the output, a certain type of blob detection, *grey-level blob* detection [22, 21], has been applied 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, while at coarser scales the operator gives a stronger response in regions that correspond to meaningful junctions in the scene.

---

[14]Some care must, however, be taken with respect to the sign pattern of $\tilde{L}_{\bar{v}\bar{v}}$ and ambiguous situations; for example, when all adjacent points in a four-cell have opposite sign in $\tilde{L}_{\bar{v}\bar{v}}$).

[15]Which were derived solely from theoretical scale-space considerations.
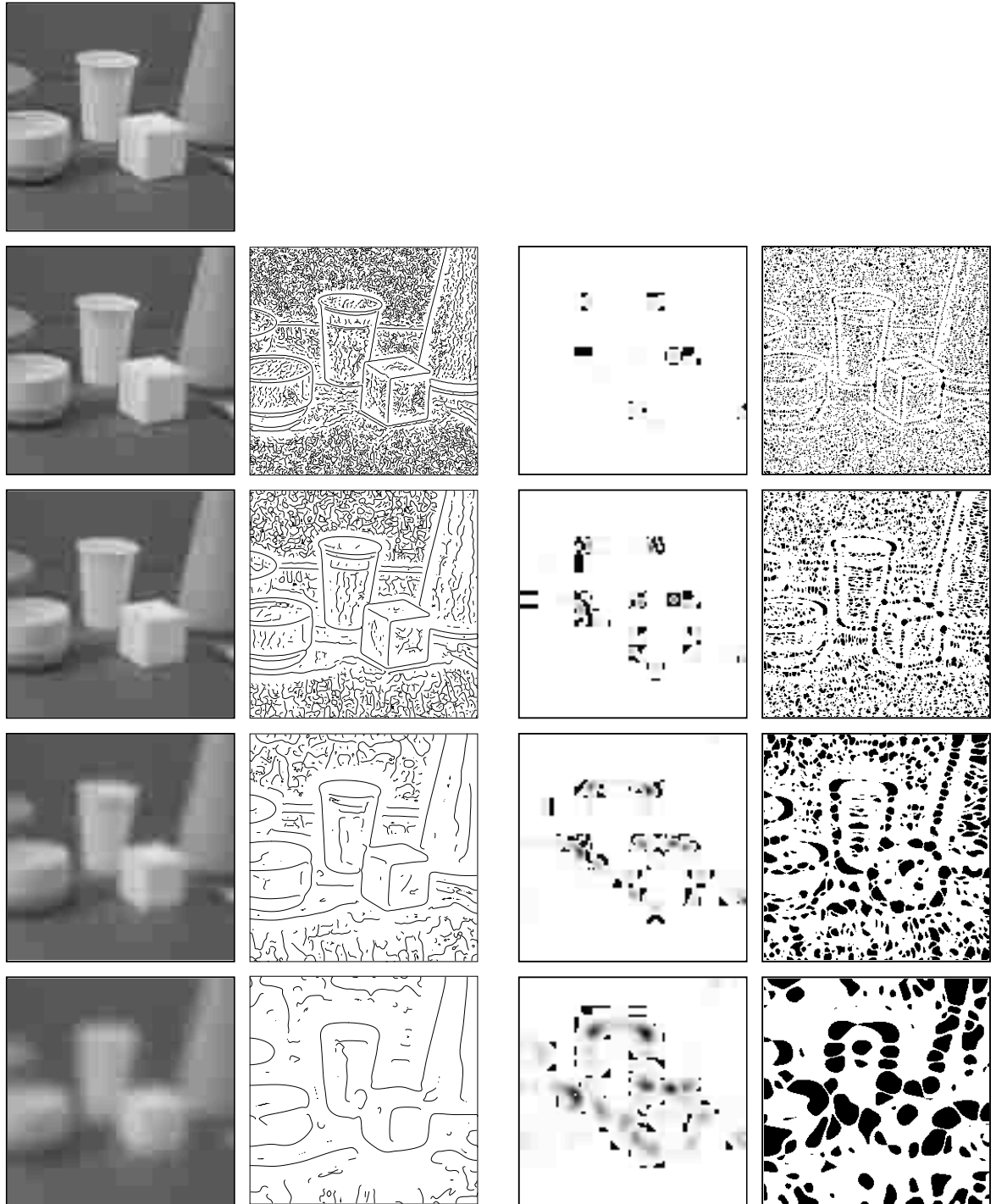
Figure 6: Edges and junctions detected by applying the presented sub-pixel edge detection scheme, and by computing (the absolute value of) the rescaled level curve curvature at scales $t = 1, 4, 16$, and 64 respectively (from top to bottom). The columns show from (left to right); the smoothed grey-level values, detected edges, the rescaled level curve curvature, and the grey-level blobs extracted from the curvature data. (No thresholding has been performed).
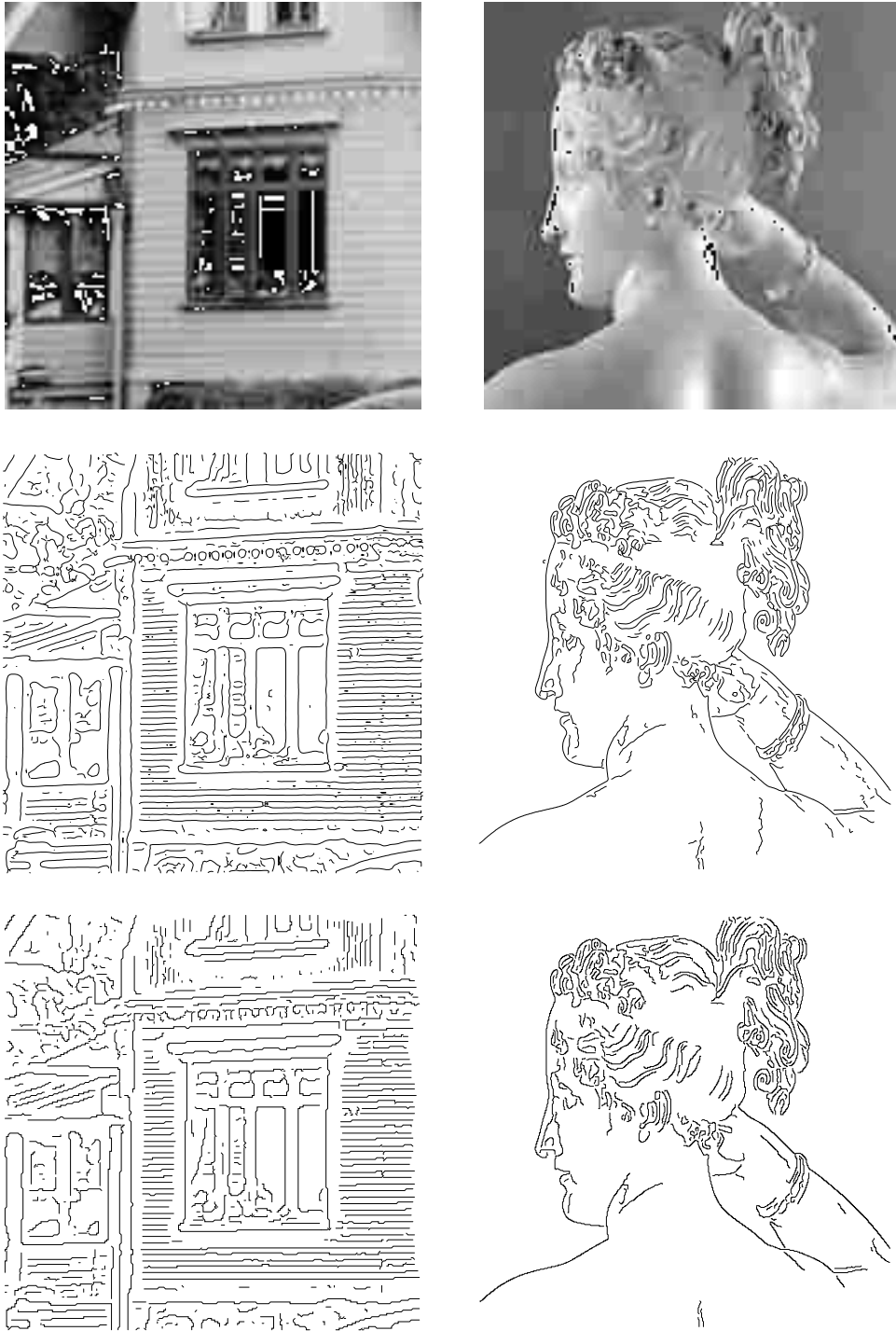
Figure 7: A comparison between the sub-pixel edge detection scheme based on discrete derivative approximations (middle row) and Canny-Deriche edge detection (bottom row). The scale values used for smoothing in the left column were t=4.0 (middle left) and α=0.7 (bottom left), while the corresponding values in the right column were t=1.4 (middle right) and α=1.2 (bottom right). (In the left column no thresholding was performed on the gradient magnitude or the length of the edge segments was performed, while 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). (Image size: left 256×256 pixels, right 512×512 pixels).

6.3 Feature Detection from Singularities in Scale-Space

The two applications above 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.

Singularities (features)

$\uparrow$

| *Nearest-neighbour* comparisons |

|

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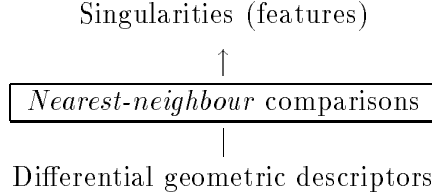
Figure 8: The low-level feature extractors have been expressed in terms of singularities of different differential geometric entities. This corresponds to the addition of one more processing step to the flow of computations illustrated in Figure 1, namely singularity detection. This operation can (at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

A main reason to why this formulation in terms of singularities is important, is because these singularities do not depend on the actual numerical values of the differential geometric entities, but 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 differential entities used above 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 $L_{\bar{u}^m \bar{v}^n} = L_{\bar{u}^\alpha}$ denote a mixed directional derivative of order $|\alpha| = m+n$, where $\alpha = (m, n)$, and let $\mathcal{D}$ be a (possibly non-linear) homogeneous differential expression of the form

$$\mathcal{D}L = \sum_{i=1}^{I} c_i \prod_{j=1}^{J} L_{\bar{u}^{\alpha_{ij}}} \quad , \qquad (65)$$

where $|\alpha_{ij}| > 0$ for all $i = [1..I]$ and $j = [1..J]$, and $\sum_{j=1}^{J} |\alpha_{ij}| = N$ for all $i \in [1..I]$. 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 follows:

Transformation	Definition			Invariance		
translation	$(\mathcal{T}L)(x; t)$	$=$	$L(x + \Delta x; t)$	$\mathcal{S}_{\mathcal{D}} \mathcal{G} \mathcal{T} f =$	$\mathcal{S}_{\mathcal{D}} \mathcal{T} \mathcal{G} f$	$= \mathcal{T} \mathcal{S}_{\mathcal{D}} \mathcal{G} f$
rotation	$(\mathcal{R}L)(x; t)$	$=$	$L(Rx; t)$	$\mathcal{S}_{\mathcal{D}} \mathcal{G} \mathcal{R} f =$	$\mathcal{S}_{\mathcal{D}} \mathcal{R} \mathcal{G} f$	$= \mathcal{R} \mathcal{S}_{\mathcal{D}} \mathcal{G} f$
uniform scaling	$(\mathcal{U}L)(x; t)$	$=$	$L(sx; t)$	$\mathcal{S}_{\mathcal{D}} \mathcal{G} \mathcal{U} f =$	$\mathcal{S}_{\mathcal{D}} \mathcal{U} \mathcal{G} f$	$= \mathcal{U} \mathcal{S}_{\mathcal{D}} \mathcal{G} f$
affine intensity	$(\mathcal{A}L)(x; t)$	$=$	$aL(x; t) + b$	$\mathcal{S}_{\mathcal{D}} \mathcal{G} \mathcal{A} f =$	$\mathcal{S}_{\mathcal{D}} \mathcal{A} \mathcal{G} f$	$= \mathcal{S}_{\mathcal{D}} \mathcal{G} f$

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,

---

[16]See also Florack et al [10] concerning necessity results regarding a similar (but not identical) set of transformations.

and it does not matter whether the transformation is performed before or after the smoothing step.

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

$$\begin{aligned}
\mathcal{TS_D}L &= \{(x;\, t) : \mathcal{D}L(x + \Delta x;\, t) = 0\}, \\
\mathcal{RS_D}L &= \{(x;\, t) : \mathcal{D}L(Rx;\, t) = 0\}, \\
\mathcal{US_D}L &= \{(x;\, t) : \mathcal{D}L(sx;\, \underline{s^2}t) = 0\}.
\end{aligned}$$

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 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 non-maximum suppression, i.e., $L_{\bar{v}\bar{v}} = 0$, the drift velocity in the normal direction of the curve, $(\alpha_{\bar{u}}, \alpha_{\bar{v}}) = (L_{\bar{v}}^2 L_{\bar{u}\bar{v}\bar{v}} + 2L_{\bar{v}}L_{\bar{u}\bar{v}}L_{\bar{u}\bar{u}},\ L_{\bar{v}}^2 L_{\bar{v}\bar{v}\bar{v}} + 2L_{\bar{v}}L_{\bar{u}\bar{v}}^2)$, is

$$(\partial_t u, \partial_t v) = -\frac{L_{\bar{v}}(L_{\bar{u}\bar{u}\bar{v}\bar{v}} + L_{\bar{v}\bar{v}\bar{v}\bar{v}}) + 2L_{\bar{u}\bar{v}}(L_{\bar{u}\bar{u}\bar{u}} + L_{\bar{u}\bar{v}\bar{v}})}{2((L_{\bar{v}}L_{\bar{u}\bar{v}\bar{v}} + 2L_{\bar{u}\bar{v}}L_{\bar{u}\bar{u}})^2 + (L_{\bar{v}}L_{\bar{v}\bar{v}\bar{v}} + 2L_{\bar{u}\bar{v}}^2)^2)}\left(\frac{\alpha_{\bar{u}}}{L_{\bar{v}}}, \frac{\alpha_{\bar{v}}}{L_{\bar{v}}}\right), \qquad (66)$$

which reflects the non-trivial 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

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

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 to use the output from these processing modules 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 first row in Figure 9 shows the effect of hysteresis thresholding on gradient magnitude in the edge detection step. The second 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 *multi-scale* 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 multi-scale analysis*, performed after the (non-linear) computation of the differential descriptor, generates output results much more useful for further processing than the single scale extraction of grey-level blobs illustrated in Figure 6; see also [4] and [27]. A more general method for scale selection is decribed in [26].

7 Summary and Discussion

The main subject of this paper has been to describe 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.
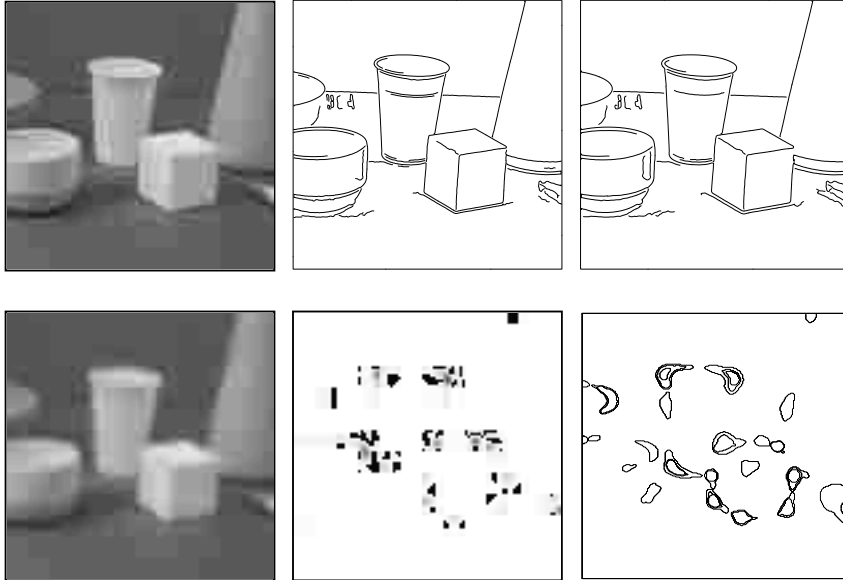
Figure 9: (row 1) Edges detected from the table scene image using a low (manually selected) hysteresis thresholds on gradient magnitude; (a) grey-level image, (b) edges detected at scale $t=2$ (low = 1.0, high = 3.0), and (c) edges detected at scale $t=4$ (low = 1.0, high = 2.0). **(row 2)** Suppression of junction candidates without thresholding on the operator response; (d) image smoothed to scale $t=8$, (e) absolute value of the rescaled level curve curvature computed at that scale, (f) boundaries of the 40 most significant curvature blobs extracted by applying a multi-scale blob detection method, the scale-space primal sketch [21, 22], to the curvature data.

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 non-enhancement property of local extrema, and the semi-group/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 re-doing the smoothing part of the transformation when computing several derivative approximations. 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.

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 upon 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; which lends itself to direct implementation in terms of natural operations in a visual front end; (i) *large support linear* smoothing, (ii) *small support linear* derivative approximations, (iii) *pointwise non-linear* computations of 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 of importance in order to improve the computational efficiency, both when 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 which extent a subsampling operator can be introduced in 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 to numerical difficulties for algorithms working on the output from the representation. For example, it is 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 concerns 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 where no information is available, the visual front-end processing should be as *uncommitted* as possible. On the other, once any (initial) knowledge is available, there is a variety of different possibilities opening up. A natural generalization to consider is *non-linear diffusion*, although further work may be needed in order to develop the notion of anisotropic smoothing as introduced into computer vision by Perona and Malik [31] and Nordstrom [30].

A   Appendix

A.1   Summary of Main Results from 2D Discrete Scale-Space Theory

Below are stated for reference purpose 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], while 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

- $T(\cdot, \cdot; \, 0) = \delta(\cdot, \cdot)$,

- the semi-group property $T(\cdot, \cdot; \, s) * T(\cdot, \cdot; \, t) = T(\cdot, \cdot; \, s + t)$,

- 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

- the continuity requirement $\| \, T(\cdot, \cdot; \, t) - \delta(\cdot, \cdot) \, \|_1 \to 0$ when $t \downarrow 0$.

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

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

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

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

$$\partial_t L = \mathcal{A} L \tag{69}$$

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 $T : Z^2 \times R_+ \to R$ is said to be a scale-space family of kernels if for any signal $f : Z^2 \to R \in l_1$ the pre-scale-space representation of f generated by T obeys the non-enhancement 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 $L : Z^2 \times R_+ \to R$ of a signal $f : Z^2 \to R$ generated by a family of kernels $T : Z^2 \times R_+ \to R$, which are scale-space kernels, is said to be a scale-space representation of f.

Theorem 17. (Scale-space for 2D discrete signals: Necessity)
A scale-space representation $L : Z^2 \times R_+ \to R$ of a signal $f : Z^2 \to R$ 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 2D discrete signals: Sufficiency)
Let $f : Z^2 \to R$ be a discrete signal in l_1 and let $L : Z^2 \times R_+ \to R$ 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]). □

---

[17] $T(x, -y; \, t) = T(x, y; \, t)$ is implied from the two other properties.

A.2 Proof of Theorem 9

The proof will be similar to the proof of Theorem 17, which was based on a number 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 due to 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 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

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

If the latter sum, B, is non-zero, 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 exist 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 non-zero, then a value of b can always be found which leads to a violation of the non-enhancement 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 $R_A = \{(x, y) \in Z^2 : (|x| \le M) \wedge (|y| \le M)\}$.

Now, the extremum point conditions (23) and (24) will be combined with Definitions 7-8 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 $(x_2, y_2) \in Z^2$ with $\max(|x_2|, |y_2|) > 1$, let $f_2 : R^2 \to R$ 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 non-zero coefficient (see Appendix A.4). Obviously, $(0, 0)$ is 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 non-zero then we can always select a value of b so that $(\partial_t L_{x^i y^j})(0, 0; 0) > 0$ and the non-enhancement 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 $T : Z^2 \times R_+ \rightarrow R$, such that $L_{x^i y^j}(\cdot, \cdot; \ t) = T(\cdot, \cdot; \ t) * L_{x^i y^j}(\cdot, \cdot; \ 0) * f(\cdot, \cdot)$. Due to 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 in analogy with the proof of Theorem 9 that the *sum of the coefficients must be zero* in $(\partial_t L_{x^i y^j})(0, 0; 0) = 4a + 4b + c$ due to the non-enhancement property of local extrema. Trivially, c must be non-positive, 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 non-negative 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 non-enhancement 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 $T : Z^2 \times R_+ \rightarrow R$. By considering the solution of (71) at time t_2 computed from the original signal f, and by considering the solution at time $t_2 - t_1$ using as input signal the solution at time t_1 computed from the original signal, it follows that T must obey the semi-group 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 $(x, y) \in Z^2$. Furthermore, the differentiability implies continuity at the origin. By replacing f by $D_{x^i y^j}(\cdot, \cdot; \ 0) * f(\cdot, \cdot)$ it follows that the solution to (33) can be written

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

Then, $D_{x^i y^j} : Z^2 \times R_+ \rightarrow R$ defined by $D_{x^i y^j}(\cdot, \cdot; \ t) = T(\cdot, \cdot; \ t) * D_{x^i y^j}(\cdot, \cdot; \ 0)$ is a pre-scale-space family of derivative kernels. Since, $L_{x^i y^j}$ satisfies the non-enhancement 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 appendix the validity of a trivial statement will be verfied, namely that given an arbitrary kernel K of finite support having at least one non-zero 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 $f_{out} = K * f_{in}$ on R. (Remark: 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*, $K * f_{in}$ 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 left-most non-zero filter coefficient in the upper-most row containing non-zero 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 upper-most 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 Derivatives 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

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

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 $\sum_{\xi \in \mathbb{Z}^N} a_\xi = 0$, as well as

(iv) the *symmetry* requirements $a_{(-\xi_1, \xi_2, ..., \xi_N)} = a_{(\xi_1, \xi_2, ..., \xi_N)}$ and $a_{P_k^N(\xi_1, \xi_2, ..., \xi_N)} = a_{(\xi_1, \xi_2, ..., \xi_N)}$ for all $\xi = (\xi_1, \xi_2, ..., \xi_N) \in \mathbb{Z}^N$ and all possible permutations P_k^N of N elements.

In 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.

References

[1] Abramowitz M., Stegun I.A. (1964) *Handbook of Mathematical Functions*, Applied Mathematics Series, **55**, National Bureau of Standards.

---

[18] In the sense that it obeys the N-dimensional analogies of Definition 6, the semi-group property, and the symmetry and continuity conditions in Definition 3.

[2] Babaud J., Witkin A.P., Baudin M., Duda R.O. (1986) "Uniqueness of the Gaussian Kernel for Scale-Space Filtering", *IEEE Trans. Patt. Anal. Machine Intell.*, 8:1, 26–33.

[3] Blom J. (1992) "Topological and Geometrical Aspects of Image Structure", *PhD thesis*, Dept. Med. Phys. Physics, Univ. Utrecht, NL-3508 Utrecht, Netherlands.

[4] Brunnström K., Lindeberg T.P, Eklundh J.-O. (1992) "Active Detection and Classification of Junctions by Foveation with a Head-Eye System Guided by the Scale-Space Primal Sketch", *Proc. 2nd Eur. Conf. Comp. Vision*, Santa Margherita Ligure, Italy, May 19-22, 701–709.

[5] Burt P.J., Adelson E.H. (1983) "The Laplacian Pyramid as a Compact Image Code", *IEEE Trans Comm.*, 9:4, 532–540.

[6] Canny J. (1986) "A Computational Approach to Edge Detection", *IEEE Trans. Patt. Anal. Machine Intell.*, 8:6, 679–698.

[7] Crowley J.L., Stern R.M. (1984) "Fast Computation of the Difference of Low Pass Transform", *IEEE Trans. Patt. Anal. Mach. Intell.*, 6:2, 212–222.

[8] Dahlquist G., Björk Å., Anderson N. (1974) *Numerical Methods*, Prentice-Hall, London.

[9] Deriche R. (1987) "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector", *Int. J. Comp. Vision*, 167–187.

[10] Florack L.M.J., ter Haar Romeny B.M., Koenderink J.J., Viergever M.A. (1991) "General Intensity Transformations and Second Order Invariants", *Proc. 7th Scand. Conf. Image Analysis*, Aalborg, Denmark, Aug 13-16, 338–345.

[11] Florack L.M.J., ter Haar Romeny B.M., Koenderink J.J., Viergever M.A. (1991) "Scale and the Differential Structure of Images", *Image and Vision Computing*, 10:6, 376–388.

[12] Freeman W., Adelson E. (1990) "The design and use of steerable filters for image analysis and wavelet decomposition", *Proc. 3rd Int. Conf. Comp. Vision*, Osaka, Japan, 406–415.

[13] Haralick, R.M (1984) "Digital Step Edges from Zero-Crossings of Second Directional Derivatives", *IEEE Trans. Patt. Anal. Mach. Intell.*, 6.

[14] Kitchen, L., Rosenfeld, R., (1982), "Gray-Level Corner Detection", *Pattern Recogn. Lett.*, 1:2, 95–102.

[15] Koenderink J.J. (1984) "The Structure of Images", *Biol. Cyb.*, Vol. 50, 363–370.

[16] Koenderink J.J., van Doorn A.J. (1987) "Representation of Local Geometry in the Visual System", *Biol. Cyb.*, Vol. 55, 367–375.

[17] Koenderink J.J., Richards W. (1988) "Two-Dimensional Curvature Operators", *J. Opt. Soc. Am.*, 5:7, 1136–1141.

[18] Koenderink J.J., van Doorn A.J. (1990) "Receptive Field Families", *Biol. Cyb.*, 63, 291–297.

[19] Korn A.F. (1988) "Toward a Symbolic Representation of Intensity Changes in Images", *IEEE Trans. Patt. Anal. Mach. Intell.*, 10:5, 610–625.

[20] Lindeberg T. (1990) "Scale-Space for Discrete Signals", *IEEE Trans. Patt. Anal. Mach. Intell.*, 12:3, 234–254.

[21] Lindeberg T. (1991) *Discrete Scale-Space Theory and the Scale-Space Primal Sketch*, Ph.D. Thesis, Royal Institute of Technology, S-100 44 Stockholm, Sweden. A revised and extended version to appear in The Kluwer International Series in Engineering and Computer Science.

[22] Lindeberg T., Eklundh J.O. (1991) "On the Computation of a Scale-Space Primal Sketch", *J. Vis. Comm. Image Repr.*, 2:1, 55–78.

[23] Lindeberg T. (1992) "Scale-Space Behaviour of Local Extrema and Blobs", *J. Math. Imaging Vision*, 1, 65–99.

[24] Lindeberg T. (1992) "Scale-Space for N-Dimensional Discrete Signals", *Proc. NATO ARW on Shape in Picture*, Driebergen, Netherlands, Sep 7-11, 1992.

[25] Lindeberg T. (1992) "Scale-Space Behaviour and Invariance Properties of Differential Singularities" *Proc. NATO ARW on Shape in Picture*, Driebergen, Netherlands, Sep 7-11, 1992.

[26] Lindeberg, T. (1992) "On Scale Selection for Differential Operators", *Proc. 8th Scand. Conf. Image Analysis*, Tromsö, Norway, to appear.

[27] Lindeberg T., Gårding J. (1993) "Shape from Texture from a Multi-Scale Perspective", *Proc. 4th Int. Conf. Comp. Vision*, Berlin, Germany, May 11-14, to appear. An extended version available as technical report from Royal Institute of Technology, S-100 44 Stockholm.

[28] Marr D., Hildreth E. (1980) "Theory of Edge Detection", *Proc. Royal Soc. London*, B:207, 187–217.

[29] Meer P., Weiss I. (1992) "Smoothed Differentiation Filters for Images", *J. Visual Comm. Image Repr.*, 3:1, 58–72.

[30] Nordström N. (1990) "Biased Anisotropic Diffusion—A Unified Regularization and Diffusion Approach to Edge Detection", *Proc. 2nd Conf. Comp. Vision*, Antibes, France, 18–27.

[31] Perona P., Malik J. (1987) "Scale Space and Edge Detection Using Anisotropic Diffusion", *Proc. IEEE Workshop on Computer Vision*, 16–22, Miami.

[32] Perona P. (1992) "Steerable-Scalable Kernels for Edge Detection and Junction Analysis", *Proc. 2nd Eur. Conf. Comp. Vision*, Santa Margherita Ligure, Italy, May 19-22, 3–18.

[33] Vieville T., Faugeras, O.D. (1992) "Robust and Fast Computation of Unbiased Intensity Derivatives in Images", *Proc. 2nd Eur. Conf. Comp, Vision*, Santa Margherita Ligure, Italy, May 19-22, 203–211.

[34] Witkin A.P. (1983) "Scale-Space Filtering", *Proc. 8:th Int. Joint Conf. Art. Intell.*, Karlsruhe, West Germany, Aug 8-12, 1019–1022.

[35] Yuille A., Poggio T. (1986) "Scaling Theorems for Zero-Crossings", *IEEE Trans. Patt. Anal. Mach. Intell.*, 9:1, 15–25.

[36] Zhang W., Bergholm F. (1991) "An Extension of Marr's Signature Based Edge Classification", *Proc. 7th Scand. Conf. Image Anal.*, Aalborg, Denmark, Aug 13-16, 435-443.