12 Edges

12.1 Introduction

The task of *edge detection* requires neighborhood operators that are sensitive to changes and suppress areas of constant gray values. In this way, a feature image is formed in which those parts of the image appear bright where changes occur while all other parts remain dark.

Mathematically speaking, an ideal edge is a discontinuity of the spatial gray value function $g(\boldsymbol{x})$ of the image plane. It is obvious that this is only an abstraction, which often does not match the reality. Thus, the first task of edge detection is to find out the properties of the edges contained in the image to be analyzed. Only if we can formulate a model of the edges, can we determine how accurately and under what conditions it will be possible to detect an edge and to optimize edge detection.

Edge detection is always based on *differentiation* in one or the other form. In discrete images, differentiation is replaced by *discrete differences*, which only approximate to differentiation. The errors associated with these approximations require careful consideration. They cause effects that are not expected in the first place. The two most serious errors are: anisotropic edge detection, i. e., edges are not detected equally well in all directions, and erroneous estimation of the direction of the edges.

While the definition of edges is obvious in scalar images, different definitions are possible in multicomponent or vectorial images (Section 12.8). An edge might be a feature that shows up in only one component or in all. Edge detection also becomes more complex in higher-dimensional images. In three dimensions, for example, volumetric regions are separated by surfaces, and edges become discontinuities in the orientation of surfaces.

Another important question is the reliability of the edge estimates. We do not only want to find an edge but also to know how significant it is. Thus, we need a measure for *edge strength*. Closely related to this issue is the question of optimum edge detection. Once edge detectors deliver not only edges but also an objective confidence measure, different edge detectors can be compared to each other and optimization of edge detection becomes possible.

B. Jähne, Digital Image Processing
ISBN 3-540-24035-7

Copyright © 2005 by Springer-Verlag
All rights of reproduction in any form reserved.

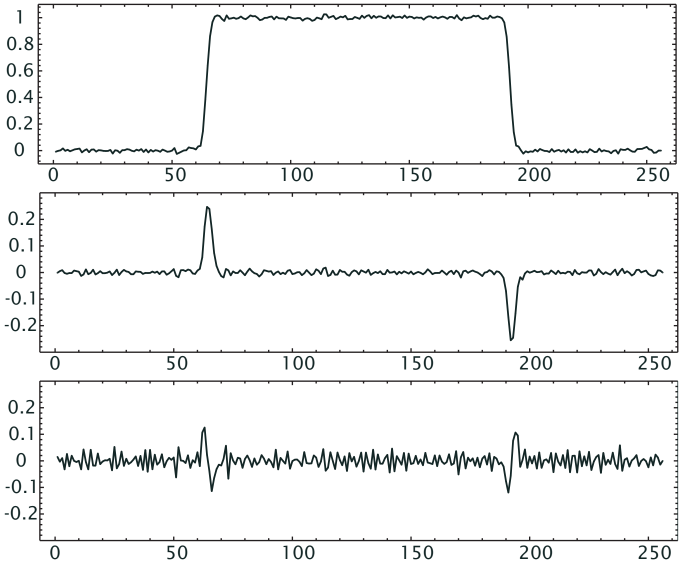

Figure 12.1: *Noisy 1-D edge and its first and second derivative.*

12.2 Differential Description of Signal Changes

Averaging filters suppress structures with high wave numbers. Edge detection requires a filter operation that emphasizes the spatial changes in signal values and suppresses areas with constant values. Figure 12.1 illustrates that derivative operators are suitable for such an operation in the one-dimensional case. The first derivative shows an extreme at the edge (maximimal positive or negative steepness), while the second derivative crosses zero (vanishing curvature) where the edge has its steepest ascent or descent. Both criteria can be used to detect edges.

In higher dimensions the description of signal change is much more complex. First, we consider 2-D images. Here we can distinguish edges, corners, lines, and local extremes as relevant features for image processing. At an *edge*, we have a large change of the signal value perpendicular to the direction of the edge. But in the direction of the edge, the change is low. However, if the curvature perpendicular to the gradient is high, the edge becomes a *corner*. A *line* is characterized by a low zero first-order derivative and second-order derivative along the line and a — in contrast to an edge — instead of the slope the curvature is high perpendicular to the direction of the line. *Local extremes* are characterized by zero first-order derivatives, but large curvatures in all directions.

In three dimensions, i. e., *volumetric images*, the situation becomes even more complex. Now there can be *surfaces* with a strong first-order change in the direction perpendicular to the surface and low slopes and curvatures in the two directions within the surface. At an edge, there are low signal changes only in the direction of the edge, and at a corner, the signal changes in all directions.

Because of this rich set of differential features to describe local changes in multi-dimensional signals, it is worthwhile to take a closer look at the basic mathematical properties of derivative operators, before we construct proper neighborhood operators to detect these features.

12.2.1 First-order Derivation and the Gradient

A pth-order *partial derivative* operator corresponds to multiplication by $(2\pi \mathrm{i} k)^p$ in the wave number space (Section 2.3, ≻ R4):

$$\boxed{\frac{\partial}{\partial x_w} \quad \circ\!\!-\!\!\bullet \quad 2\pi \mathrm{i} k_w, \qquad \frac{\partial^2}{\partial x_w^2} \quad \circ\!\!-\!\!\bullet \quad -4\pi^2\, k_w^2.} \tag{12.1}$$

The first-order partial derivatives into all directions of a W-dimensional signal form the W-dimensional *gradient vector*:

$$\boxed{\boldsymbol{\nabla} = \left[\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \ldots, \frac{\partial}{\partial x_W}\right]^T \quad \circ\!\!-\!\!\bullet \quad 2\pi \mathrm{i} \boldsymbol{k}.} \tag{12.2}$$

Under a rotation of the coordinate system, the gradient operator transforms as any other vector by multiplication with an orthogonal rotation matrix $\boldsymbol{R}$ (Section 7.2.2):

$$\boldsymbol{\nabla}' = \boldsymbol{R}\boldsymbol{\nabla}. \tag{12.3}$$

The first-order derivation in a specific direction, the so called *directional derivate* [15] is given as the scalar product between the gradient and a unit vector $\bar{\boldsymbol{n}}$, pointing in this direction:

$$\frac{\partial}{\partial \bar{\boldsymbol{n}}} = \boldsymbol{\nabla}^T \bar{\boldsymbol{n}}. \tag{12.4}$$

The magnitude of the gradient vector,

$$|\boldsymbol{\nabla}| = \|\boldsymbol{\nabla}\|_2 = \left(\boldsymbol{\nabla}^T \boldsymbol{\nabla}\right)^{1/2} = \left(\sum_{w=1}^{W} \left(\frac{\partial}{\partial x_w}\right)^2\right)^{1/2}, \tag{12.5}$$

is invariant to rotation of the coordinate system. If we rotate the coordinate system so that the gradient vector is parallel to the direction of the new x' axis, all other components of the gradient vector vanish and the directional derivate in this direction reaches a maximal value and is equal to the magnitude of the gradient vector.

12.2.2 Second-order Derivation and Curvature

Second-order derivatives detect *curvature*. All possible combinations of second-order partial differential operators of a W-dimensional signal form a symmetric $W \times W$ matrix, known as the *Hessian matrix*:

$$\boxed{\boldsymbol{H} = \begin{bmatrix} \dfrac{\partial^2}{\partial x_1^2} & \dfrac{\partial^2}{\partial x_1 x_2} & \cdots & \dfrac{\partial^2}{\partial x_1 x_W} \\ \dfrac{\partial^2}{\partial x_1 x_2} & \dfrac{\partial^2}{\partial x_2^2} & \cdots & \dfrac{\partial^2}{\partial x_2 x_W} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2}{\partial x_1 x_W} & \dfrac{\partial^2}{\partial x_2 x_W} & \cdots & \dfrac{\partial^2}{\partial x_W^2} \end{bmatrix} \quad \circ\!\!-\!\!\!-\!\!\bullet \quad -4\pi^2 \boldsymbol{k}\,\boldsymbol{k}^T.} \tag{12.6}$$

Under a rotation of the coordinate system, the Hessian matrix transforms by pre- and post-multiplication with an orthogonal rotation matrix $\boldsymbol{R}$

$$\boldsymbol{H}' = \boldsymbol{R} \nabla \boldsymbol{R}^T. \tag{12.7}$$

As we have already discussed at the end of Section 3.3.3 it is always possible to find a coordinate transform $\boldsymbol{R}$ into the *principal coordinate system* so that the Hessian matrix becomes diagonal:

$$\boldsymbol{H}' = \begin{bmatrix} \dfrac{\partial^2}{\partial {x'_1}^2} & 0 & \cdots & 0 \\ 0 & \dfrac{\partial^2}{\partial {x'_2}^2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dfrac{\partial^2}{\partial {x'_W}^2} \end{bmatrix}. \tag{12.8}$$

The gradient has only one nonzero component in the principal coordinate system. This is not the case for curvatures. Generally, *all* curvatures are nonzero in the principal coordinate system.

The trace of this matrix, i. e., the sum of the diagonal is called the *Laplacian operator* and is denoted by Δ:

$$\boxed{\Delta = \operatorname{trace} \boldsymbol{H} = \sum_{w=1}^{W} \frac{\partial^2}{\partial x_w^2} \quad \circ\!\!-\!\!\!-\!\!\bullet \quad -4\pi^2 \sum_{w=1}^{W} k_w^2 = -4\pi k^2.} \tag{12.9}$$

Because the Laplace operator is the trace of the Hessian matrix, it is invariant to rotation of the coordinate system.

12.3 General Properties of Edge Filters

In Sections 12.3.1–12.3.5, we discuss the general properties of filters that form the basis of edge detection. This discussion is similar to that on the general properties of averaging filters in Sections 11.2.1–11.2.4.

12.3.1 Zero Shift

With respect to object detection, the most important feature of a derivative convolution operator is that it must not shift the object position. For a smoothing filter, this constraint required a real transfer function and a symmetric convolution mask (Section 11.2.1). For a first-order derivative filter, a real transfer function makes no sense, as extreme values should be mapped onto zero crossings and the steepest slopes to extreme values. This mapping implies a 90° phase shift. Therefore, the transfer function of a first-order derivative filter must be imaginary. An imaginary transfer function implies an antisymmetric filter mask. An antisymmetric convolution mask is defined as

$$h_{-n} = -h_n. \tag{12.10}$$

For a convolution mask with an odd number of coefficients, this implies that the central coefficient is zero.

A second-order derivative filter detects curvature. Extremes in function values should coincide with extremes in curvature. Consequently, a second-order derivative filter should be symmetric, like a smoothing filter. All the symmetric filter properties discussed for smoothing filters also apply to these filters (Section 11.2.1).

12.3.2 Suppression of Mean Value

A derivative filter of any order must not show response to constant values or an offset in a signal. This condition implies that the sum of the coefficients must be zero and that the transfer function is zero for a zero wave number:

$$\begin{array}{lll}\text{1-D:} & \hat{h}(0) = 0, & \sum\limits_n h_n = 0 \\ \text{2-D:} & \hat{h}(\mathbf{0}) = 0, & \sum\limits_m \sum\limits_n h_{mn} = 0 \\ \text{3-D:} & \hat{h}(\mathbf{0}) = 0, & \sum\limits_l \sum\limits_m \sum\limits_n h_{lmn} = 0. \end{array} \tag{12.11}$$

Also, a second-order derivative filter should not respond to a constant slope. This condition implies no further constraints as it can be derived from the symmetry of the filter and the zero sum condition Eq. (12.11).

12.3.3 Symmetry Properties

The symmetry properties deserve further consideration as they form the basis for computing the convolution more efficiently by reducing the number of multiplications and simplifying the computations of the transfer functions. The zero-shift condition (Section 12.3.1) implies that a first-order derivative filter generally has a 1-D mask of odd symmetry with $2R + 1$ or $2R$ coefficients:

$$[h_R, \ldots, h_1, 0, -h_1, \ldots, -h_R] \quad \text{or} \quad [h_R, \ldots, h_1, -h_1, \ldots, -h_R]\,. \tag{12.12}$$

Therefore, the computation of the convolution reduces to

$$g'_n = \sum_{n'=1}^{R} h_{n'}(g_{n-n'} - g_{n+n'}) \quad \text{or} \quad g'_{n+1/2} = \sum_{n'=1}^{R} h_{n'}\left(g_{n+1-n'} - g_{n+n'}\right). \tag{12.13}$$

For $2R+1$ ($2R$) filter coefficients only R multiplications are required. The number of additions, however, is still $2R - 1$.

The symmetry relations also significantly ease the computation of the transfer functions because only the sine terms of the complex exponential from the Fourier transform remain in the equations. The transfer functions for a 1-D odd mask is

$$\hat{g}(\tilde{k}) = 2\mathrm{i}\sum_{v=1}^{R} h_v \sin(v\pi\tilde{k}) \quad \text{or} \quad \hat{g}(\tilde{k}) = 2\mathrm{i}\sum_{v=1}^{R} h_v \sin[(v - 1/2)\pi\tilde{k}]. \tag{12.14}$$

For second-order derivative filters, we can use all the equations derived for the averaging filters in Section 11.2.1, as these feature an even symmetry in the direction of the deviation.

12.3.4 Nonselective Derivation

Intuitively, we expect that a derivative operator amplifies smaller scales more strongly than coarser scales, because according to Eq. (12.1) the transfer function of an ideal pth-order derivative operator into the direction w goes with $(2\pi \mathrm{i} k_w)^p$. Consequently, we could argue that the transfer function of a good discrete derivative operator should approximate the ideal transfer functions in Eq. (12.1) as close as possible.

However, this condition is too strong a restriction. The reason is the following. Imagine that we first apply a smoothing operator to an image before we apply a derivative operator. We would still recognize the joint operation as a derivation. The mean gray value is suppressed and the operator is still only sensitive to spatial gray value changes.

Therefore, the ideal transfer function in Eq. (12.1) could be restricted to small wave numbers by expanding the transfer function into a Taylor series at wave number zero. This leads to the following conditions for a pth-order derivative 1-D operator:

$$\left.\frac{\partial^{p'}\hat{h}(\tilde{k})}{\partial\tilde{k}^{p'}}\right|_{\tilde{k}=0} = (\imath\pi)^p p!\,\delta_{p-p'} \quad \text{mit} \quad p' \le p+1. \tag{12.15}$$

In two dimensions, we need to distinguish between the x and y directions:

$$\begin{aligned} x: &\quad \left.\frac{\partial^{r+s}\hat{h}(\tilde{\boldsymbol{k}})}{\partial\tilde{k}_1^r\partial\tilde{k}_2^s}\right|_{\tilde{\boldsymbol{k}}=\mathbf{0}} = (\imath\pi)^p p!\,\delta_{p-r}\delta_s \quad \text{mit} \quad r+s \le p+1, \\ y: &\quad \left.\frac{\partial^{r+s}\hat{h}(\tilde{\boldsymbol{k}})}{\partial\tilde{k}_1^r\partial\tilde{k}_2^s}\right|_{\tilde{\boldsymbol{k}}=\mathbf{0}} = (\imath\pi)^p p!\,\delta_r\delta_{p-s} \quad \text{mit} \quad r+s \le p+1. \end{aligned} \tag{12.16}$$

These conditions can be transformed into the spatial domain by applying the momentum theorem of the Fourier transform (≻ R4). Equation (12.15) for 1-D derivative operators transforms to

$$\sum_n n^{p'} h_n = p!\,\delta_{p-p'} \tag{12.17}$$

and Eq. (12.16) for a 2-D derivative operator to

$$\begin{aligned} x: &\quad \sum_n\sum_m n^r m^s h_{n,m} = p!\,\delta_{p-r}\delta_s, \\ y: &\quad \sum_n\sum_m n^r m^s h_{n,m} = p!\,\delta_r\delta_{p-s}. \end{aligned} \tag{12.18}$$

As an example, for a two-dimensional second-order derivate operator in x direction, these conditions result in

$$\begin{array}{lll} \displaystyle\sum_n\sum_m h_{n,m} = 0, & \displaystyle\sum_n\sum_m n h_{n,m} = 0, & \displaystyle\sum_n\sum_m m h_{n,m} = 0, \\ \displaystyle\sum_n\sum_m nm h_{n,m} = 0, & \displaystyle\sum_n\sum_m n^2 h_{n,m} = 2, & \displaystyle\sum_n\sum_m m^2 h_{n,m} = 0, \\ \displaystyle\sum_n\sum_m n^2 m h_{n,m} = 0, & \displaystyle\sum_n\sum_m n m^2 h_{n,m} = 0. & \end{array} \tag{12.19}$$

These conditions include the suppression of the mean value as discussed in Section 12.3.2 and also force the symmetry conditions that result from the zero-shift property (Section 12.3.1).

12.3.5 Isotropy

For good edge detection, it is important that the response of the operator does not depend on the direction of the edge. If this is the case, we speak of an *isotropic edge detector*. The isotropy of an edge detector can best be analyzed by its transfer function. The most general form for an isotropic derivative operator of order p is given by

$$\hat{h}(\boldsymbol{k}) = (2\pi \mathrm{i} k_w)^p \hat{b}(|\boldsymbol{k}|) \quad \text{with} \quad \hat{b}(0) = 1 \quad \text{and} \quad \nabla_k \hat{b}(|\boldsymbol{k}|) = \boldsymbol{0}. \qquad (12.20)$$

The constraints for derivative operators are summarized in Appendix A (≻ R24 and ≻ R25).

12.4 Gradient-Based Edge Detection

12.4.1 Principle

In terms of first-order changes, an edge is defined as an extreme (Fig. 12.1). Thus edge detection with first-order derivative operators means to search for the steepest changes, i. e., maxima of the magnitude of the gradient vector (Eq. (12.2)). Therefore, first-order partial derivatives in all directions must be computed. In the operator notation, the gradient can be written as a vector operator. In 2-D and 3-D space this is

$$\mathcal{D} = \begin{bmatrix} \mathcal{D}_x \\ \mathcal{D}_y \end{bmatrix} \quad \text{or} \quad \mathcal{D} = \begin{bmatrix} \mathcal{D}_x \\ \mathcal{D}_y \\ \mathcal{D}_z \end{bmatrix}. \qquad (12.21)$$

Because the gradient is a vector, its magnitude (Eq. (12.5)) is invariant upon rotation of the coordinate system. This is a necessary condition for isotropic edge detection. The computation of the magnitude of the gradient can be expressed in the 2-D space by the operator equation

$$|\mathcal{D}| = \left[\mathcal{D}_x \cdot \mathcal{D}_x + \mathcal{D}_y \cdot \mathcal{D}_y\right]^{1/2}. \qquad (12.22)$$

The symbol $\cdot$ denotes a pointwise multiplication of the images that result from the filtering with the operators $\mathcal{D}_x$ and $\mathcal{D}_y$, respectively (Section 4.1.4). Likewise, the square root is performed pointwise in the space domain. According to Eq. (12.22), the application of the operator $|\mathcal{D}|$ to the image $\boldsymbol{G}$ means the following chain of operations:

1. filter the image $\boldsymbol{G}$ independently with $\mathcal{D}_x$ and $\mathcal{D}_y$,
2. square the gray values of the two resulting images,
3. add the resulting images, and
4. compute the square root of the sum.

At first glance it appears that the computation of the magnitude of the gradient is computationally expensive. Therefore it is often approximated by

$$|\mathcal{D}| \approx |\mathcal{D}_x| + \left|\mathcal{D}_y\right|. \qquad (12.23)$$

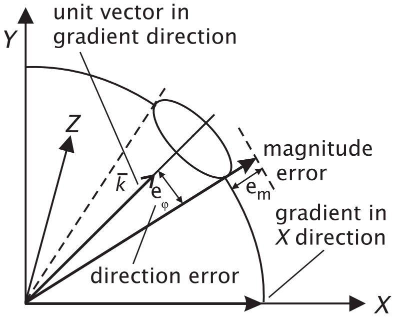

Figure 12.2: *Illustration of the magnitude and direction error of the gradient vector.*

However, this approximation is anisotropic even for small wave numbers. It detects edges along the diagonals $\sqrt{2}$ times more sensitively than along the principal axes. The computation of the magnitude of the gradient can, however, be performed as a *dyadic point operator* efficiently by a *look-up table* (Section 10.7.2).

12.4.2 Error in magnitude and direction

The principal problem with all types of edge detectors is that on a discrete grid a derivative operator can only be approximated. In general, two types of errors result from this approximation (Fig. 12.2). First, edge detection will become anisotropic, i. e., the computation of the magnitude of the gradient operator depends on the direction of the edge. Second, the direction of the edge deviates from the correct direction. For both types of errors it is useful to introduce error measures. All error measures are computed from the transfer functions of the gradient filter operator.

The magnitude of the gradient is then given by

$$\left|\hat{\boldsymbol{d}}(\boldsymbol{k})\right| = \left(\hat{d}_x(\boldsymbol{k})^2 + \hat{d}_y(\boldsymbol{k})^2\right)^{1/2}, \tag{12.24}$$

where $\hat{\boldsymbol{d}}(\boldsymbol{k})$ is the vectorial transfer function of the gradient operator. The anisotropy in the magnitude of the gradient can then be expressed by the deviation of the magnitude from the magnitude of the gradient in x direction, which is given by

$$e_m(\boldsymbol{k}) = \left|\hat{\boldsymbol{d}}(\boldsymbol{k})\right| - \left|\hat{d}_x(\boldsymbol{k})\right|. \tag{12.25}$$

This error measure can be used for signals of any dimension.

In a similar way, the error in the direction of the gradient can be computed. From the components of the gradient, the computed angle

ϕ' of the 2-D gradient vector is

$$\phi' = \arctan \frac{\hat{d}_y(k,\phi)}{\hat{d}_x(k,\phi)}. \tag{12.26}$$

The error in the angle is therefore given by

$$e_\phi(k,\phi) = \arctan \frac{\hat{d}_y(k,\phi)}{\hat{d}_x(k,\phi)} - \phi. \tag{12.27}$$

In higher dimensions, angle derivation can be in different directions. Even so we can find a direction error by using the scalar product between a unit vector in the direction of the true gradient vector and the computed gradient vector $\hat{\boldsymbol{d}}(k)$ (Fig. 12.2):

$$\cos e_\varphi = \frac{\bar{\boldsymbol{k}}^T \hat{\boldsymbol{d}}(\boldsymbol{k})}{\left|\hat{\boldsymbol{d}}(\boldsymbol{k})\right|} \quad \text{with} \quad \bar{\boldsymbol{k}} = \frac{\boldsymbol{k}}{|\boldsymbol{k}|}. \tag{12.28}$$

In contrast to the angle error measure (Eq. (12.27)) for two dimensions, this error measure has only positive values. It is a scalar and thus cannot give the direction of the deviation.

A wide variety of solutions for edge detectors exist. We will discuss some of them carefully in Sections 12.4.3–12.6.

12.4.3 First-Order Discrete Differences

First-order discrete differences are the simplest of all approaches to compute a gradient vector. For the first partial derivative in the x direction, one of the following approximations for $\partial g(x_1,x_2)/\partial x_1$ may be used:

$$\begin{array}{ll} \text{Backward difference} & \dfrac{g(x_1,x_2) - g(x_1 - \Delta x_1, x_2)}{\Delta x_1} \\[2ex] \text{Forward difference} & \dfrac{g(x_1 + \Delta x_1, x_2) - g(x_1,x_2)}{\Delta x_1} \\[2ex] \text{Symmetric difference} & \dfrac{g(x_1 + \Delta x_1, x_2) - g(x_1 - \Delta x_1, x_2)}{2\Delta x_1}. \end{array} \tag{12.29}$$

These approximations correspond to the filter masks

$$\begin{array}{llll} \text{Backward} & {}^{-}\boldsymbol{D}_x & = & [1_\bullet \; -1] \\ \text{Forward} & {}^{+}\boldsymbol{D}_x & = & [1 \; -1_\bullet] \\ \text{Symmetric} & \boldsymbol{D}_{2x} & = & 1/2\;[1 \; 0 \; -1]. \end{array} \tag{12.30}$$

The subscript • denotes the central pixel of the asymmetric masks with two elements. Only the last mask shows the symmetry properties required in Section 12.3.3. We may also consider the two-element masks

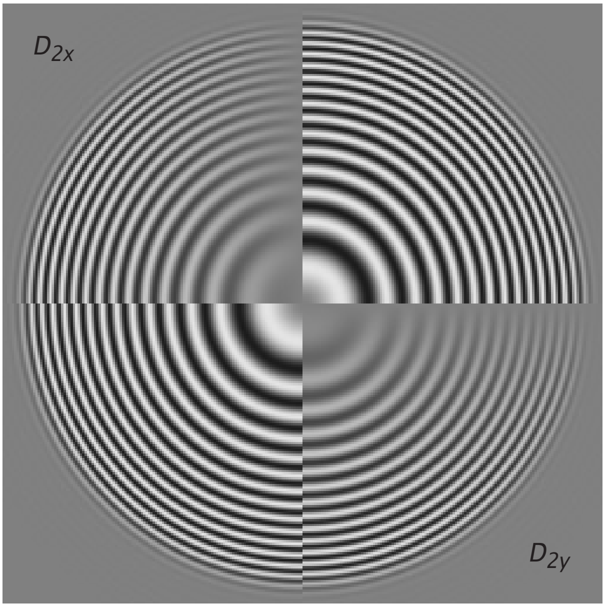

Figure 12.3: *Application of the first-order symmetric derivative filters $\mathcal{D}_x$ and $\mathcal{D}_y$ to the test image shown in Fig. 11.4.*

corresponding to the backward or forward difference as odd masks provided that the result is not stored at the position of the right or left pixel but at a position halfway between the two pixels. This corresponds to a shift of the grid by half a pixel distance. The transfer function for the backward difference is then

$$^{-}\hat{d}_x = \exp(\mathrm{i}\pi\tilde{k}_x/2)\left[1 - \exp(-\mathrm{i}\pi\tilde{k}_x)\right] = 2\mathrm{i}\sin(\pi\tilde{k}_x/2), \qquad (12.31)$$

where the first term results from the shift by half a grid point.

Using Eq. (12.14), the transfer function of the symmetric difference operator reduces to

$$\hat{d}_{2x} = \mathrm{i}\sin(\pi\tilde{k}_x) = \mathrm{i}\sin(\pi\tilde{k}\cos\phi). \qquad (12.32)$$

This operator can also be computed from

$$\boldsymbol{D}_{2x} = {}^{-}\boldsymbol{D}_x\, {}^{1}\boldsymbol{B}_x = [1_\bullet\ \ -1] * 1/2\ \ [1\ 1_\bullet] = 1/2\,[1\ 0\ -1]\,.$$

The first-order difference filters in other directions are given by similar equations. The transfer function of the symmetric difference filter in y direction is, e. g., given by

$$\hat{d}_{2y} = \mathrm{i}\sin(\pi\tilde{k}_y) = \mathrm{i}\sin(\pi\tilde{k}\sin\phi). \qquad (12.33)$$

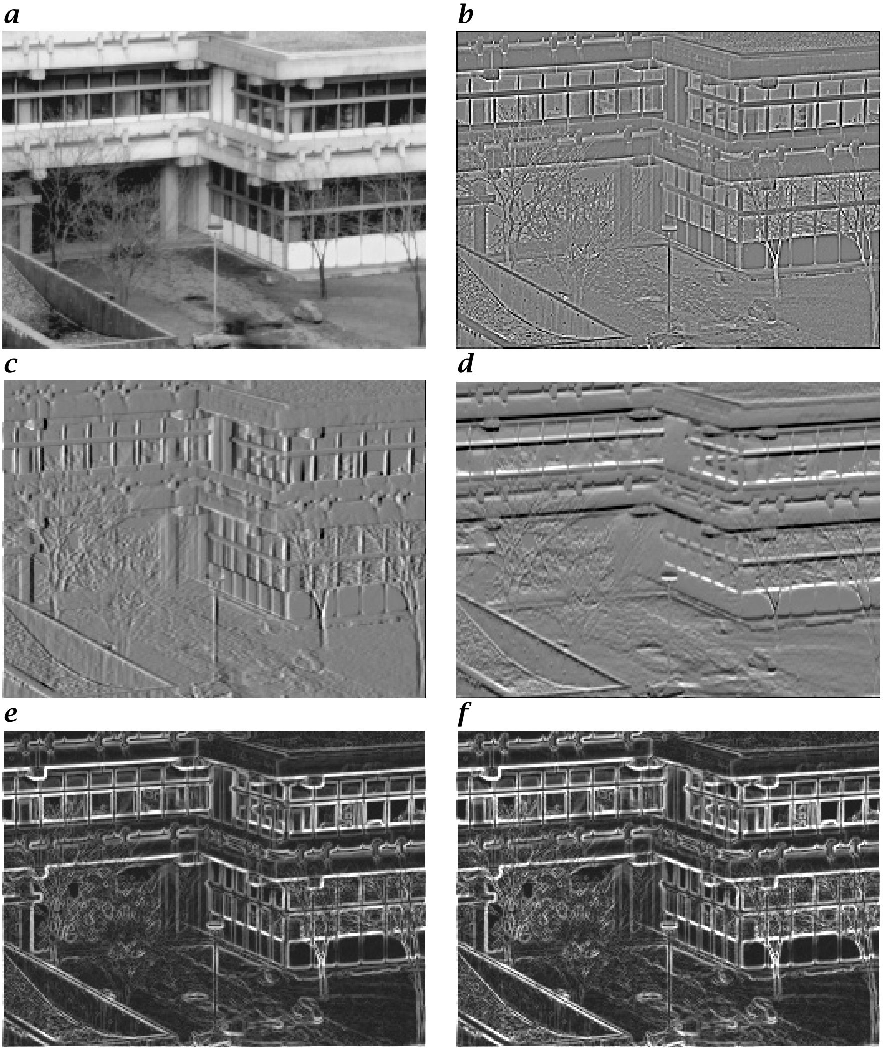

Figure 12.4: Detection of edges by derivative filters: ***a*** *Original image,* ***b*** *Laplacian operator $\mathcal{L}$,* ***c*** *horizontal derivative $\mathcal{D}_{2x}$,* ***d*** *vertical derivative $\mathcal{D}_{2y}$,* ***e*** *magnitude of the gradient $(\mathcal{D}_{2x} \cdot \mathcal{D}_{2x} + \mathcal{D}_{2y} \cdot \mathcal{D}_{2y})^{1/2}$, and* ***f*** *sum of the magnitudes of* ***c*** *and* ***d*** *after Eq. (12.23).*

The application of $\mathcal{D}_{2x}$ to the ring test pattern in Fig. 12.3 illustrates the directional properties and the 90° phase shift of these filters. Figure 12.4 shows the detection of edges with these filters, the magnitude of the gradient, and the sum of the magnitudes of $\mathcal{D}_{2x}$ and $\mathcal{D}_{2y}$.

Unfortunately, these simple difference filters are only poor approximations for an edge detector. From Eqs. (12.32) and (12.33), we infer

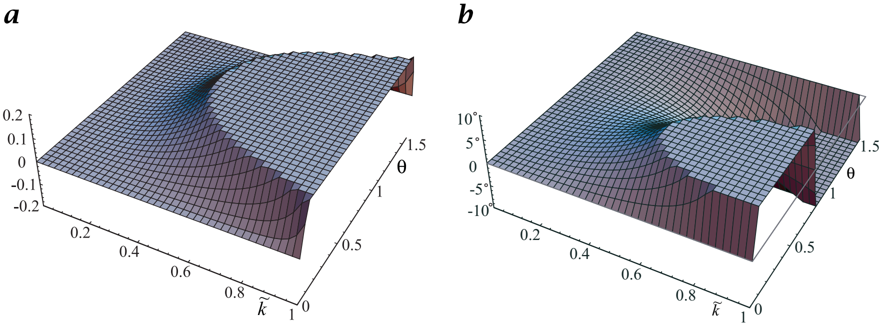

Figure 12.5: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the symmetrical gradient operator* $\left[\mathcal{D}_{2x}, \mathcal{D}_{2y}\right]^T$*. The parameters are the magnitude of the wave number (0 to 1) and the angle to the* x *axis (0 to* $\pi/2$*).*

that the magnitude and direction of the gradient are given by

$$|\hat{\boldsymbol{d}}| = \left(\sin^2(\pi\tilde{k}\cos\phi) + \sin^2(\pi\tilde{k}\sin\phi)\right)^{1/2} \qquad (12.34)$$

and

$$\phi' = \arctan\frac{\sin^2(\pi\tilde{k}\sin\phi)}{\sin(\pi\tilde{k}\cos\phi)}, \qquad (12.35)$$

where the wave number is written in polar coordinates (k, ϕ). The resulting errors are shown in a pseudo 3-D plot in Fig. 12.5 as a function of the magnitude of the wave number and the angle to the x axis. The magnitude of the gradient decreases quickly from the correct value. A Taylor expansion of Eq. (12.34) in $\tilde{k}$ yields for the relative error in the magnitude

$$e_m(\tilde{k}, \phi) \approx \frac{(\pi\tilde{k})^3}{12}\sin^2 2\phi + O(\tilde{k}^5). \qquad (12.36)$$

The decrease is also anisotropic; it is slower in the diagonal direction. The errors in the direction of the gradient are also large (Fig. 12.5b). While in the direction of the axes and diagonals the error is zero, in the directions in between it reaches values of about ± 10° at $\tilde{k} = 0.5$. A Taylor expansion of Eq. (12.35) in $\tilde{k}$ yields the angle error according to Eq. (12.27) in the approximation for small $\tilde{k}$:

$$e_\phi(\tilde{k}, \phi) \approx \frac{(\pi\tilde{k})^2}{24}\sin 4\phi + O(\tilde{k}^4). \qquad (12.37)$$

As observed in Fig. 12.5b, the angle error is zero for $\phi = n\pi/4$ with $n \in \mathbb{Z}$, i. e., for $\phi = 0°, 45°\ 90°, \ldots$

12.4.4 Spline-Based Edge Detection

The cubic B-spline transform discussed in Section 10.6.1 for interpolation yields a continuous representation of a discrete image that is also continuous in its first and second derivative:

$$g_3(x) = \sum_n c_n \beta_3(x - n), \tag{12.38}$$

where $\beta_3(x)$ is the cubic B-spline function defined in Eq. (10.51). From this continuous representation, it is easy to compute the spatial derivative of $g_3(x)$:

$$\frac{\partial g_3(x)}{\partial x} = \sum_n c_n \frac{\partial \beta_3(x - n)}{\partial x}. \tag{12.39}$$

For a discrete derivative filter, we only need the derivatives at the grid points. From Fig. 10.20a it can be seen that the cubic B-spline function covers at most 5 grid points. The maximum of the spline function occurs at the central grid point. Therefore, the derivative at this point is zero. It is also zero at the two outer grid points. Thus, the derivative is only unequal to zero at the direct left and right neighbors of the central point. Therefore, the derivative at the grid point x_m reduces to

$$\left.\frac{\partial g_3(x)}{\partial x}\right|_{x_m} = (c_{m+1} - c_{m-1})/2. \tag{12.40}$$

Thus the computation of the first-order derivative based on the cubic B-spline transformation is indeed an efficient solution. We apply first the cubic B-spline transform in the direction of the derivative to be computed (Section 10.6.1) and then the $\mathcal{D}_{2x}$ operator. Therefore, the transfer function is given by

$$\hat{\boldsymbol{D}}_x = \mathrm{i}\frac{\sin(\pi\tilde{k}_x)}{2/3 + 1/3\cos(\pi\tilde{k}_x)} = \mathrm{i}\pi\tilde{k}_x - \mathrm{i}\frac{\pi^5\tilde{k}_x^5}{180} + O(\tilde{k}_x^7). \tag{12.41}$$

The errors in the magnitude and direction of a gradient vector based on the B-spline derivative filter are shown in Fig. 12.6. They are considerably less than for the simple difference filters (Fig. 12.5). This can be seen more quantitatively from Taylor expansions for the relative errors in the magnitude of the gradient

$$e_m(\tilde{k}, \phi) \approx -\frac{(\pi\tilde{k})^5}{240}\sin^2 2\phi + O(\tilde{k}^7) \tag{12.42}$$

and the angle error

$$e_\phi(\tilde{k}, \phi) \approx \frac{(\pi\tilde{k})^4}{720}\sin 4\phi + O(\tilde{k}^6). \tag{12.43}$$

The error terms are now contained only in terms with $\tilde{k}^4$ (and higher powers of $\tilde{k}$). Compare also Eqs. (12.42) and (12.43) with Eqs. (12.36) and (12.37).

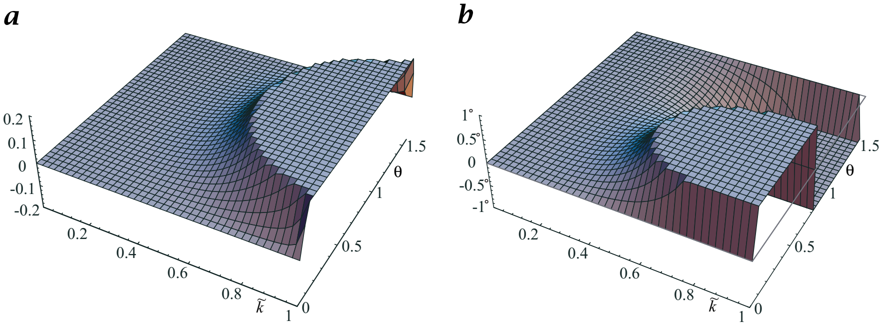

Figure 12.6: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the cubic B-spline derivative operator according to Eq. (12.41). Parameters are the magnitude of the wave number (0 to 1) and the angle to the x axis (0 to $\pi/2$).*

12.5 Edge Detection by Zero Crossings

12.5.1 Principle

Edges constitute *zero crossings* in second-order derivatives (Fig. 12.1). Therefore, the second-order derivatives in all directions can simply be added up to form a linear isotropic edge detector with the transfer function $-(\pi\tilde{k})^2$ (Eq. (12.9)), known as the *Laplacian operator* . From Fig. 12.1 it is also obvious that not every zero crossing constitutes an edge.

Only peaks before and after a zero that are significantly higher than the noise level indicate valid edges. From Fig. 12.1 we can also conclude that edge detection with the Laplace operator is obviously much more sensitive to noise in the signal than edge detection using a gradient-based approach.

12.5.2 Laplace Filter

We can directly derive second-order derivative operators by a twofold application of first-order operators

$$\mathcal{D}_x^2 = {}^{-}\mathcal{D}_x \; {}^{+}\mathcal{D}_x. \tag{12.44}$$

In the spatial domain, this means

$$[1_\bullet \; -1] * [1 \; -1_\bullet] = [1 \; -2 \; 1]. \tag{12.45}$$

The discrete Laplace operator $\mathcal{L} = \mathcal{D}_x^2 + \mathcal{D}_y^2$ for 2-D images thus has the filter mask

$$\boldsymbol{L} = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix} + \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{12.46}$$

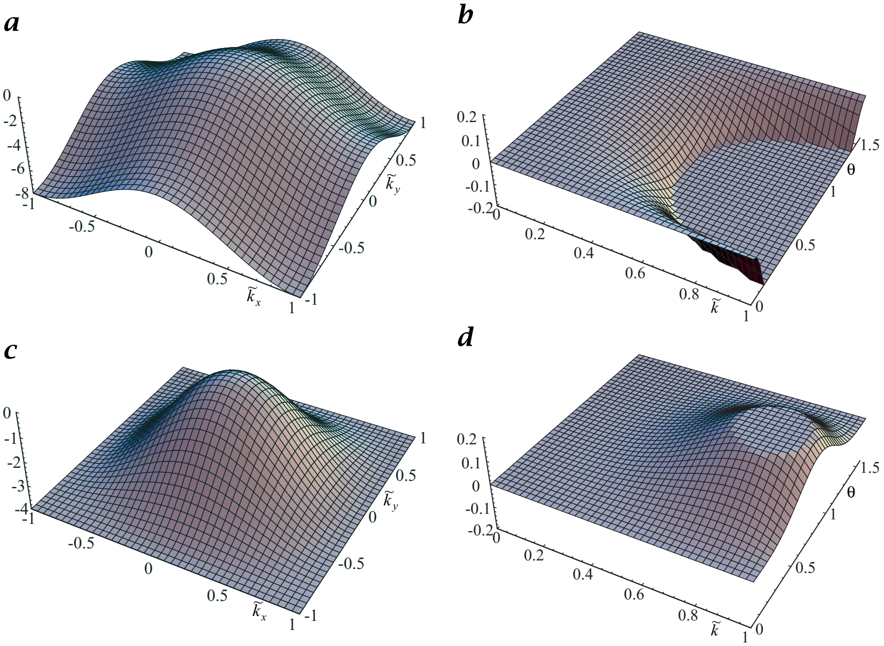

Figure 12.7: *Transfer functions of discrete Laplace operators and their anisotropy:* ***a*** *$\mathcal{L}$ Eq. (12.46),* ***b*** *$\hat{l}(k,\theta) - \hat{l}(k,0)$,* ***c*** *$\mathcal{L}'$ Eq. (12.50),* ***d*** *$\hat{l}'(k,\theta) - \hat{l}'(k,0)$.*

and the transfer function

$$\hat{l}(\tilde{\boldsymbol{k}}) = -4\sin^2(\pi\tilde{k}_x/2) - 4\sin^2(\pi\tilde{k}_y/2). \tag{12.47}$$

Like other discrete approximations of operators, the Laplace operator is only isotropic for small wave numbers (Fig. 12.7a):

$$\hat{l}(\tilde{k},\phi) = -(\pi\tilde{k})^2 + \frac{3}{48}(\pi\tilde{k})^4 + \frac{1}{48}\cos 4\phi(\pi\tilde{k})^4 + O(\tilde{k}^6). \tag{12.48}$$

There are many other ways to construct a discrete approximation for the Laplace operator. An interesting possibility is the use of binomial masks. With Eq. (11.25) we can approximate all binomial masks for sufficiently small wave numbers by

$$\hat{b}^{2R}(\tilde{k}) \approx 1 - \frac{R}{4}(\tilde{k}\pi)^2 + O(\tilde{k}^4). \tag{12.49}$$

From this equation we can conclude that any operator $\mathcal{B}^p - \mathcal{I}$ constitutes a Laplace operator for small wave numbers. For example,

$$\boldsymbol{L}' = 4(\boldsymbol{B}^2 - \boldsymbol{I}) = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{12.50}$$

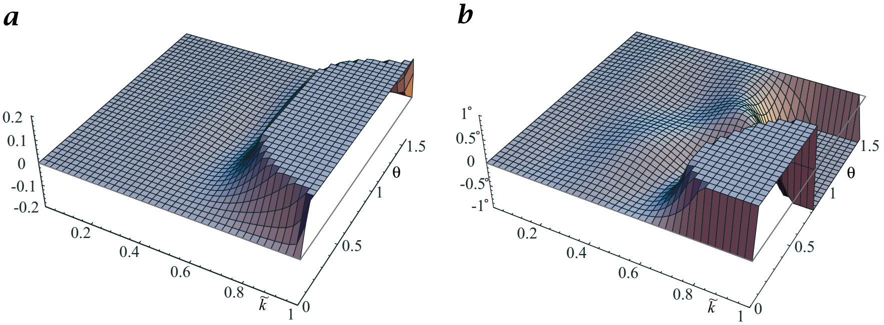

Figure 12.8: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the least squares optimized derivative filter according to Eq. (12.56) for $R = 3$ ($d_1 = -0.597949$,$d_2 = 0.189835$, $d_3 = -0.0357216$). Parameters are the magnitude of the wave number (0 to 1) and the angle to the x axis (0 to $\pi/2$).*

with the transfer function

$$\hat{l}'(\tilde{\boldsymbol{k}}) = 4\cos^2(\pi\tilde{k}_x/2)\cos^2(\pi\tilde{k}_y/2) - 4 \tag{12.51}$$

is another example of a discrete Laplacian operator. For small wave numbers it can be approximated by

$$\hat{l}'(\tilde{k},\phi) \approx -(\pi\tilde{k})^2 + \frac{3}{32}(\pi\tilde{k})^4 - \frac{1}{96}\cos 4\phi(\pi\tilde{k})^4 + O(\tilde{k}^6). \tag{12.52}$$

For large wave numbers, the transfer functions of both Laplace operators show considerable deviations from an ideal Laplacian, $-(\pi\tilde{k})^2$. $\mathcal{L}'$ is significantly less anisotropic than $\mathcal{L}$ (Fig. 12.7).

12.6 *Optimized Edge Detection*

In this section first-order derivative filters are discussed that have been optimized using the least squares technique already used in Section 10.6.2 to optimize interpolation filters. The basic idea is to use a one-dimensional $2R + 1$ filter mask with odd symmetry in the corresponding direction w and to vary the coefficients so that the transfer function approximates the ideal transfer function of a derivative filter, $\mathrm{i}\pi\tilde{k}_w$, with a minimum deviation. Thus the *target function* is

$$\hat{t}(\tilde{k}_w) = \mathrm{i}\pi\tilde{k}_w \tag{12.53}$$

and the transfer function of a one-dimensional $2R + 1$ filter with R unknown coefficients is

$${}^R\hat{d}(\tilde{k}_w) = -\mathrm{i}\sum_{v=1}^{R} 2d_v \sin(v\pi\tilde{k}_w). \tag{12.54}$$

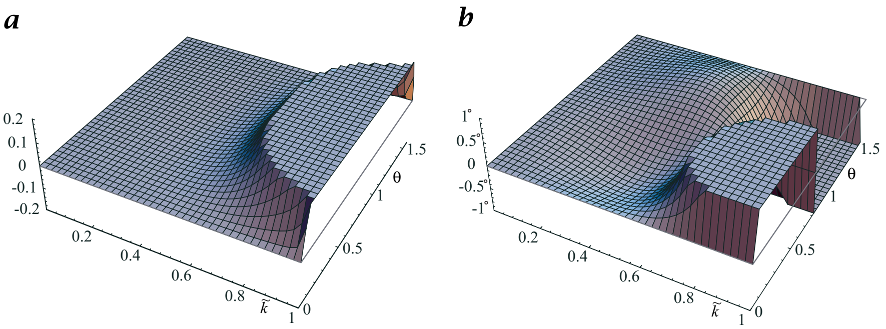

Figure 12.9: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the least squares recursive derivative filter according to Eq. (12.58) for $R = 2$ ($\beta = -0.439496$, $d_1 = -0.440850$, $d_2 = -0.0305482$. Parameters are the magnitude of the wave number (0 to 1) and the angle to the x axis (0 to $\pi/2$).*

As for the interpolation filters in Section 10.6.2, the coefficients are determined in such a way that ${}^{R}\hat{d}(\tilde{\boldsymbol{k}})$ shows a minimum deviation from $\hat{t}(\tilde{\boldsymbol{k}})$ in the least-squares sense:

$$\int_0^1 w(\tilde{k}_w) \left| {}^{R}\hat{d}(\tilde{k}_w) - \hat{t}(\tilde{k}_w) \right|^2 \mathrm{d}\tilde{k}_w. \tag{12.55}$$

The wave number-dependent weighting function $w(\tilde{k}_w)$ determines the weighting of the individual wave numbers.

One useful additional constraint is to force the transfer function to be equal to $\mathrm{i}\pi\tilde{k}$ for small wave numbers. This constraint reduces the degree of freedom by one for a filter with R coefficients, so only $R-1$ can be varied. The resulting equations are

$$ {}^{R}\hat{d} = -\mathrm{i}\sin(\pi\tilde{k}_w) - \mathrm{i}\sum_{v=2}^{R} 2d_v \left(\sin(v\pi\tilde{k}_w) - v\sin(\pi\tilde{k}_w)\right) \tag{12.56}$$

and

$$d_1 = 1 - \sum_{v=2}^{R} v d_v. \tag{12.57}$$

As a comparison of Figs. 12.6 and 12.8 shows, this filter exhibits a significantly lower error than a filter designed with the cubic B-spline interpolation.

Derivative filters can be further improved by compensating the decrease in the transfer function by a forward and backward running recursive relaxation filter (Section 4.5.5, Fig. 4.5b). Then the resulting transfer function is

$$ {}^{(R,\beta)}\hat{d} = \frac{-\mathrm{i}\sin(\pi\tilde{k}) - \mathrm{i}\sum\limits_{v=2}^{R} 2d_v \left(\sin(v\pi\tilde{k}_w) - v\sin(\pi\tilde{k}_w)\right)}{1 + \beta - \beta\cos(\pi\tilde{k}_w)} \tag{12.58}$$

with the additional parameter β. Figure 12.9 shows the errors in the magnitude and direction of the gradient for $R = 2$.

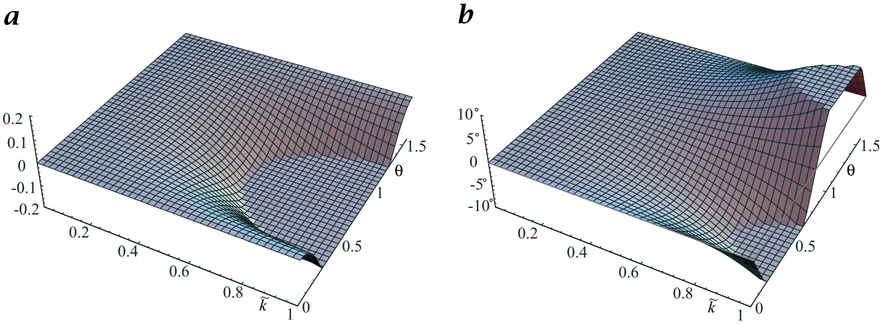

Figure 12.10: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the* 2×2 *cross-smoothing edge detector Eq. (12.59). The parameters are the magnitude of the wave number (0 to 1) and the angle to the* x *axis (0 to* $\pi/2$*).*

A more detailed discussion on the design of optimal derivative filters including tables with filter coefficients can be found in Jähne [89].

12.7 *Regularized Edge Detection*

12.7.1 *Principle*

The edge detectors discussed so far are still poor performers, especially in noisy images. Because of their small mask sizes, they are most sensitive to high wave numbers. At high wave numbers there is often more noise than signal in images. In other words, we have not yet considered the importance of scales for image processing as discussed in Section 5.1.1. Thus, the way to optimum edge detectors lies in the tuning of edge detectors to the scale (wave number range) with the maximum signal-to-noise ratio. Consequently, we must design filters that perform a derivation in one direction but also smooth the signal in all directions.

Smoothing is particularly effective in higher dimensional signals because it does not blur the edge in all directions perpendicular to the direction of the gradient. Derivative filters that incorporate smoothing are also known as *regularized edge detectors* because they result in robust solutions for the ill-posed problem of estimating derivatives from discrete signals.

12.7.2 2 × 2 Cross-Smoothing Operator

The smallest cross-smoothing derivative operator has the following 2×2 masks

$$\boldsymbol{D}_x\boldsymbol{B}_y = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \quad \text{and} \quad \boldsymbol{D}_y\boldsymbol{B}_x = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \tag{12.59}$$

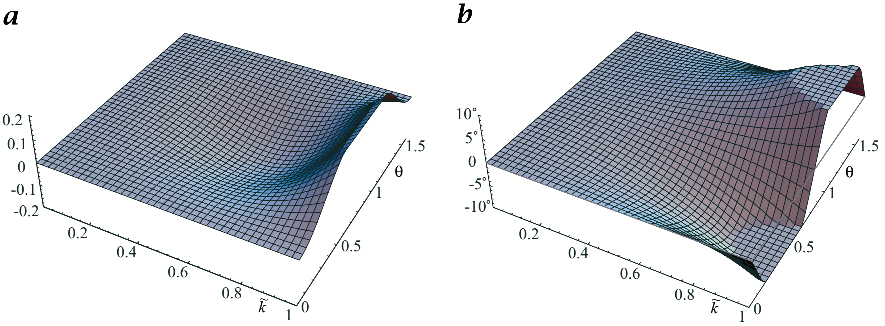

***Figure 12.11:** **a** Anisotropy of the magnitude and **b** error in the direction of the gradient based on the Sobel edge detector Eq. (12.63). Parameters are the magnitude of the wave number (0 to 1) and the angle to the x axis (0 to $\pi/2$).*

and the transfer functions

$$\begin{aligned} \hat{d}_x \hat{b}_y(\tilde{\boldsymbol{k}}) &= 2\mathrm{i}\sin(\pi\tilde{k}_x/2)\cos(\pi\tilde{k}_y/2) \\ \hat{d}_y \hat{b}_x(\tilde{\boldsymbol{k}}) &= 2\mathrm{i}\sin(\pi\tilde{k}_y/2)\cos(\pi\tilde{k}_x/2). \end{aligned} \tag{12.60}$$

There is nothing that can be optimized with this small filter mask. The filters $\boldsymbol{D}_x = [1\ -1]$ and $\boldsymbol{D}_y = [1\ -1]^T$ are not suitable to form a gradient operator, because $\boldsymbol{D}_x$ and $\boldsymbol{D}_y$ shift the convolution result by half a grid constant in the x and y directions, respectively.

The errors in the magnitude and direction of the gradient for small wave numbers are

$$e_m(\tilde{k}, \phi) \approx -\frac{(\pi\tilde{k})^3}{24}\sin^2 2\phi + O(\tilde{k}^5). \tag{12.61}$$

$$e_\phi(\tilde{k}, \phi) \approx -\frac{(\pi\tilde{k})^2}{48}\sin 4\phi + O(\tilde{k}^4). \tag{12.62}$$

The errors are significantly lower (a factor two for small wave numbers) as compared to the gradient computation based on the simple difference operator $\boldsymbol{D}_2 = 1/2\,[1\ 0\ -1]$ (Figs. 12.5 and 12.10), although the anisotropic terms occur in terms of the same order in Eqs. (12.36) and (12.37).

12.7.3 Sobel Edge Detector

The Sobel operator is the smallest difference filter with odd number of coefficients that averages the image in the direction perpendicular to the differentiation:

$$\boldsymbol{D}_{2x}\boldsymbol{B}_y^2 = \frac{1}{8}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad \boldsymbol{D}_{2y}\boldsymbol{B}_x^2 = \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{12.63}$$

The errors in the magnitude and direction of the gradient based on Eq. (12.63) are shown in Fig. 12.11. The improvement over the simple symmetric derivative operator (Fig. 12.5) is similar to the 2 × 2 cross-smoothing difference operator (Fig. 12.10). A Taylor expansion in the wave number yields the same approximations (compare Eqs. (12.61) and (12.62)):

$$e_m(\tilde{k}, \phi) \approx -\frac{(\pi\tilde{k})^3}{24} \sin^2 2\phi + O(\tilde{k}^5) \tag{12.64}$$

for the error of the magnitude and

$$e_\phi(\tilde{k}, \phi) \approx -\frac{(\pi\tilde{k})^2}{48} \sin 4\phi + O(\tilde{k}^4) \tag{12.65}$$

for the direction of the gradient. A comparison with the corresponding equations for the simple difference filter Eqs. (12.36) and (12.37) shows that both the anisotropy and the angle error of the Sobel operator are a factor of two smaller. However, the error still increases with the square of the wave number. The error in the direction of the Sobel gradient is still up to 5° at a wave number of 0.5. For many applications, such a large error cannot be tolerated.

12.7.4 Derivatives of Gaussian

A well-known general class of regularized derivative filters is the class of derivates of a Gaussian smoothing filter. Such a filter was, e. g., used by Canny [21] for optimal edge detection and is also known as the *Canny edge detector*. On a discrete lattice this class of operators is best approximated by a derivative of a binomial operator (Section 11.4) as

$$^{(B,R)}\mathcal{D}_w = \mathcal{D}_{2w}\mathcal{B}^R \tag{12.66}$$

with nonsquare $(2R+3) \times (2R+1)^{W-1}$ W-dimensional masks and the transfer function

$$^{(B,R)}\hat{d}_w(\tilde{\boldsymbol{k}}) = \mathrm{i}\sin(\pi\tilde{k}_w) \prod_{w=1}^{W} \cos^{2R}(\pi\tilde{k}_w/2). \tag{12.67}$$

Surprisingly, this filter turns out to be a bad choice, because its anisotropy is the same as for the simple symmetric difference filter. This can be seen immediately for the direction of the gradient. The smoothing term is the same for both directions and thus cancels out in Eq. (12.27). The remaining terms are the same as for the symmetric difference filter.

In the same way, Sobel-type R^W-sized difference operators

$$^{R}\mathcal{S}_w = \mathcal{D}_w\mathcal{B}_w^{R-1} \prod_{w' \neq w} \mathcal{B}_{w'}^R \tag{12.68}$$

with a $(2R+1)^W$ W-dimensional mask and the transfer function

$$^{R}\hat{S}_d(\tilde{k}) = \mathrm{i}\tan(\pi\tilde{k}_d/2) \prod_{w=1}^{W} \cos^{2R}(\pi\tilde{k}_d/2) \tag{12.69}$$

show the same anisotropy at the same wave number as the 3 × 3 Sobel operator.

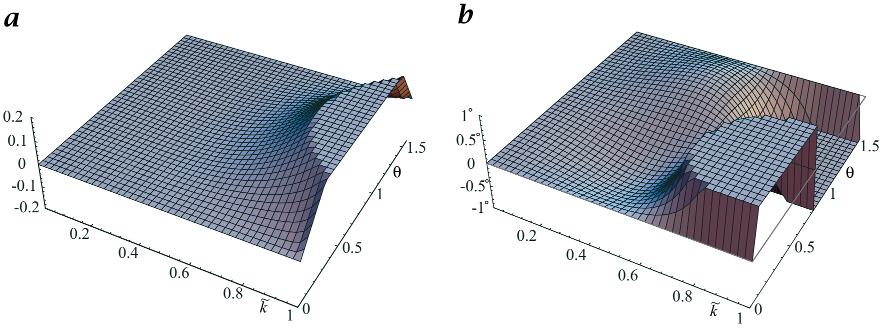

Figure 12.12: ***a*** *Anisotropy of the magnitude and* ***b*** *error in the direction of the gradient based on the optimized Sobel edge detector Eq. (12.70). Parameters are the magnitude of the wave number (0 to 1) and the angle to the x axis (0 to $\pi/2$).*

12.7.5 Optimized Regularized Edge Detectors

It is easy to derive an optimized regularized derivative operator with a significantly lower error in the estimate of edges. A comparison of Eqs. (12.35) and (12.65) shows that the two filters have angle errors in opposite directions. Thus it appears that the Sobel operator performs too many cross-smoothings, while the symmetric difference operator performs too few. Consequently, we may suspect that a combination of both operators may result in a much lower error. Indeed, it is easy to reduce the cross-smoothing by increasing the central coefficient. Jähne et al. [96] show using a nonlinear optimization techniqe that the operators

$$\begin{aligned} 1/4\boldsymbol{D}_{2x}(3\boldsymbol{B}_y^2+\boldsymbol{I}) &= \frac{1}{32}\begin{bmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{bmatrix}, \\ 1/4\boldsymbol{D}_{2y}(3\boldsymbol{B}_x^2+\boldsymbol{I}) &= \frac{1}{32}\begin{bmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix} \end{aligned} \qquad (12.70)$$

have a minimum angle error (Fig. 12.12). Similar optimizations are possible for larger-sized regularized derivative filters.

12.7.6 LoG and DoG Filter

Laplace filters tend to enhance the noise level in images considerably, because the transfer function is proportional to the wave number squared. Thus, a better edge detector may be found by first smoothing the image and then applying the Laplacian filter. This leads to a kind of regularized edge detection and to a class of filters called *Laplace of Gaussian* filters (*LoG* for short) or *Marr-Hildreth operator* [133].

In the discrete case, a LoG filter is approximated by first smoothing the image with a binomial mask and then applying the discrete Laplace filter. Thus we have the operator $\mathcal{L}\mathcal{B}^p$ with the transfer function

$$\hat{L}\hat{B}^p(\tilde{\boldsymbol{k}}) = -4\left[\sin^2(\pi\tilde{k}_x/2) + \sin^2(\pi\tilde{k}_y/2)\right]\cos^p(\pi\tilde{k}_x/2)\cos^p(\pi\tilde{k}_y/2). \tag{12.71}$$

For small wave numbers, this transfer function can be approximated by

$$\hat{L}\hat{B}^p(\tilde{k},\phi) \approx -(\pi\tilde{k})^2 + \left[\frac{1}{16} + \frac{1}{8}p + \frac{1}{48}\cos(4\phi)\right](\pi\tilde{k})^4. \tag{12.72}$$

In Section 12.5.2 we saw that a Laplace filter can be even better approximated by operators of the type $\mathcal{B}^p - \mathcal{I}$. If additional smoothing is applied, this approximation for the Laplacian filter leads to the *difference of Gaussian* type of Laplace filter, or *DoG* filters:

$$4(\mathcal{B}^q - \mathcal{I})\mathcal{B}^p = 4(\mathcal{B}^{p+q} - \mathcal{B}^p). \tag{12.73}$$

The DoG filter $4(\mathcal{B}^{p+2} - \mathcal{B}^p)$ has the transfer function

$$\begin{aligned} 4(\hat{B}^{p+2} - \hat{B}^p)(\boldsymbol{k}) &= 4\cos^{p+2}(\pi\tilde{k}_x/2)\cos^{p+2}(\pi\tilde{k}_y/2) \\ &- 4\cos^p(\pi\tilde{k}_x/2)\cos^p(\pi\tilde{k}_y/2), \end{aligned} \tag{12.74}$$

which can be approximated for small wave numbers by

$$4(\hat{B}^{p+2} - \hat{B}^p)(\tilde{k},\phi) \approx -(\pi\tilde{k})^2 + \left[\frac{3}{32} + \frac{1}{8}p - \frac{1}{96}\cos(4\phi)\right](\pi\tilde{k})^4. \tag{12.75}$$

The transfer function of the LoG and DoG filters are compared in Fig. 12.13. It is obvious that the DoG filter is significantly more isotropic. A filter with even less deviation in the isotropy can be obtained by comparing Eqs. (12.72) and (12.75). The anisotropic $\cos 4\phi$ terms have different signs. Thus they can easily be compensated by a mix of LoG and DoG operators of the form 2/3DoG + 1/3LoG, which corresponds to the operator $(8/3\mathcal{B}^2 - 8/3\mathcal{I} - 1/3\mathcal{L})\mathcal{B}^p$.

DoG and LoG filter operators have some importance for the human visual system [132].

12.8 *Edges in Multichannel Images*

In multichannel images, it is significantly more difficult to analyze edges than to perform averaging, which was discussed in Section 11.7. The main difficulty is that the different channels may contain conflicting information about edges. In channel A, the gradient can point to a different direction than in channel B. The simple addition of the gradients in all channels

$$\sum_{p=1}^{P} \nabla g_p(\boldsymbol{x}) \tag{12.76}$$

is of no use here. It may happen that the gradients in two channels point in opposite directions and, thus, cancel each other. Then, the sum of the gradient over all channels would be zero, although the individual channels would

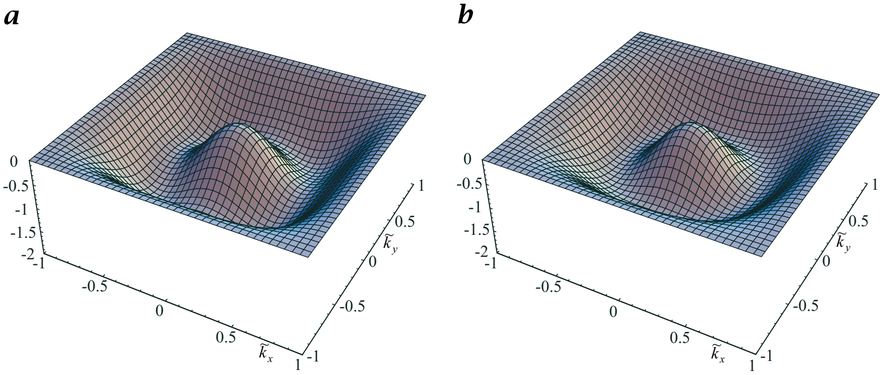

Figure 12.13: *Pseudo 3-D plot of the transfer function of* ***a*** *the LoG filter* $\mathcal{L}\mathcal{B}^2$ *and* ***b*** *the DoG filter* $4(\mathcal{B}^4 - \mathcal{B}^2)$*.*

have non-zero gradients and we would be unable to distinguish this case from constant areas in both channels.

Thus, a more suitable measure of the total edge strength is the sum of the squared magnitudes of gradients in all channels:

$$\sum_{p=1}^{P} |\nabla g_p|^2 = \sum_{p=1}^{P} \sum_{w=1}^{W} \left(\frac{\partial g_p}{\partial x_w} \right)^2 . \tag{12.77}$$

While this expression gives a useful estimate of the overall edge strength, it still does not handle the problem of conflicting edge directions. An analysis of how edges are distributed in a W-dimensional multichannel image with P channels is possible with the following symmetric $W \times W$ matrix $\boldsymbol{S}$ (where W is the dimension of the image):

$$\boldsymbol{S} = \boldsymbol{J}^T \boldsymbol{J}, \tag{12.78}$$

where $\boldsymbol{J}$ is known as the *Jacobian matrix*. This $P \times W$ matrix is defined as

$$\boldsymbol{J} = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \cdots & \frac{\partial g_1}{\partial x_W} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \cdots & \frac{\partial g_2}{\partial x_W} \\ \vdots & & \ddots & \vdots \\ \frac{\partial g_P}{\partial x_1} & \frac{\partial g_P}{\partial x_2} & \cdots & \frac{\partial g_P}{\partial x_W} \end{bmatrix} . \tag{12.79}$$

Thus the elements of the matrix $\boldsymbol{S}$ are

$$S_{kl} = \sum_{p=1}^{P} \frac{\partial g_p}{\partial x_k} \frac{\partial g_p}{\partial x_l} . \tag{12.80}$$

As $\boldsymbol{S}$ is a symmetric matrix, we can diagonalize it by a suitable coordinate transform. Then, we can write

$$\boldsymbol{S}' = \begin{bmatrix} \sum\limits_p \left(\dfrac{\partial g_p}{\partial x_1'}\right)^2 & 0 & \cdots & 0 \\ 0 & \sum\limits_p \left(\dfrac{\partial g_p}{\partial x_2'}\right)^2 & \ddots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \sum\limits_p \left(\dfrac{\partial g_p}{\partial x_W'}\right)^2 \end{bmatrix}. \tag{12.81}$$

In the case of an ideal edge, only one of the diagonal terms of the matrix will be non-zero. This is the direction perpendicular to the discontinuity. In all other directions it will be zero. Thus, $\boldsymbol{S}$ is a matrix of rank one in this case.

In contrast, if the edges in the different channels point randomly in all directions, all diagonal terms will be non-zero and equal. In this way, it is possible in principle to distinguish random changes by noise from coherent edges. The trace of the matrix $\boldsymbol{S}$

$$\operatorname{trace}(\boldsymbol{S}) = \sum_{w=1}^{W} S_{ww} = \sum_{w=1}^{W}\sum_{p=1}^{P} \left(\frac{\partial g_p}{\partial x_w}\right)^2 \tag{12.82}$$

gives a measure of the edge strength which we have already defined in Eq. (12.77). It is independent of the orientation of the edge since the trace of a symmetric matrix is invariant to a rotation of the coordinate system.

12.9 *Exercises*

12.1: Edge and line detection

Interactive demonstration of edge and line detection with several edge detectors based on first-order and second-order derivative filters (dip6ex12.01)

12.2: Edge and line detection on pyramids

Interactive demonstration of edge and line detection with several first-order and second-order derivative filters at different scales on pyramids (dip6ex12.02)

12.3: *First-order difference filters

These are often used first-order difference filters in x direction:

$$a)\ \frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \end{bmatrix},\quad b)\ \frac{1}{6}\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix},\quad c)\ \frac{1}{8}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix},$$

1. Compute the transfer functions of the three filters!

2. Compare and describe the properties of the three filters!
3. Which filter is most suitable for edge detection? Argue for your choice!

12.4: *A bad first-order difference filter

Why is the first-order difference filter

$$[1\ -1], \quad \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

a bad filter to compute the 2-D gradient and for detection of edges?

12.5: **Robert's first-order difference filter

Robert suggested the filter

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

to compute the 2-D gradient and to detect edges.

1. In which directions do these filters detect edges?
2. Compute the transfer function of these filters
3. Compare the quality of this filter with the filter from Exercise 12.4

12.6: **Unknown filters

Here are some unknown filters

$$a)\ \frac{1}{8}\begin{bmatrix} 1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b)\ \frac{1}{8}\begin{bmatrix} 1 & 0 & -2 & 0 & 1 \end{bmatrix},$$

$$c)\ \frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad d)\ \frac{1}{2}\begin{bmatrix} 0 & -1 & 0 \\ -1 & -6 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

to be analyzed.

1. Compute the transfer function of these filters!
2. Are these difference filters of first or second order?
3. How do they compare to the filters described in this chapter?

12.7: **Design of second-order difference filter

Use all necessary properties for a second-order difference filter to show that there can be only one such filter with three coefficients ($[\alpha\ \beta\gamma]$).
If a filter has five coefficients, one free parameter remains. What are the coefficients of this filter and what is its transfer function if you apply the additional constraint that the filter should eliminate structures with the highest wave numbers ($\hat{h}(1) = 0$)?

12.8: *****Isotropy of a 2-D gradient filter**

Isotropy of filters plays a large role in image processing. Smoothing filters should smooth fine structures equally in all directions and derivative filters should detect edges in all directions equally well.

Examine the isotropy of the simple gradient filter

$$\boldsymbol{D}_x = 1/2\,[1\ 0\ -1], \quad \boldsymbol{D}_y = 1/2 \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$$

by expanding the two transfer functions in a Taylor series up to third order in th wave number.

Hint: Isotropy means that the magnitude of the gradient is the same in all directions and that the direction of the gradient is computed correctly. (Section 12.4.2). The computation is easier if you express the wave numbers in polar coordinates: $k_1 = k\cos\varphi, k_2 = k\sin\varphi$.

12.10 *Further Readings*

A vast body of literature about edge detection is available. We will give here only a few selected references. The development of edge detection based on first-order difference filters can nicely be followed by a few key papers. Canny [21] formulated an optimal edge detector based on derivatives of the Gaussian, Deriche [34] introduced a fast recursive implementation of Canny's edge detector, Lanser and Eckstein [116] improved the isotropy of Deriche's recursive filter, and Jähne et al. [96] provide a nonlinear optimization strategy for edge detectors with optimal isotropy. Edge detection based on second-order difference (zero crossings) was strongly influenced by biological vision. The pioneering work is described by Marr and Hildreth [133] and Marr [132]. More recent work towards unified frameworks for neighborhood operators can be found in Koenderink and van Doorn [113] and Danielsson et al. [28].