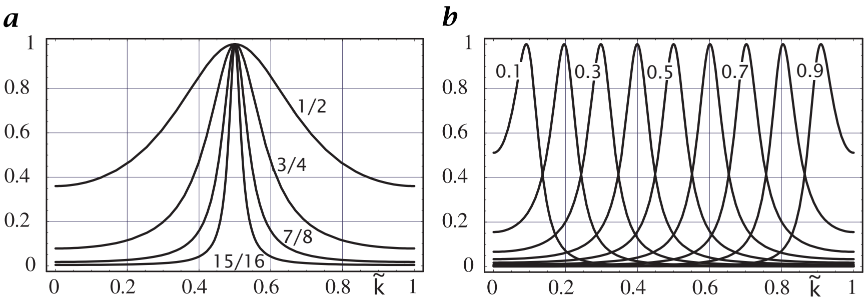

Figure 9.4: *Transfer function of the zero-phase recursive resonance filter for* ***a*** $\tilde{k}_0 = 1/2$ *and values of* r *as indicated; and* ***b*** $r = 7/8$ *and values of* $\tilde{k}_0$ *as indicated.*

The denominator in Eq. (9.61) is still the same as in Eq. (9.60); it has only been expanded in terms with $\cos(n\pi\tilde{k}_0)$. The corresponding recursive filter coefficients are:

$$g'_n = (1 - r^2)\sin(\pi\tilde{k}_0)g_n + 2r\cos(\pi\tilde{k}_0)g'_{n\mp1} - r^2 g'_{n\mp2} \qquad (9.62)$$

Figure 9.4 shows the transfer function of this filter for values of $\tilde{k}_0$ and r as indicated.

For symmetry reasons, the factors become most simple for a resonance wave number of $\tilde{k}_0 = 1/2$. Then the recursive filter is

$$g'_n = (1 - r^2)g_n - r^2 g'_{n\mp2} = g_n - r^2(g_n + g'_{n\mp2}) \qquad (9.63)$$

with the transfer function

$$\hat{s}(\tilde{k}) = \frac{(1 - r^2)^2}{1 + r^4 + 2r^2\cos(2\pi\tilde{k})} \qquad (9.64)$$

The maximum response of this filter at $\tilde{k} = 1/2$ is one and the minimum response at $\tilde{k} = 0$ and $\tilde{k} = 1$ is $((1 - r^2)/(1 + r^2))^2$.

This resonance filter is the discrete analog to a linear system governed by the second-order differential equation $\ddot{y} + 2\tau\dot{y} + \omega_0^2 y = 0$, the damped harmonic oscillator. The circular eigenfrequency ω_0 and the time constant τ of a real-world oscillator are related to the parameters of the discrete oscillator, r and $\tilde{k}_0$ by [4]

$$r = \exp(-\Delta t/\tau) \quad \text{and} \quad \tilde{k}_0 = \omega_0\Delta t/\pi \qquad (9.65)$$

9.4 Classes of nonlinear filters

9.4.1 Limitations of linear filters

In the previous sections, the theory of linear shift-invariant filters was discussed in detail. Although the theory of these filters is well established and they can be applied, they still have some severe limitations. Basically, linear filters cannot distinguish between a useful feature and noise. This property can be best demonstrated with a simple example. We assume a simple signal model with additive noise:

$$g'(\boldsymbol{x}) = g(\boldsymbol{x}) + n(\boldsymbol{x}) \Longleftrightarrow \hat{g}'(\boldsymbol{k}) = \hat{g}(\boldsymbol{k}) + \hat{n}(\boldsymbol{k}) \tag{9.66}$$

The signal to noise ratio (SNR) is defined by $\left|\hat{g}(\boldsymbol{k})\right| / \left|\hat{n}(\boldsymbol{k})\right|$. If we now apply a linear filter with the transfer function $\hat{h}(\boldsymbol{k})$ to this signal, the filtered signal is

$$\hat{h}(\boldsymbol{k})\hat{g}'(\boldsymbol{k}) = \hat{h}(\boldsymbol{k})(\hat{g}(\boldsymbol{k}) + \hat{n}(\boldsymbol{k})) = \hat{h}(\boldsymbol{k})\hat{g}(\boldsymbol{k}) + \hat{h}(\boldsymbol{k})\hat{n}(\boldsymbol{k}) \tag{9.67}$$

It is immediately evident that the noise and the signal are damped by the same factor. Consequently, the SNR does not increase at all by linear filtering, it just stays the same.

From the preceding considerations, it is obvious that more complex approaches are required than linear filtering. Common to all these approaches is that in one or another way the filters are made dependent on the context or are tailored for specific types of signals. Often a control strategy is an important part of such filters that controls which filter or in which way a filter has to be applied at a certain point in the image. Here, we will outline only the general classes for nonlinear filters. Pitas and Venetsanopoulos [7] give a detailed survey on this topic.

9.4.2 Rank-value filters

Rank-value filters are based on a quite different concept than linear-shift invariant operators. These operators consider all pixels in the neighborhood. It is implicitly assumed that each pixel, distorted or noisy, carries still useful and correct information. Thus, convolution operators are not equipped to handle situations where the value at a pixel carries incorrect information. This situation arises, for instance, when an individual sensor element in a CCD array is defective or a transmission error occurred.

To handle such cases, operations are required that apply selection mechanisms and do not use all pixels in the neighborhood to compute the output of the operator. The simplest class of operators of this type are rank-value filters. While the convolution operators may be characterized by “weighting and accumulating,” rank-value filters may be characterized by “comparing and selecting.”

For this we take all the gray values of the pixels that are within the filter mask and sort them by ascending gray value. This sorting is common to all rank-value filters. They only differ by the position in the list from which the gray value is picked out and written back to the center pixel. The filter operation that selects the medium value is called the *median filter*. The median filter is an excellent example for a filter that is adapted to a certain type of signal. It is ideally suited for removing a single pixel that has a completely incorrect gray value because of a transmission or data error. It is less well suited, for example, to reduce white noise.

Other known rank-value filters are the *minimum filter* and the *maximum filter*. As the names indicate, these filters select out of a local neighborhood, either the minimum or the maximum gray value forming the base for gray-scale *morphological filters* (Chapter 14).

As rank-value filters do not perform arithmetic operations but select pixels, we will never run into rounding problems. These filters map a discrete set of gray values onto itself. The theory of rank-value filters has still not been developed to the same extent as convolution filters. As they are nonlinear filters, it is much more difficult to understand their general properties. Rank-value filters are discussed in detail by Pitas and Venetsanopoulos [7].

9.4.3 Pixels with certainty measures

Linear filters as discussed in Section 9.2 treat each pixel equally. Implicitly, it is assumed that the information they are carrying is of equal significance. While this seems to be a reasonable first approximation, it is certain that it cannot be generally true. During image acquisition, the sensor area may contain bad sensor elements that lead to erroneous gray values at certain positions in the image. Furthermore, the sensitivity and noise level may vary from sensor element to sensor element. In addition, transmission errors may occur so that individual pixels may carry wrong information. Thus we may attach in one way or another a certainty measurement to each picture element.

Once a certainty measurement has been attached to a pixel, it is obvious that the normal convolution operators are no longer a good choice. Instead, the certainty has to be considered when performing any kind of operation with it. A pixel with suspicious information should only get a low weighting factor in the convolution sum. This kind of approach leads us to what is known as *normalized convolution* [8, 9].

This approach seems to be very natural for a scientist or engineer who is used to qualifying any measurement with an error. A measurement without a careful error estimate is of no value. The standard deviation of a measured value is required for the further analysis of any quantity that is related to the measurement. In normalized convolution this common principle is applied to image processing.

The power of this approach is related to the fact that we have quite different possibilities to define the certainty measurement. It need not only be related to a direct measurement error of a single pixel. If we are, for example, interested in computing an estimate of the mean gray value in an object, we can take the following approach. We devise a kind of certainty measurement that analyzes neighborhoods and attaches low weighting factors where we may suspect an edge so that these pixels do not contribute much to the mean gray value or feature of the object.

In a similar way, we can, for instance, also check how likely the gray value of a certain pixel is if we suspect some distortion by transmission errors or defective pixels. If the certainty measurement of a certain pixel is below a critical threshold, we replace it by a value interpolated from the surrounding pixels.

9.4.4 Adaptive and steerable filters

Adaptive filters can be regarded as a linear filter operation that is made dependent on the neighborhood. Adaptive filtering can best be explained by a classical application, that is, the suppression of noise without significant blurring of image features.

The basic idea of adaptive filtering is that in certain neighborhoods we could very well apply a smoothing operation. If, for instance, the neighborhood is flat, we can assume that we are within an object with constant features and thus apply an isotropic smoothing operation to this pixel to reduce the noise level. If an edge has been detected in the neighborhood, we could still apply some smoothing, namely, along the edge. In this way, some noise is still removed but the edge is not blurred. With this approach, we need a set of filters for various unidirectional and directional smoothing operations and choose the most appropriate smoothing filter for each pixel according to the local structure around it. Because of the many filters involved, adaptive filtering may be a very computational-intensive approach. This is the case if either the coefficients of the filter to be applied have to be computed for every single pixel or if a large set of filters is used in parallel and after all filters are computed it is decided at every pixel which filtered image is chosen for the output image.

With the discovery of steerable filters [10], however, adaptive filtering techniques have become attractive and computationally much more efficient.

9.4.5 Nonlinear combinations of filters

Normalized convolution and adaptive filtering have one strategy in common. Both use combinations of linear filters and nonlinear point operations such as pointwise multiplication and division of images. The combination of linear filter operations with nonlinear point operations makes the whole operation nonlinear.

The combination of these two kinds of elementary operations is a very powerful instrument for image processing. Operators containing combinations of linear filter operators and point operators are very attractive as they can be composed of very simple and elementary operations that are very well understood and for which analytic expressions are available. Thus, these operations in contrast to many others can be the subject of a detailed mathematical analysis. Many advanced signal and image-processing techniques are of that type. This includes operators to compute local structure in images and various operations for texture analysis.

9.5 Local averaging

Averaging is an elementary neighborhood operation for multidimensional signal processing. Averaging results in better feature estimates by including more data points. It is also an essential tool to regularize otherwise ill-defined quantities such as derivatives (Chapters 10 and 12). Convolution provides the framework for all elementary averaging filters. In this chapter averaging filters are considered for continuous signals and for discrete signals on square, rectangular and hexagonal lattices. The discussion is not restricted to 2-D signals. Whenever it is possible, the equations and filters are given for signals with arbitrary dimension.

The common properties and characteristics of all averaging filters are discussed in Section 9.5.1. On lattices two types of averaging filters are possible [3, Section 5.7.3]. Type I filters generate an output on the same lattice. On a rectangular grid such filters are of odd length in all directions. Type II filters generate an output on a grid with lattice points between the original lattice points (intermediate lattice). On a rectangular grid such filters are of even length in all directions. In this chapter two elementary averaging filters for digital multidimensional signals are discussed—box filters (Section 9.5.3) and binomial filters (Section 9.5.4). Then we will deal with techniques to cascade these elementary filters to large-scale averaging filters in Section 9.5.5, and filters with weighted signals (normalized convolution) in Section 9.5.6.

9.5.1 General properties

Transfer function. Any averaging filter operator must preserve the mean value. This condition means that the transfer function for zero wave number is 1 or, equivalently, that the sum of all coefficients of the mask is 1:

$$\hat{h}(\mathbf{0}) = 1 \iff \int_{-\infty}^{\infty} h(\boldsymbol{x})\,\mathrm{d}^D x = 1 \quad \text{or} \quad \sum_{\boldsymbol{n}\in\text{mask}} H_{\boldsymbol{n}} = 1 \tag{9.68}$$