Design and Implementation of Multi-Steerable Matched Filters

Matthias Mühlich and David Friedrich and Til Aach
Institute of Imaging and Computer Vision
RWTH Aachen University, 52056 Aachen, Germany
tel: +49 241 80 27860, fax: +49 241 80 22200
web: www.lfb.rwth-aachen.de

in: IEEE Transactions on Pattern Analysis and Machine Intelligence. See also BibTeX entry below.

\bibitem{MUE12a}
\author{Matthias M{"u}hlich and David Friedrich and Til Aach}
title{Design and Implementation of Multi-Steerable Matched Filters}
journal{IEEE Transactions on Pattern Analysis and Machine Intelligence}
volume{34}, number{2}, year{2012}, pages{279--291}

© 2011 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.
Design and Implementation of Multisteerable Matched Filters

Matthias Mühlich, David Friedrich, Student Member, IEEE, and Til Aach, Senior Member, IEEE

Abstract—Image analysis problems such as feature tracking, edge detection, image enhancement, or texture analysis require the detection of multi-oriented patterns which can appear at arbitrary orientations. Direct rotated matched filtering for feature detection is computationally expensive, but can be sped up with steerable filters. So far, steerable filter approaches were limited to only one direction. Many important low-level image features are, however, characterized by more than a single orientation. We therefore present here a framework for efficiently detecting specific multi-oriented patterns with arbitrary orientations in grayscale images. The core idea is to construct multisteerable filters by appropriate combinations of single-steerable filters. We exploit that steerable filters are closed under addition and multiplication. This allows to derive a design guide for multisteerable filters by means of multivariate polynomials. Furthermore, we describe an efficient implementation scheme and discuss the use of weighting functions to reduce angular oscillations. Applications in camera calibration, junction analysis of images from plant roots, and the discrimination of L, T, and X-junctions demonstrate the potential of this approach.

Index Terms—Steerable filters, feature detection, junction analysis, orientation estimation, rotated matched filtering, multi-oriented patterns, template equation, trigonometric polynomials, multivariate polynomials, camera calibration.

1 INTRODUCTION

LOW-LEVEL features like lines, edges, corners, and junctions appear in images at arbitrary orientations. They carry important information about the structure of a scene. Their detection is essential for tasks such as feature tracking, contour detection, image enhancement, and camera calibration, and is often based on matched filtering.

For instance, to calibrate a camera, often a picture of a checkerboard is taken. By detecting the checkerboard crossings in the image and relating them to the expected positions, the parameters of the camera can be estimated, and distortions of the optical system can be corrected [1], [2], [3].

A solution to detect such features is to filter the images with a set of rotated filter kernels. Features, however, can appear at arbitrary orientations. For each possible orientation, the image has to be filtered with the corresponding rotated filter kernel, which is computationally expensive. For the special case of edge and line detection, Jacob and Unser showed that an efficient implementation can be found by basing the filter design scheme on steerable filters [4], [5]. The rotated filter is then given by a weighted sum of a small number of predefined base filters [5]. Single orientations can thus be detected efficiently.

However, many relevant image features are characterized by two or more orientations. Examples of such multi-oriented patterns are corners, junctions, or the above mentioned checkerboard patterns (see Fig. 1). To estimate multiple orientations, the structure tensor approach for estimating a single orientation in images [6], [7], [8], [9] has already been extended toward double [10], [11], [12], [13] or multiple [14], [15] orientations. In these eigensystem-based approaches, orientation is defined as a vanishing directional derivative, and the minimum number of directional derivatives needed to make the observed signal vanish corresponds to the number of orientations. Correspondingly, these approaches for orientation estimation cannot distinguish between the five patterns shown in Fig. 1: They detect and estimate the two orientations, but provide no further information on the specific underlying pattern.

Strategies for junction analysis, such as rotated wedge averaging [16], Perona’s steerable filter kernels [17], the orientation-selective quadrature filter approach by Michaelis and Sommer [18], or the steerable wedge filters by Simoncelli and Farid [19], [20], yield a filter response to a kernel that is steered in one direction. The kernels can be designed to respond to terminating and nonterminating lines or edges [18]. Junction analysis then requires finding maxima in the filter responses when the filters are successively steered in “all” directions. Here we develop filters which can be steered simultaneously in multiple orientations and which discriminate multi-oriented patterns based on modeling these patterns.

Single-oriented steerable filters can be constructed by approximating a polar separable template. Subsequently, we combine these single-oriented filters such that their orientations reflect the multiple orientations of the combined filter. Toward this end, we utilize that steerable filters are closed under multiplication and addition. First, these properties enable to derive a deductive design guide: By inserting single-steerable filters as arguments into a
multivariate polynomial and equating it to the target pattern, we can solve for the coefficients of the polynomial. Combining the single-steerable filters as prescribed by the polynomial then yields the desired pattern-specific filter. Second, it is guaranteed that the outcome is always again a polynomial function (Sections 5.5 and 6.3) which tapers off toward zero near the center of the pattern.

In recent work [21], [22], [23], we have shown how polar separable templates can be approximated by steerable filters and how they can be extended to multisteerable filters, thus allowing us to represent corners, junctions, or other multi-oriented features with a linear combination of a small number of base filters. For single-steerable filters, we briefly summarize and extend these results in Section 3. In addition, we discuss the quality of approximation and justify our choice of a truncated Fourier series. We then show exemplarily in Section 4.1 how single-steerable filters can be combined into a filter for detecting checkerboard crossings. In the remainder of Section 4, we develop a general theory for multisteerable filters and employ it to derive filters for the detection of the pattern shown in Fig. 1. Section 5 describes the efficient implementation of multisteerable filters in the Fourier domain, which also allows to control the oscillations introduced by finite Fourier-series approximations. In Section 6, we describe the use of multisteerable filters for feature detection and discrimination, and provide results for the detection of checkerboard crossings and for junction analysis.

2 Matched Filtering and Steerable Filters

Let us assume that we intend to detect a known image template \( g \) in an image \( f \), with both \( f \) and \( g \) mapping from \( \mathbb{R}^2 \) to \( \mathbb{R} \). This problem can be formulated as maximizing the correlation between image patch and template. For white noise, the optimal matched filter is the mirrored version \( g_m \) of the known signal [24], i.e., \( g_m(x) = g(-x) \), for \( x \in \mathbb{R}^2 \). In images, however, one often seeks to detect the features for arbitrary rotations, thus requiring extension of the concept of matched filtering toward rotated matched filtering. To simplify notation, we introduce the rotation operator \((\cdot)^\theta \) that rotates a bivariate function\(^1\) by the angle \( \theta \):

\[
g^\theta(r, \phi) = g(r, \phi - \theta),
\]

(1)

with \( r \geq 0 \) and \( \phi, \theta \in [0, 2\pi) \). This allows the definition of the principle of rotated matched filtering: Let a sought feature be represented by a template \( g(x) \). A measure of how strongly this feature is present in an image \( f(x) \) at a given position \( x_0 \) is \( A_{\max}(x_0) \):

\[
A_{\max}(x_0) = A(\theta|x_0) = \max_\theta ([f(x) * g^\theta(x)]|_{x_0}),
\]

(2)

with \( \cdot \) denoting correlation, \([\cdot]|_{x_0} \) meaning “evaluated at \( x_0 \)”, and assuming, without loss of generality, that the template and the image patch are normalized to unit energy. Each point \( x_0 \) in the image is assigned an estimated orientation angle \( \theta \). At this rotation angle, the cross-correlation of the image patch centered at \( x_0 \) and the rotated template is maximized. The corresponding filter kernel is \( g_m(x) = g(-x) \). The feature is said to be present at those points \( x_0 \) and oriented in direction \( \theta \) where \( A_{\max}(x_0) \) exhibits a sufficiently prominent local maximum.

Steerable filters [5] allow an efficient computation of (2). If any rotated version of \( g \) can be written as a linear combination,

\[
g^\theta(x) = \sum_{j=1}^{M} k_j(\theta)g_j(x),
\]

(3)

the linearity of the correlation operator implies that we only have to compute the correlation with all base functions \( g_j \). The correlation of \( f \) with \( g^\theta \) at a position \( x \) is then given by

\[
f(x) * g^\theta(x) = \sum_{j=1}^{M} k_j(\theta)(f(x) * g_j(x)).
\]

(4)

3 Single Steerable Filters Based on Complex Exponentials

One option for steerable filters—which is also used by Jacob and Unser—is to base them on derivatives of the two-dimensional isotropic Gaussian function (cf. also [25]). This leads to Cartesian-separable filters, though at the expense of exceeding the minimum number of filters needed. Another variant (e.g., [19], [20]) defines additional filter properties like phase-invariant behavior.

Here, we approximate the template by a steerable function which closely resembles the template itself. We assume polar separability for the template—a property which is fulfilled by many relevant image features like wedges or edge functions. A template \( g \) then can be split into

\[
g(r, \phi) = g_{\text{dist}}(r) g_{\text{ang}}(\phi),
\]

(5)

1. With slight abuse of notation, we will always use the same variable for a function, regardless of whether it is represented in Cartesian coordinates \( x \) or polar coordinates \( r, \phi \).
The edge template is, for instance, modeled by the radial and angular functions

\[
g_{\text{dist}}(r) = \begin{cases} 
1, & r \leq r_{\text{max}}, \\
0, & \text{else},
\end{cases}
\]

\[
g_{\text{ang}}(\phi) = \begin{cases} 
1, & 0 \leq \phi < \pi, \\
-1, & -\pi \leq \phi < 0.
\end{cases}
\]  

The 2π-periodicity of the angular function allows us to develop \(g_{\text{ang}}\) into the Fourier series

\[
g_{\text{ang}}(\phi) = \sum_{p=-\infty}^{\infty} a_p \exp(jp\phi).
\]

With a truncated Fourier series of order of \(P\), we approximate the angular function by

\[
\tilde{g}_{\text{ang}}(\phi) := \sum_{p=-P}^{P} a_p \exp(jp\phi),
\]

and the template by

\[
g(r, \phi) \approx \tilde{g}(r, \phi) := g_{\text{dist}}(r) \tilde{g}_{\text{ang}}(\phi).
\]

Fig. 2 shows real and imaginary parts of the first five basis functions \(g_{\text{dist}}(r) \cdot \exp(jp\phi)\) of \(\tilde{g}(r, \phi)\) according to (8) and (9).

Using a truncated Fourier series for approximation is justified by a theorem from Fourier analysis, which states that for a piecewise differentiable function, the Fourier coefficients \(a_p\) are bounded by \(|a_p| < c/|p|\), for \(p \in \mathbb{Z} \setminus \{0\}\) for a suitable constant \(c\) [26]. The Fourier coefficients of the edge template, for instance, obey this bound with \(c = 4\). Thus, most of the information for reconstructing \(g\) from its Fourier series is accumulated in the lower coefficients. Furthermore, the truncated Fourier series is the trigonometric polynomial of order \(P\) which exhibits the lowest L2-distance to \(g\).

Using complex exponentials to describe the angular function enables computing the correlation of \(f\) and a rotated version of \(\tilde{g}\) at a position \(x\) in a steerable manner. With \(f_\Omega(\cdot) := f(\cdot - x)\), we have

\[
f_\Omega(y) * \tilde{g}^\theta(y) = \int_\Omega f_\Omega(y) * \tilde{g}^\theta(y) \, dy
\]

\[
= \int_0^{2\pi} \int_0^{2\pi} f_\Omega(r, \phi) \tilde{g}_\text{dist}(r, \phi - \theta) r \, d\phi dr
\]

\[
= \int_0^{2\pi} \int_0^{2\pi} f_\Omega(r, \phi) g_{\text{dist}}(r) \tilde{g}_{\text{ang}}(\phi - \theta) r \, d\phi dr
\]

\[
= \int_0^{2\pi} \int_0^{2\pi} f_\Omega(r, \phi) g_{\text{dist}}(r) \sum_{p=-P}^{P} a_p e^{j p (\phi - \theta)} r \, d\phi dr
\]

\[
= \sum_{p=-P}^{P} a_p e^{-jp\theta} \int_0^{2\pi} f_\Omega(r, \phi) g_{\text{dist}}(r) e^{j p \theta} r \, d\phi dr
\]

\[
= \sum_{p=-P}^{P} a_p e^{-jp\theta} \int_0^{2\pi} f_\Omega(y) g_p(y) \, dy
\]

\[
= \sum_{p=-P}^{P} a_p e^{-jp\theta} f(y - x) \star g_p(y),
\]

where \(g_p(y) = g_{\text{dist}}(r)e^{j p \theta}\) in Cartesian coordinates, \(\Omega\) is the support of the template, and \(f(y - x) \star g_p(y)\) can be computed by convolving \(f(y)\) with \(g_p(-y)\) and evaluating at \(x\). Consequently, the correlation of the image with any rotated version of \(\tilde{g}\) is given by multiplying the Fourier coefficients \(a_p\) with \(e^{-jp\theta}\) and using these for a linear representation by \(2P + 1\) filtered images. As the rotation angle \(\theta\) only affects the coefficients of the linear combination, we have constructed a steerable filter. Using exponentials for the approximation is often convenient for the construction of a multisteerable filter (see also [5], [27]).

Complex exponentials are also used by the steerable wedge filters of Simoncelli and Farid; however, the weights \(a_p\) for their steerable filters do not originate from an approximation of the template with a Fourier series but by maximally localizing the orientation map impulse response [19], [20].

In the Appendix, which can be found in the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.143, it is shown that the error we induce by approximating \(g\) by the template \(\tilde{g}\) is bounded by the quality of approximating \(g_{\text{ang}}\) by the trigonometric polynomial \(\tilde{g}_{\text{ang}}\) via

\[
|f(x) \ast g(x) - f(x) \ast \tilde{g}(x)| \leq C \|f\|_2 \|g_{\text{ang}} - \tilde{g}_{\text{ang}}\|_2,
\]

with a finite constant \(C\). As mentioned above, the truncated Fourier series is the trigonometric polynomial of order \(P\) which exhibits the smallest L2-distance to \(g_{\text{ang}}\), thus justifying approximating the template in this way.

4 Theory of Multisteerable Filters

The key idea for creating checkerboard patterns (and, later on, corners and other more complex patterns) is the following:

We combine two or more single-steerable templates \(g\) and \(h\) such that they represent a given multi-oriented template \(k\), and such that the result is steerable again with two or more steering angles.

4.1 Example 1: Creating the Checkerboard Pattern from Two Edge Filters

The checkerboard pattern is characterized by two individual orientations. Let \(g_{\text{edge}}\) and \(h_{\text{edge}}\) represent two edge templates. Then, our approach can be expressed as

\[
k_\text{check} = g_{\text{edge}} \circ h_{\text{edge}},
\]

where \(\circ\) is an as yet unknown operator. As illustrated in Fig. 3, the desired steering properties automatically follow if we find a pointwise operator which fulfills the four equations.
A solution is found by identifying white with 1, black with 2 (single) steerable filters: From the general definition of

\[ \hat{g}(r, \phi) = \sum_{p=1}^{\nu} w_p(\alpha) g_p(r, \phi) \]  
\[ \hat{h}(r, \phi) = \sum_{q=1}^{\nu} w_q(\beta) h_q(r, \phi), \]

with steering angle \( \alpha \), weights \( w_p(\alpha) \) and base filters \( g_p(r, \phi) \) for the first filter \( g \) and angle \( \beta \), weights \( w_q(\beta) \) and base filters \( h_q(r, \phi) \) for the second filter \( h \), it follows that

\[ \hat{k}_{\text{mult}}(r, \phi) = \hat{g}(r, \phi) \cdot \hat{h}(r, \phi) = \sum_{p=1}^{\nu} \sum_{q=1}^{\nu} \frac{w_p(\alpha)}{w_p(\alpha) w_q(\beta)} g_p(r, \phi) \cdot h_q(r, \phi) . \]

Note that the numbers of coefficients for the steerable filters do not need to be equal. The resulting filter \( k \) can again be represented as a linear combination of base functions \( k_{p q} \), which can be computed as point-by-point products of the base functions for the standard steerable filter. The number of base functions increases from \( \nu \) to \( \nu^2 \). Similarly, the new weight coefficients \( w_{p q} \) are found as products of the individual weights; therefore, they now depend on two angles, i.e., we have thus generated a double-steerable filter (DSF). Extension to multisteerability is straightforward. Fig. 4 shows approximations for the checkerboard pattern rotated to \( \alpha = 20^\circ \) and \( \beta = 130^\circ \) computed as the product of two edges which were each approximated up to a certain order \( P \). Once the multisteerable filter sought is available, extending rotated matched filtering to multiple rotated matched filtering for feature detection is feasible. In the following, we will discuss a framework for the derivation of templates other than checkerboards and present an implementation which is more efficient than computing products of base filters as done in (14).

### 4.2 Properties of Multisteerable Filters

The previous section demonstrated one key ingredient of multisteerability, namely the representation of multi-oriented patterns as a product of single-oriented patterns which are regarded as steerable filters. This product is a steerable filter again. Indeed, steerable filters are closed under multiplication and addition [28], [29]. Thus, the sum of two steerable filters is also a steerable filter since

\[ h_{\text{add}} = \sum_{p=1}^{\nu} w_p(\alpha) g_p + \sum_{q=1}^{\nu} w_q(\beta) h_q \]

(15)

with

\[ w_p(\alpha, \beta) := \begin{cases} w_p(\alpha) & \text{for } p = 1 \ldots \nu, \\ w_{p-\nu}(\beta) & \text{for } p = \nu + 1 \ldots 2\nu, \end{cases} \]

(16)

\[ k_p := \begin{cases} g_p & \text{for } p = 1 \ldots \nu, \\ h_{p-\nu} & \text{for } p = \nu + 1 \ldots 2\nu. \end{cases} \]

(17)

A steerable filter remains steerable after multiplication with a scalar \( c \) only have to multiply the original weights with \( c \). Finally, a constant mapping \( g(x) = c \) is a steerable filter itself with weight 1, as \( g^\alpha = g \) for any rotation angle \( \alpha \).

Summarizing, we have the following properties:

- **(SF1)** \( g, h \) steerable filters \( \Rightarrow g \cdot h \) is steerable.
- **(SF2)** \( g, h \) steerable filters \( \Rightarrow g + h \) is steerable.
- **(SF3)** \( g \) steerable, \( c \in \mathbb{C} \) \( \Rightarrow c \cdot g \) is steerable.
- **(SF4)** A constant mapping is steerable.

The operations in these four properties are exactly those used within (multivariate) polynomials, and this immediately suggests using steerable filters as arguments for these polynomials. Consider, for example, \( p(x, y) = 1 + 2x + xy + y^2 \) and two steerable filters \( g, h \). Then, for any \( x \) from the image domain \( \Omega, p(g(x), h(x)) \) is a scalar, again yielding a filter. The successive usage of conditions (SF1)-(SF4) guarantees that the result is a steerable filter: We know that the constant mapping \( x \mapsto 1 \) is a steerable filter from (SF4) and that \( 2g \) is steerable from (SF3). (SF1) implies that \( gh \) and \( h^2 \) are both steerable. Ultimately, adding all these filters with (SF2) shows that \( 1 + 2g + gh + h^2 \) is indeed a steerable filter. The inductive principle presented here can be applied for any multivariate polynomial, and this proves that several steerable filters, when combined by means of a multivariate polynomial, yield a steerable filter.

Properties (SF1)-(SF4) above state that steerable filters are closed under addition and multiplication. In [27], [28], [29], the steerability properties of functions under certain families of transforms were analyzed using Lie group theory. In particular, in [28], [29] requirements were
derived under which a function space spanned by basis functions is equivariant with respect to the transform. Furthermore, it is shown that these basis functions span equivariant function spaces when the basis functions are combined to a new basis, or when they are multiplied with each other [28, Definition 2 and Corollaries 1-3]. This, in turn, implies that steerable filters are closed under addition and multiplication.

4.3 Constructing Multisteerable Filters from Polynomials

The polynomial approach paves a deductive way to design templates. Let us examine how we can represent a corner template from the two edges $g_{\text{edge}}$ and $g_{\text{edge}}^{00}$, which can be modeled by the template equation:

$$p \left( \begin{array}{c} \text{edge} \\ \text{edge} \end{array} \right) = \text{edge}$$

with a yet unknown bivariate polynomial $p$ of degree 1,

$$p(x, y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy. \quad (19)$$

For any point $x_1$ in the first quadrant we have $g_{\text{edge}}(x_1) = 1$ and $g_{\text{edge}}^{00}(x_1) = -1$, while the required output is 1. Thus,

$$p(g_{\text{edge}}(x_1), g_{\text{edge}}^{00}(x_1)) = p(1, -1) = \alpha_1 + \alpha_2 \cdot 1 + \alpha_3 \cdot (-1) + \alpha_4 \cdot 1 \cdot (-1) = 1. \quad (20)$$

Likewise, we derive conditions for the remaining quadrants:

$$p(1, 1) = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = -1,$$
$$p(-1, 1) = \alpha_1 - \alpha_2 + \alpha_3 - \alpha_4 = -1,$$
$$p(-1, -1) = \alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 = -1,$$

and obtain a system of four equations with four unknowns

$$\begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \end{bmatrix}. \quad (22)$$

The solution of this system of equations is $\alpha_1 = -0.5$, $\alpha_2 = 0.5$, $\alpha_3 = -0.5$, $\alpha_4 = -0.5$; thus a corner is represented by the steerable filter

$$k_{\text{corner}}(r, \phi) = -0.5 + 0.5 g_{\text{edge}} - 0.5 g_{\text{edge}}^{00} - 0.5 g_{\text{edge}} g_{\text{edge}}^{00}. \quad (23)$$

This demonstrates how our polynomial approach can be applied to solve template equations where (SF1)-(SF4) ensure that this approach yields a steerable filter. Extension to multisteerable filters is again straightforward by using the corresponding number of variables for the polynomial.

The polynomial approach can be used to synthesize junctions from one-sided and two-sided wedges, see Fig. 5. Besides the colors white for representing the numerical value 1 and black for representing -1, these templates have a gray region encoding the numerical value 0. During computation, the points of the image which are correlated with the gray region are nulled out and do not contribute to the correlation. The angular function of these templates is given by

$$g_{\text{ang}}(\phi) = \begin{cases} 1, & |\phi| \leq \frac{\pi}{2}, \\ -1, & \frac{\pi}{2} < |\phi| \leq \omega, \\ 0, & \text{else}, \end{cases} \quad (24)$$

$$g_{\text{ang}}(\phi) = g_{\text{ang}}^{\text{wed-1}}(\phi) + g_{\text{ang}}^{\text{wed-1}}(\phi - 180^\circ; \omega). \quad (25)$$

Multiplication with a radial function, such as the function $g_{\text{dist}}$ from (6), then yields

$$g_{\text{wed-1}}(r, \phi) = g_{\text{dist}}(r) \cdot g_{\text{ang}}^{\text{wed-1}}(\phi)$$
$$g_{\text{wed-2}}(r, \phi) = g_{\text{dist}}(r) \cdot g_{\text{ang}}^{\text{wed-2}}(\phi), \quad (26)$$

as shown in Fig. 5. The L-template, for instance, is constructed from two one-sided wedges by turning the template equation

$$p \left( \begin{array}{c} \text{edge} \\ \text{edge} \end{array} \right) = \text{edge}$$

into requirements upon the polynomial $p$. Whenever one wedge lies within the gray region of the other wedge, this wedge should be preserved: $p(1, 0) = 1$, $p(-1, 0) = -1$, $p(0, 1) = 1$, and $p(0, -1) = -1$. Furthermore, the combination of two gray regions should be a gray region again: $p(0, 0) = 0$. The requirements presented so far are fulfilled by $p(x, y) = x + y$. For the L-template, however, we have to pay special attention to the case where the rotation angles of the two wedge templates are close to each other. Then, the black regions of two templates might overlap, i.e.,

$$p \left( \begin{array}{c} \text{edge} \\ \text{edge} \end{array} \right) = \text{edge}$$

and $p(x, y) = x + y$ would yield $p(-1, -1) = -2$. To prevent this model violation (only the values 1, -1, and 0 are, in this example, allowed in our template equations), we furthermore require $p(-1, -1) = -1$.

Solving for the coefficients of $p$ with a degree 1 yields the solution

$$k^{\text{L-Template}} = g_{\text{wed-1}} + g_{\text{wed-1}} + g_{\text{wed-1}} g_{\text{wed-1}}. \quad (29)$$

The remaining junctions shown in Fig. 1 are given by

$$k^{\text{X-Template}} = g_{\text{wed-2}} + g_{\text{wed-1}} + g_{\text{wed-2}} g_{\text{wed-1}}. \quad (30)$$

$$k^{\text{X-Template}} = g_{\text{wed-2}} + g_{\text{wed-2}} + g_{\text{wed-2}} g_{\text{wed-2}}. \quad (31)$$

We have thus seen that the theory of multisteerable filters allows the representation of various patterns which are relevant for image analysis and how these patterns can be derived. The impulse response of such a filter is directly
related to the corresponding model. Sections 4.4 and 4.5 will discuss the range of the templates and how to choose the degree of the multivariate polynomial \( p \) in (18) before we turn to implementation details in Section 5.

4.4 Range of the Templates

The templates used in this paper are binary or ternary patterns, covering the range \([-1, 1]\) or \([-1, 0, 1]\). The theory derived in the last section rests on solving a finite system of linear equations. Therefore, any real number can be taken as a template value and the approach is also valid for constructing a template whose range covers a finite amount of real numbers. The only prerequisite is that the corresponding system of equations can be solved. By choosing values different from \(-1, 0, 1\), a higher emphasis could be given to parts of the template, for instance one branch of an L template. However, for detecting low-level features, the ternary case often suffices since transitions can be modeled as well as regions be masked out. Finally, approximation of a template with the continuous exponentials generally yields templates which map to a real valued interval. Next, we show that restricting to ternary patterns has the additional advantage of limiting the search domain for the multivariate polynomials to polynomials with degree two.

4.5 Degree of the Multivariate Polynomial

Solving of the template equations was performed using a multivariate polynomial of degree one, where using higher degrees would also be possible. First, however, any multiplication or addition of steerable filters increases the amount of operations or base filters, and this should be kept as low as possible. Second, using a higher degree requires solving for more polynomial coefficients and yields an underdetermined system of equations in the examples shown so far. Consequently, we recommend using higher degrees only if a template equation does not have a solution for polynomials with lower degree. Note, though, that there is no need to consider variables with an exponent higher than two when restricting oneself to ternary templates: We derive the system of equations from a template equation. As the ternary templates only contain the values \(-1, 0, 1\), all power functions with odd exponent equal the mapping \( x \mapsto x \) and all power functions with even exponent equal \( x \mapsto x^2 \). Thus, using exponents higher than two would only add linear dependent columns to our system of equations, but no new information.

To discuss a case where a multivariate polynomial with degree one is not sufficient, we return to the construction of the L-template. We have considered the case when two black regions overlap, as shown in (28). If the difference between the rotation angles of the two wedge templates becomes even smaller, the black region from one template overlaps with the white region of the other wedge, and vice versa. The current solution yields a value of \(-1\); thus the white regions of the L-template become thinner. However, to keep the wedges for detecting lines at constant width, a value of 1 is required. This implies the additional conditions \( p(-1, 1) = 1 \) and \( p(1, 1) = 1 \), and leads to a system of eight equations with four unknowns. This system does not have a solution anymore. Considering a polynomial of degree two,

\[
p(x, y) = a_1 + a_2x + a_3y + a_4xy + a_5x^2 + a_6y^2 + a_7xy^2 + a_8x^2y^2 + a_9x^2y^2,
\]

results in an underdetermined system of equations with eight equations and nine unknowns. From the solution space \( p(x, y) = x + y + (k-1)xy + (k-1)x^2y + (k-1)xy^2 + kx^2y^2 \) with \( k \in \mathbb{R} \), we choose \( k = 0 \), which drops the last summand to reduce the computational load. Still, the multiplication of three steerable filters is required, which leads to more base functions compared to the solution from (29). While demonstrating a case where a polynomial of degree two is needed, in practice the model improvement might not justify the resulting increase in the number of base functions.

5 Efficient Implementation of Multisteerable Filters

In the previous section, we have seen that many relevant image features can be modeled using the class of multisteerable templates. We now discuss how to implement these models efficiently. Replacing the straightforward implementation in the spatial domain by modeling in the Fourier domain will turn out to have several benefits, most notably less computational complexity and better control over oscillations using windowing techniques from FIR filter design.

5.1 Implementation in the Fourier Domain

Double-steerable filters obtained by multiplying two single steerable filters are steerable according to (SF1), i.e., the product can be represented as a sum of weighted base filters. From (14), we observe that, simultaneously, the number of base filters increases quadratically with the number of base functions \( \nu \) of the two factors. For multisteerability beyond two orientations, the increase would quickly become prohibitive. However, this increase in the number of base functions can be reduced to linear complexity. Computing the product of

\[
g^d(r, \phi) = g_{\text{dis}}(r) \sum_{p=-P}^{P} a_p \exp(-jp\alpha) \exp(jp\phi),
\]

and

\[
h^d(r, \phi) = g_{\text{dis}}(r) \sum_{q=-Q}^{Q} b_q \exp(-jq\beta) \exp(jq\phi),
\]

where we assume the same angular approximation order \( P \) for simplicity, yields

\[
k^d(r, \phi) = g^2_{\text{dis}}(r) \sum_{p=-P}^{P} \sum_{q=-Q}^{Q} a_p \alpha b_q \exp(j(p + q)\phi).
\]

With \( q = s - p \) and setting coefficients outside their definition range to zero, we find
We used ideal angular function than convolving two

two dimensional vectors and approach the

approximation problem completely in the (angular) fre-

terior to 2P 1-dimensional vectors (left). We used P = 9 for this example.

\[ k^{\alpha,\beta}(r, \phi) = g^2_{\text{dist}}(r) \sum_{k=-2P}^{2P} \left( \sum_{p=-P}^{P} a^s_p(\alpha) b^s_{-p}(\beta) \right) \exp(j s \phi). \] (36)

Thus, multiplying two approximated single-steerable tem-

plates is equivalent to convolving their Fourier coefficient

Vectors. The latter requires only 4P + 1 coefficients. From

we can furthermore conclude that exponentials from the or-

of the order of 2P to 2P are needed as base functions for

the double-steerable filters, if the single-steerable templates

Use exponentials from the order of

This approach

makes the complexity rise linearly with the approximation

order, instead of the quadratic increase in the spatial domain.

The operations from (SF2)-(SF4) can also be executed

efficiently on the Fourier coefficients. Adding two steerable

filters is performed by adding their corresponding Fourier

coefficients. Scaling a filter equals scaling of the Fourier

coefficient vector and adding a constant to a filter is done by

adding this constant to the 0th Fourier coefficient (DC offset).

5.2 The Number of Base Filters

In the Fourier domain, we need P additional higher order

sine/cosine pairs for the double-steerable template com-

pared to the single-steerable case. A question which arises

then is whether we can find a better approximation of the

multisteerable template if we depart from the derivation as a

product in the spatial domain and approach the approxima-

tion problem completely in the (angular) fre-

quency domain: Since base filters of order 2P to 2P are

needed for the result, the single-steerable “factors” could be

computed with the same number of coefficients (and hence

accuracy) as well.

Let us assume that we approximate the single-steerable

filters to the order of 2P to 2P (instead of P to P as before). Convolving their 4P + 1-dimensional complex Fourier coef-

ficient vectors then yields an 8P + 1-dimensional result, from

which we extract in turn the innermost 4P + 1 coefficients.

Information from higher order Fourier coefficients is thus

also included in the multisteerable filter. Fig. 6 shows that

doing so indeed results in a clear reduction of oscillations and

a steeper transition. Effectively, we replace two coarsely

approximated vectors and their exact multiplication, realized

as convolution in the Fourier domain, by finer vector approximations whose multiplication is only approximated in

the Fourier domain. Doing so increases the overall

approximation quality of the filter product considerably at

very low-additional cost.

5.3 Windowing of Fourier Coefficients

Approximating the angular function by a finite Fourier series of order P can be interpreted as multiplying the

theoretically infinite Fourier series with a rectangular

window function that selects only the complex Fourier coefficients a_p, for −P ≤ p ≤ P. (Unless otherwise noted, in

the following P always denotes the approximation order, i.e., the number of sine/cosine pairs used, for the filter

being discussed, regardless of whether it is single, double,

or multisteerable.) This leads to the Gibbs phenomenon of

undesirable oscillations, which are also observable in our

framework (Fig. 6). From FIR filter design, it is well known

that using window functions with smoother transitions reduces oscillations. We therefore apply such window

functions for the design of double-steerable filters as well.

Potential window functions are the

Hamming window, the

Bartlett window, and the Blackman window [24]. Using these windows, the transition at discontinuities

(black-white) is not as sharp as before, but oscillations are

reduced. Considering that the low-pass characteristics of an

imaging system prevent perfectly sharp transitions anyway,

approximations using weighting functions may actually

perform better in modeling realistic multi-oriented textures.

Fig. 7 shows approximations of the checkerboard pattern

resp. T-junction pattern computed with different window

functions. The reason why we need approximately twice as

many base filters for the T-junction as for the checkerboard

is that the symmetry of the checkerboard pattern lets all

odd-order Fourier coefficients vanish; this is not the case for

T-junctions.

5.4 Normalizing Multisteerable Templates for

Feature Detection

Multisteerable templates are continuous functions of two or

more angles; this means that they can be easily synthesized

for arbitrary angles. This makes them attractive for feature

detection approaches. Instead of correlating an image patch

with a large set of predefined sample feature templates, it is

possible to run an iterative optimization to find the best

match. This is both faster and more accurate because the

entire family of two-parameter templates, e.g., for a

T-junction, can be tested against the image patch and not

just a predefined set of samples.
Before we describe feature detection approaches for specific multi-oriented patterns, we first discuss a potential problem which does not occur in feature detection with single-steerable filters. For single-steerable filters, the template is rotated and the energy of the template remains constant under this rotation. For multisteerable filters, however, we combine two templates with different rotation angles. Interference effects between the single-steerable templates may cause the energy of the resulting double-steerable template to vary over the rotation angles. A varying energy in turn might mislead detection approaches based on maximizing the correlation between image and template by biasing these to angles corresponding to templates with higher energy.

To assess the variation in energy for double-steerable filters, one can, without loss of generality, fix the first angle to zero and express the energy as a function of the difference angle. We observed that especially the interference of oscillations of the finite Fourier series causes noticeable variations in energy. As the windowing introduced in the last section reduces the oscillations, this also mitigates the energy variations (see Fig. 8). Still, in cases where the difference angle between the two templates for constructing the L, T, and X-junctions is very small, even the idealized templates change in energy depending on the difference angle. Consequently, the templates have to be normalized for energy.

Given an image patch \( I_{\text{feature}} \) containing a double-oriented image feature and the corresponding family of double-steerable templates \( I_{\text{template}}(\alpha, \beta) \) with steering angles \( \alpha \) and \( \beta \), we have to maximize the similarity function

\[
Q(\alpha, \beta) = \frac{\langle I_{\text{feature}}, I_{\text{template}} \rangle}{\sqrt{\langle I_{\text{feature}}, I_{\text{feature}} \rangle} \sqrt{\langle I_{\text{template}}, I_{\text{template}} \rangle}}, \tag{37}
\]

where \( \langle \cdot, \cdot \rangle \) stands for the matrix scalar product between the image patches (or standard scalar vector product if all elements of \( N \times N \) templates are stacked to form \( N^2 \) vectors). The values of \( \alpha \) and \( \beta \) maximizing \( Q \) are the sought angles.

### 5.5 Similarity Computation in the Fourier Domain

Since \( I_{\text{feature}} \) is independent of \( \alpha \) and \( \beta \), it can be normalized before optimizing the angles. Still, the cross correlation between template and reference patch and the energy of the template have to be calculated. Since the Fourier coefficients of the templates are at this stage already computed, we can compute the energy of the template directly using Parseval’s theorem [24]:

\[
\|g\|^2 = \int_0^\infty \int_0^{2\pi} |g(r, \phi)|^2 r \, dr \, d\phi \\
= \int_0^\infty |g_{\text{dist}}(r)|^2 r \, dr \int_0^{2\pi} |g_{\text{ang}}(\phi)|^2 d\phi \\
= \frac{1}{2} r_{\text{max}}^2 \sum_{p=-P}^{P} |a_p|^2. \tag{38}
\]

Here, however, we use sampled base functions which are, unlike their continuous counterparts, not necessarily perfectly orthonormal, as required for Parseval’s theorem to hold. Especially in the center, the sampling theorem is violated. Therefore, we introduce a radial weighting function which attenuates the center of the patch. As already discussed in the introduction, and further elaborated in Section 6.3, use of the radial weighting function also improves compliance of junctions in images formed by lines rather than by wedges with the polar separable model.

This means that we can replace (37) by

\[
Q'(\alpha, \beta) = \frac{\langle \vec{v}_{\text{feature}}, \vec{v}_{\text{template}} \rangle}{\sqrt{\langle \vec{v}_{\text{template}}, \vec{v}_{\text{template}} \rangle}}, \tag{39}
\]

where \( \vec{v} \) now stands for \( 4P + 1 \)-element vectors instead of \( N \times N \) image templates, and where we assume that \( \vec{v}_{\text{feature}} \) has already been normalized. Note that \( Q' \) is different from \( Q \) because we project the reference image template to the image subspace which can be represented by the sine/cosine base functions up to order \( P \), thus losing some of its energy. Using Fourier coefficient vectors makes optimizing \( Q' \) directly more efficient than making a detour over the spatial representation of the templates. Summarizing this section, we see that the spatial representation was suited for motivating and deriving the theory, while the actual implementation should be done in the Fourier domain to obtain both lower computation complexity and better approximation quality.

Fig. 9 summarizes the double-steerable filter approach: In the end, we correlate two Fourier coefficient vectors. The given reference vector is computed from the investigated image patch, after applying radial weighting and projection to the space of \( P \) sine/cosine pairs plus DC offset. This vector can be considered as a “fingerprint” of the local structure in the image. (Note that this feature vector is independent of the model to be fitted to it.)

In addition, we need the model synthesizing coefficient vectors of the same type from two (or more) angles, and to
find those angles which result in a coefficient vector best matching the fingerprint generated from the image. Here, the model for the target pattern is specified in the first step (“model selection”). This includes specification of, e.g., the sought junction type (L, T, or X-junction) or of a checkerboard pattern, and the selection of the radial and angular functions (Section 3). Next, the approximation order $P$ is specified, and the Fourier approximations of the single-oriented templates are computed as also described in Section 3. Then, the generating polynomial of the double-steerable filter is determined (Section 4.3). This is followed by computation of the double-steerable filter coefficients (Section 5.1), which includes selecting and applying a window function (Section 5.3).

With an analytical expression mapping any two input angles into a coefficient vector for the desired model and, subsequently, into a measure of correlation with the fixed vector generated from the image patch, the angles can now be determined by iterative optimization (Section 6.1).

### 6 Applications, Experiments, and Results

We tested our algorithm for both checkerboard crossings and various junction type features (L, T, and X-junctions). We first use synthetic data to measure the accuracy of feature detection and angle estimation. For junctions, we then examine the capability of the filters to distinguish between the different junction types in the presence of noise. For both types of features, we furthermore show that the algorithms can also be applied successfully to real image data.

#### 6.1 Feature Detection by Multisteerable Matched Filters

Feature detection basically requires assigning a likelihood for the presence of different features to every pixel in an image. This can be done by considering a small image patch centered at the test pixel and fitting a double-steerable filter to it. To speed up the algorithm, we exploit that, in practice, most locations in images are not likely candidates for multi-oriented features. Recalling that filtering can be rewritten as correlation and that correlation between a 2D image template and a double-steerable filter can be expressed as scalar product between coefficient vectors, we therefore propose finding candidate regions first and then matching an appropriate double-steerable filter to these using scalar products, as discussed in Section 5.5. We first determine a list of candidate points using Harris’ corner measure [30] $M_t = \lambda_1 \lambda_2 - \kappa (\lambda_1 + \lambda_2)^2$, where $\lambda_1$ and $\lambda_2$ denote the eigenvalues of the 2 x 2 structure tensor. The parameter $\kappa$ penalizes regions where the sum but not the product is high, i.e., it penalizes lines or edges. We employed values in the range 0.1 ± 0.05. Points where this measure is above a certain threshold are considered as candidates for double-oriented features. Usually, at most a few percent of all pixels qualify as candidate points, unless images with low resolution and/or many double-oriented features are used.

Next, we analyze the candidate regions further by applying the radial weight function and computing the Fourier coefficient vector (the “fingerprint”). Then, we exploit that (almost) all meaningful low-level image features, whether single-steerable (line, edge) or multi-steerable (corner, crossing, checkerboard), can be approximated well by polar-separable templates. Thus, we compute the ratio between coefficient vector energy and image patch energy. If this ratio is low, then the patch cannot be modeled well as polar separable and is not analyzed further. This again reduces the number of candidate points.

Only for the candidates passing both tests is the double-steerable filter approach applied to find the angles and the model which maximize the correlation between synthesized and measured coefficient vector. The sought angles $\alpha$ and $\beta$ are found using a Levenberg-Marquardt optimization approach. If model selection (e.g., distinction between L, T, and X-junctions) is required, this has to be done for all models; the best fit determines the model deemed to be correct.

The choice of initial values requires some care, though. While optimization of checkerboard crossings usually converges to the correct result even with random initialization, optimization of junctions is more difficult because of the small angular support of the filter and the sharp black-white transitions. The normalized correlation (or similarity) between reference patch and model exhibits several local maxima, see Fig. 10. We have applied two...
If necessary, the best fitting angles at the apex can be found for each local maximum. Its apex is taken as the final feature location. This is sufficient.

For junctions, however, we propose an alternative approach: As the angular support for these templates is small, the double-steerable template for L, T, or X-junctions can be modeled reasonably well as a sum of 2, 3, or 4 one-sided wedge templates, depicted in Fig. 5 left (cf. also [19], [20]). For a given test coefficient vector, one can then sample the wedge response in steps of 10 degrees and compute estimates for the correlation of various DSF models from it. The initial value with the best estimated correlation is then fed into the Levenberg-Marquardt optimizer. (Compared to Section 4.3, the above wedge approach is consistent with modeling the L-template occluding model] can be solved for the initial angles; we used this technique in the checkerboard case.

The final step is obtaining subpixel accuracy. Having found the best fitting double-steerable filter and their correlation values on an (incomplete) pixel grid, we find the local maxima of the correlation and fit a paraboloid to the nine correlation values in a $3 \times 3$-neighborhood around each local maximum. Its apex is taken as the final feature location. If necessary, the best fitting angles at the apex can be interpolated from the angles at the integer grid.

6.2 Detection of Checkerboard Crossings

We first examined the application of double-steerable filters to the detection of checkerboard crossings. Here, we focused on the localization of the crossing center because the localization of these features with subpixel accuracy is of high relevance for the calibration of a grayscale camera.

We tested our algorithm on both synthetic and real data. Experiments on synthetic data with known ground truth enable measuring the root mean square (RMS) error of the localization over varying signal-to-noise ratios (SNRs). This also allows a comparison to the corner finder from Bouguet’s toolbox.

Our experimental setup was as follows: For SNRs from $-5$ to $20$ dB in steps of $2.5$ dB using additive white Gaussian noise, we calculated 10 noisy realizations for each of three different synthetic input images, resulting in 30 realizations for each noise level. For each realization, we estimated the locations of the crossings in an $8 \times 8$ tiles checkerboard, i.e., 49 inner crossings. Subsequently, the RMS error was computed. Then the average RMS error of the 30 estimation results was plotted against the noise level. The same was done for Bouguet’s corner finder. Here, we even gave Bouguet’s corner finder an advantage: It needs an initial value, which we always initialized with the true optimum.

The results of both algorithms are shown in Fig. 11. The pixel error of our approach is roughly one-third of Bouguet’s approach. For low-noise levels, our algorithm achieves a localization accuracy of 0.028 pixels (Bouguet: 0.084 pixels).

Besides finding the location of checkerboard crossings, the steerable filter approach also provides the angles at which the checkerboard lines cross, a quantity which is not provided by Bouguet’s toolbox. The orientations which can be deduced from these angles are especially helpful when it comes to finding the neighboring crossings in the case of strong distortions (see Fig. 12). A neighboring crossing can be expected in a small fan-shaped region around the orientation estimated with the DSF approach. The accuracy of the angle estimates for our test data is approximately 1.25 degrees for low and medium noise levels and is achieved with approximation order $P = 9$ (which means only five different sine/cosine base filter pairs as even orders vanish).

6.3 Junction Analysis, Synthetic Data

The second major group of double-oriented features are junctions: light lines on dark background or vice versa. Rather than on localization accuracy, we focus here more on discrimination between different junction types and on the angle estimation accuracy. In Section 6.1, we already discussed radial weighting in the context of feature detection. For junctions, it turns out that the radial function $g_{\text{dist}}$ from (6), i.e., simple cut-off at some distance $r_{\text{max}}$ is not a good choice. Our double-steerable templates are polar separable. A junction formed from two lines is not. However, if we select the radial function already addressed in Section 5.5 such that the center of the template is faded out, junctions can be modeled very well. These radial weight functions effectively “look” at a ring-like structure around the center pixel of the patch.
Again, we start with synthetic data. We first tested the accuracy of the angle estimation in presence of noise. Similar to the checkerboard detection problem, we varied the SNRs from \(\text{SNR} = 5\) to 20 dB in steps of 2.5 dB using additive white Gaussian noise. We calculated 10 noisy realizations for each of three different synthetic input images (difference angle 40, 80, and 120 degrees), resulting in 30 realizations for each noise level. We implemented this setup for all three junction types (L, T, X).

Fig. 13 (left) shows the angular RMS estimation error for the three junction types over increasing noise levels (approximation order \(P = 20\)). It is always below 1 degree. Fig. 13 (right) then shows the angular error for different approximation orders (with fixed SNR of 10 dB). It can be seen that the system breaks down if the approximation order is reduced below \(P = 12\).

Fig. 14 shows the angular RMS estimation error for the three junction types over increasing noise levels (approximation order \(P = 20\)). It is always below 1 degree. The correlation averaged over 30 realizations for each SNR is plotted over decreasing SNR in Fig. 14. Evidently, on average and over a wide range of SNR, the correct model provides the best fit by a relatively wide margin. Only for very low SNR (\(-5\) dB) did we find that the correct model did not always provide the best fit for each realization: Here, 14 percent of the X-junctions were mistaken for T-junctions because apparently one of the four branches could not be recovered from noise. L and T-junctions at \(-5\) dB and X-junctions at \(-2.5\) dB were classified wrong in less than 3 percent of all experiments. All other runs converged to the correct result. Fig. 15 illustrates the performance of the matching scheme in presence of noise. Note that the projection of the observed image signal to the vector space of base matrices can be considered as radial noise filtering.

6.4 Junction Analysis, Real Data

We also applied the junction analysis tool to three different types of real images: plant roots, brick wall, and a photo of a nine men’s morris board game. We first analyzed a grayscale image of a branching plant root (cf. [31], [32]). The left image in Fig. 16 shows an example for a “difficult” T-junction. In general, this type of junction as well as X or Y-junctions can be found in various biological or medical image data. T or Y-junctions are formed by branching, while X-junctions are formed by two lines (or roots, blood vessels, etc.) crossing each other. Fig. 16 shows that fitting double-steerable template models real T and X-junctions very well.

Another real data experiment was carried out on a brick wall image from Liu’s texture database. Fig. 17 shows that the T-junctions formed by the gaps between the stones were modeled successfully. For comparison, we also included the response of the template to a wedge filter with the angle being steered from 0 to 360 degrees (cf. orientation maps in [20]). While the resulting profile serves as an intermediate result toward feature detection and/or classification, a suitable postprocessing algorithm is still needed to identify
the number and positions of the maxima, and to eliminate those local maxima not representing junction branches.

Fig. 18 finally shows the model selection performance of our approach. In an image of a nine men’s morris board game, we searched for all junctions and let our algorithm classify them as L, T, or X-junction. Fig. 18 shows that all junctions were classified correctly, and that the estimated angles are visually consistent with the image data. As in the checkerboard case, all comparisons between reference patch and model are computed as scalar products between coefficient vectors and all necessary derivatives can be computed in analytic form, which speeds up the iterative optimization process for the sought orientation angles.

7 SUMMARY AND CONCLUSIONS

We developed a theory of multisteerable filters and showed how to deductively construct these filters with multivariate polynomials. These filters allow us to represent low-level image features, but unlike (single) steerable filters, the templates computed with our theory can be steered in two (and potentially more) directions individually, thus allowing us to represent, for instance, junctions, corners, or checkerboard crossings.

Using complex exponentials as base filters and the Fourier-domain implementation of Section 5, it is possible to synthesize multiparameter families of important low-level image feature templates as a linear combination of a comparatively low number of base filters.

The rotated matched filtering approach presented in this work is well suited for detecting features with arbitrary orientations in the presence of white noise. For future research, it will be interesting to examine the performance of our approach in the presence of colored noise or interlacing, where noise depends on the direction of a feature. Other efforts could be directed at theoretical as well as experimental comparisons to the single-steerable approaches to junction analysis discussed in the introduction, viz., [18], [16], [17], [19], [20].


Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, AA5/3-1). Parts of this work were presented at IEEE ICIP 2007, San Antonio, EUSIPCO 2007, Poznan, and Mustererkennung 2007, Heidelberg.

References


Matthias Mühlich received the diploma and PhD degrees from Johann Wolfgang Goethe University Frankfurt, Germany, in 1999 and 2005, respectively. From 2006 to 2009, he was a research scientist with the Institute of Imaging and Computer Vision, RWTH Aachen University, and since 2009, he has been with CanControls, Aachen, Germany. His research interests are the estimation of orientations, motions, and subspaces.

David Friedrich received the Diploma in mathematics from RWTH Aachen University in June 2009. During his studies he focused on signal processing and he completed the diploma thesis in cooperation with Philips Research Europe. Results of this thesis led to the filing of three patents. Currently, he is a research scientist working toward the PhD degree with the Institute of Imaging and Computer Vision, RWTH Aachen University. His research interests are in biomedical image and Fourier analysis, and a student member of the IEEE.

Til Aach received the Diploma and Doctoral degrees in electrical engineering from RWTH Aachen University in 1987 and 1993, respectively. From 1993 to 1998, he was with Philips Research Labs, Aachen, Germany. In 1996, he was also an independent lecturer with the University of Magdeburg, Germany. In 1998, he was appointed a full professor and director of the Institute for Signal Processing, University of Luebeck. Since 2004, he has been a full professor with RWTH Aachen University. His research interests are in medical and industrial image processing, signal processing, pattern recognition, and computer vision. From 2002 to 2008, he was an associate editor of the IEEE Transactions on Image Processing. He is a member of the IEEE Bio-Imaging and Signal Processing Committee (BISP-TC). He is a senior member of the IEEE.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.