

# Recognition by Symmetry Derivatives and the Generalized Structure Tensor

Josef Bigun, *Fellow, IEEE*, Tomas Bigun, and Kenneth Nilsson, *Student Member, IEEE*

**Abstract**—We suggest a set of complex differential operators that can be used to produce and filter dense orientation (tensor) fields for feature extraction, matching, and pattern recognition. We present results on the invariance properties of these operators, that we call symmetry derivatives. These show that, in contrast to ordinary derivatives, all orders of symmetry derivatives of Gaussians yield a remarkable invariance: They are obtained by replacing the original differential polynomial with the same polynomial, but using ordinary coordinates  $x$  and  $y$  corresponding to partial derivatives. Moreover, the symmetry derivatives of Gaussians are closed under the convolution operator and they are invariant to the Fourier transform. The equivalent of the structure tensor, representing and extracting orientations of curve patterns, had previously been shown to hold in harmonic coordinates in a nearly identical manner. As a result, positions, orientations, and certainties of intricate patterns, e.g., spirals, crosses, parabolic shapes, can be modeled by use of symmetry derivatives of Gaussians with greater analytical precision as well as computational efficiency. Since Gaussians and their derivatives are utilized extensively in image processing, the revealed properties have practical consequences for local orientation based feature extraction. The usefulness of these results is demonstrated by two applications: 1) tracking cross markers in long image sequences from vehicle crash tests and 2) alignment of noisy fingerprints.

**Index Terms**—Gaussians, orientation fields, structure tensor, differential invariants, cross detection, fingerprints, tensor voting, tracking, filtering, feature measurement, wavelets and fractals, moments, invariants, vision and scene understanding, representations, shape, tracking, registration, alignment.

## 1 INTRODUCTION

GAUSSIAN filters, and derivatives of Gaussian filters, are frequently used to produce dense orientation maps of images. Example applications of such maps include tracking corners in image sequences [17], [29], and extracting minutiae points in fingerprint image processing, [21]. Here, we present symmetry derivatives of Gaussians along with analytical and experimental results that are useful for pattern recognition. The *structure tensor*, [6], [23], that has been in use in many contexts [19], [24], [28], [35] to compute and/or represent orientation fields, can be analytically extended to yield the *generalized structure tensor* [3], which allows to represent and to detect more intricate patterns than straight lines and edges, e.g., those in Fig. 1. Since this tensor, and the applications which use it are significant benefactors of our results, we summarize it in Section 2 along with the prior background. We present our main results on symmetry derivatives, that are nonspecific to the theory of the structure tensor, as theorems and lemmas in Section 3. The proofs of the novel theorems and lemmas are given in the Appendix whereas known results, to the extent they are indispensable to our illustrations of these theorems, are briefly stated with references to proofs. The main idea of Section 4 is to illustrate the impact of the results in Section 3 on the structure tensor theory summarized in Section 2. The novel analytical results

important to the practice of the structure tensor theory, is presented as a lemma in Section 4. The pattern orientation parameter along with a useful error measure, the most crucial parameters in practice, have been shown to be implementable via 1D correlations, only. However, to obtain the minimum and maximum errors explicitly, a 2D filtering is still required for patterns with odd symmetry orders which will be defined further below. The applications we used to illustrate the theoretical findings consists of 1) cross marker tracking in vehicle crash tests and 2) fingerprint alignment. These are presented in Section 5. It is demonstrated that both applications are realized by filtering (structure) tensor fields and that a contribution of our results has been robust and computationally effective detection schemes delivering features having a precise meaning. We present our conclusions in Section 6.

Albeit in parametric statistics domain, the report of [31] provides a valuable insight into estimation of angular variables. Another relevant contribution is the motion estimation technique suggested by [29] that primarily concerns characterization of regions lacking orientation so that the aperture problem can be avoided in regions having a distinct orientation. However, the earliest efforts on invariant pattern matching, including rotation and translation invariance, are represented by the reports of [8], [16], [20], although their formulations concern binary images or contours.

The popularity of Gaussians as filters, [32], is due to their valuable properties including: 1) directional isotropy, i.e., in polar coordinates they depend on radius only, 2) separability in  $x$  and  $y$  coordinates, and 3) simultaneous concentration in the spatial and the frequency domain. These motivations increasingly make the Gaussians the prime choice in Finite Impulse Response (FIR) filter implementation of the linear operators, e.g., [11], [15], [41],

- J. Bigun and K. Nilsson are with Halmstad University, Box 823, SE-30118, Halmstad, Sweden.  
E-mail: {josef.bigun, kenneth.nilsson}@ide.hh.se.
- T. Bigun is with TietoEnator AB, Storg. 3, 58223 Linköping, Sweden.  
E-mail: tomas.bigun@tietoenerator.com.

Manuscript received 17 Dec. 2002; revised 10 Feb. 2004; accepted 13 May 2004.

Recommended for acceptance by W.T. Freeman.

For information on obtaining reprints of this article, please send e-mail to: [tpami@computer.org](mailto:tpami@computer.org), and reference IEEECS Log Number 117903.

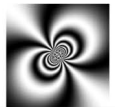



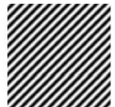
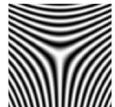
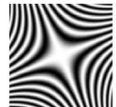
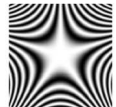
| $z^{-1}$  | $z^{-0.5}$  | $\log(z)$   | $z^{0.5}$   | $z^1$   | $z^{1.5}$  | $z^2$   | $z^{2.5}$   |
|---|---|---|---|---|--|---|---|
|  |  |  |  |  |  |  |  |
| $\Gamma\{-4, \sigma^2\}$  | $\Gamma\{-3, \sigma^2\}$  | $\Gamma\{-2, \sigma^2\}$  | $\Gamma\{-1, \sigma^2\}$  | $\Gamma\{0, \sigma^2\}$   | $\Gamma\{1, \sigma^2\}$  | $\Gamma\{2, \sigma^2\}$   | $\Gamma\{3, \sigma^2\}$   |
| $n = -4$  | $n = -3$  | $n = -2$  | $n = -1$  | $n = 0$   | $n = 1$  | $n = 2$   | $n = 3$   |

Fig. 1. The top row shows the harmonic functions, (25), that generate the patterns in the second row. The isocurves of the images are given by a linear combination of the real and the imaginary parts of the harmonic functions on the top according to (26) with a constant parameter ratio, i.e.,  $\varphi = \tan^{-1}(a, b) = \frac{\pi}{4}$ . The third row shows the symmetry derivative filters that are tuned to detect these curves for any  $\varphi$  while the last row shows the symmetry order of the filters.

as well as nonlinear operators which rely on linear operators e.g., edge operators [32], scale analysis [9], [26], [28], orientation analysis [23], [35], and singularity points detection schemes [14], [17].

## 2 THE STRUCTURE TENSOR AND ITS GENERALIZATION

Here, our goal is to present the generalized structure tensor that will be used in other sections. However, to do this, we need the ordinary structure tensor which we state below in two variants. To fix the ideas, we first present the most known version of it, so that we can switch to the less known variant in which we can identify the same measurement parameters. This allows to present the generalized structure tensor, which is based on the second variant but uses curvilinear coordinates, without effort.

We will refer to an image neighborhood in a 2D image as *image*, to the effect that we will treat the local images in the same way as the global image. Let the scalar function  $f$ , taking a two-dimensional vector  $r = (x, y)^T$  as argument, represent an image. Consider the matrix

$$\mathbf{S}(f) = \begin{pmatrix} \iint (D_x f(x, y))^2 & \iint (D_x f(x, y))(D_y f(x, y)) \\ \iint (D_x f(x, y))(D_y f(x, y)) & \iint (D_y f(x, y))^2 \end{pmatrix}, \quad (1)$$

where integrations of the elements are carried over the entire real axis for the variables  $x$  and  $y$  assuming that  $f$  already contains a possible window function. Introduced by [6], [23] in pattern recognition, this matrix is a tensor. Sometimes with the notion “matrix” replacing “tensor,” it has been called symmetry tensor, inertia tensor, structure tensor, moment tensor, orientation tensor, etc., among others, [6], [24], [23], [35]. We retain here the term structure tensor because it appeared most common to us.

The image  $f$  is called linearly symmetric if its isocurves have a common direction, i.e., there exists a scalar function of one variable  $g$  such that  $f(x, y) = g(k^T r)$ , where  $k$  is a two-dimensional real vector that is constant with regard to  $r = (x, y)^T$ . The term is justified in that the spectral energy of  $g(k^T r)$  is concentrated to a line in addition to that the linear symmetry direction  $k$  represents the common direction, or the mirror symmetry direction of the isocurves of  $g(k^T r)$ .

Whether or not an image is linearly symmetric can be determined by eigen analysis of the structure tensor via the

following result, [6]. We assume that the capitalized  $F$  is the Fourier transform of  $f$  and we denote with  $|F|^2$  the power spectrum of  $f$ .

**Theorem 1 (Structure tensor I).** *The extremal inertia axes of the power spectrum,  $|F|^2$ , are determined by the eigenvectors of the structure tensor*

$$\mathbf{S} = \begin{pmatrix} \iint (\omega_x)^2 |F(\omega_x, \omega_y)|^2 & \iint \omega_x \omega_y |F(\omega_x, \omega_y)|^2 \\ \iint \omega_x \omega_y |F(\omega_x, \omega_y)|^2 & \iint (\omega_y)^2 |F(\omega_x, \omega_y)|^2 \end{pmatrix} \quad (2)$$

$$= \begin{pmatrix} \iint (D_x f)^2 & \iint (D_x f)(D_y f) \\ \iint (D_x f)(D_y f) & \iint (D_y f)^2 \end{pmatrix}.$$

The eigenvalues  $\lambda_{min}$ ,  $\lambda_{max}$ , and the corresponding eigenvectors  $k_{min}$ ,  $k_{max}$  of the tensor represent the minimum inertia, the maximum inertia, the axis of the maximum inertia, and the axis of the minimum inertia of the power spectrum, respectively.

We note that  $k_{min}$  is the least eigen vector but it represents the axis of the maximum inertia. This is because the inertia tensor  $\mathbf{R}$  in mechanics equals to  $\mathbf{R} = \text{Trace}(\mathbf{S})\mathbf{I} - \mathbf{S}$  with  $\mathbf{I}$  being the unit matrix. If  $\mathbf{S}k = \lambda k$ , then  $\mathbf{R}k = (\text{Trace}(\mathbf{S}) - \lambda)k$ , so that the structure and the inertia tensors share eigenvectors, for any dimension. But, since in 2D  $\text{Trace}(\mathbf{S}) = \lambda_{max} + \lambda_{min}$ , the two tensors additionally share eigenvalues in 2D, although the correspondence between the eigenvalues and the eigenvectors is reversed. Because of this tight relationship between the two tensors, the structure tensor  $\mathbf{S}$  and the inertia tensor  $\mathbf{R}$  can replace each other in applications of computer vision. While its major eigenvector fits the minimum inertia axis to the power spectrum, the image itself does not need to be Fourier transformed according to the Theorem. The eigenvalue  $\lambda_{max}$  represents the largest inertia or error, which is achieved with the inertia axis having the direction  $k_{min}$ . The worst error is useful too, because it indicates the scale of the error when judging the size of the smallest error,  $\lambda_{min}$ . By contrast, the axis of the maximum inertia provides no additional information, because it is always orthogonal to the minimum inertia axis as a consequence of the matrix  $\mathbf{S}$  being symmetric, and positive semidefinite. Via Taylor expansion, a spatial interpretation as an alternative to the spectral inertia interpretation can be obtained. In this alternative view, the same structure tensor, via its minor eigenvector, encodes the direction in which a small translation of the image departs it from the original the least.

Before coping with generalization of the structure tensor, we need to restate Theorem 1 in terms of complex moments which will be instrumental. The complex scalar  $I_{mn}$

$$I_{mn}(\kappa) = \iint (x + iy)^m (x - iy)^n \kappa(x, y) dx dy, \quad (3)$$

with  $m$  and  $n$  being nonnegative integers, is the complex moment  $m, n$  of the function  $\kappa$ . The order number and the symmetry number of a complex moment refer to  $m + n$  and  $m - n$ , respectively. We will be particularly interested in the second-order complex moments because of the structure tensor. However, also higher-order complex moments and, thereby, higher-order symmetry derivatives that will be defined below, are valuable image analysis tools, e.g., in texture discrimination and segmentation, [5].

**Theorem 2 (Structure tensor II).** *The minimum and maximum inertia as well as the minimum inertia axis of the power spectrum,  $|F|^2$ , are given by its second-order complex moments*

$$\begin{aligned} I_{20}(|F|^2) &= (\lambda_{max} - \lambda_{min})e^{i2\varphi_{min}} = \iint (\omega_x + i\omega_y)^2 |f|^2 d\omega_x d\omega_y \\ &= \iint ((D_x + iD_y)f)^2 dx dy \end{aligned} \quad (4)$$

$$\begin{aligned} I_{11}(|F|^2) &= \lambda_{max} + \lambda_{min} = \iint (\omega_x + i\omega_y)(\omega_x - i\omega_y) |F|^2 d\omega_x d\omega_y \\ &= \iint |(D_x + iD_y)f|^2 dx dy, \end{aligned} \quad (5)$$

which are computable in the spatial domain without Fourier transformation. The quantities  $\lambda_{min}$ ,  $\lambda_{max}$ , and  $\varphi_{min}$  are, respectively, the minimum inertia, the maximum inertia, and the axis of the minimum inertia of the power spectrum.

The eigenvalues of the tensor in Theorem 1 and the  $\lambda$ s appearing in this theorem are identical. Likewise, the direction of the major eigenvector of Theorem  $k_{max}$  and the  $\varphi_{min}$ , of Theorem 2 coincide. While the proof of this version of the structure tensor theorem is due to [6], a more recent study has also provided a proof and a different motivation for its existence, [2]. In fact, the complex scalar  $I_{20}$  and the real scalar  $I_{11}$  are linear combinations of the elements of the real and symmetric tensor,  $\mathbf{S}(f)$ . Thus, Theorem 1 and Theorem 2 are mathematically fully equivalent, to the effect that the tuple  $(I_{20}, I_{11})$  is just another way of representing the structure tensor. More importantly, however, from Theorem 2, the following attractive conclusions, that do not easily follow from Theorem 1, emerge:

1. a simple averaging of the "square" of the gradient  $(D_x f + iD_y f)^2$  automatically fits an optimal axis to the spectrum in that the resulting complex number directly encodes the optimal direction and the error difference,
2. a simple averaging of  $|D_x f + iD_y f|^2$  yields the error sum, and
3. the Schwartz inequality  $|I_{20}| = \lambda_{max} - \lambda_{min} \leq I_{11} = \lambda_{max} + \lambda_{min}$ , holds with equality if and only if the image  $f$  has perfect linear symmetry.

Recently, decomposition of the structure tensor, combining differences and sums of the eigenvalues has found novel uses. Called tensor voting, the tensor averaging has been demonstrated as being effective in 3D interpolation problems, [33].

Is it possible, with a fixed number of correlations, to extend the structure tensor idea to find the direction of

sophisticated curve structures and yet provide good precision for location and orientation? The answer to this question is yes. The next theorem, [3], generalizes the structure tensor idea, yielding a method of obtaining the global orientation of other curves than lines, e.g., the orientation of a cross pattern or a fingerprint core point.

**Theorem 3 (Generalized structure tensor).** *The Structure Tensor Theorem holds in harmonic coordinates.<sup>1</sup> In particular, the second-order complex moments determining the minimum inertia axis of the power spectrum,  $|F(\omega_\xi, \omega_\eta)|^2$ , can be obtained in the (Cartesian) spatial domain as:*

$$\begin{aligned} I_{20} &= (\lambda_{max} - \lambda_{min})e^{i2\varphi_{min}} = \iint (\omega_\xi + i\omega_\eta)^2 |F|^2 d\omega_\xi d\omega_\eta \quad (6) \\ &= \iint ((D_\xi + iD_\eta)f)^2 d\xi d\eta, \end{aligned}$$

$$= \iint e^{i \arg((D_x - iD_y)\xi)^2} [(D_x + iD_y)f]^2 dx dy, \quad (7)$$

$$I_{11} = \lambda_{max} + \lambda_{min} = \iint (\omega_\xi + i\omega_\eta)(\omega_\xi - i\omega_\eta) |F|^2 d\omega_\xi d\omega_\eta \quad (8)$$

$$= \iint |(D_\xi + iD_\eta)f|^2 d\xi d\eta, \quad (9)$$

$$= \iint |(D_x + iD_y)f|^2 dx dy.$$

The quantities  $\lambda_{min}$ ,  $\varphi_{min}$ , and  $\lambda_{max}$  are, respectively, the minimum inertia, the direction of the minimum inertia axis, and the maximum inertia of the power spectrum of the harmonic coordinates,  $|F(\omega_\xi, \omega_\eta)|^2$ .

It should be emphasized that the complex moments  $I_{20}$  and  $I_{11}$  are taken with regard to *harmonic coordinates*, e.g., as in (6), although their actual computations only involve the Cartesian grid, e.g., as in (7).

The theorem provides a principle that can be used to extract the position and the orientation of a target pattern with a few filters. Just like in the ordinary structure tensor approach in which only horizontal and vertical filters are used to detect the positions and the orientations of target patterns that possess linear symmetry, the generalized structure tensor determines the position and orientation of its target patterns via two orthogonal filters. The drawback is that a straightforward derivation of these filters yields nonseparable filters, [3]. However, by means of the results introduced in the next section, we will present an alternative technique that yields 1D implementations. Evidently, when the harmonic coordinate transformation is the identity transformation, i.e.,  $\xi = x$  and  $\eta = y$ , Theorem 3 reduces to Theorem 2.

We emphasize that it is the coordinate transformation that determines what  $I_{20}$  and  $I_{11}$  represent and detect. Central to generalized structure tensor is the harmonic function pair  $x = \xi(x, y)$  and  $y = \eta(x, y)$  which creates new coordinate curves to represent the points of the 2D plane. An image  $f(x, y)$  can always be expressed by such a coordinate pair  $\xi = \xi(x, y)$  and  $\eta = \eta(x, y)$  as long as the transformation from  $(x, y)$  to  $(\xi, \eta)$  is one to one and onto. The deformation by itself does not create new gray tones, i.e., no new function values of  $f$  are created, but rather it is the isogray curves of  $f$  that are

1. A coordinate pair  $\xi(x, y), \eta(x, y)$  is harmonic iff  $D_x \xi = D_y \eta$  and  $D_y \xi = -D_x \eta$ , i.e., the curves  $\xi(x, y) = \xi = 0$  and  $\eta(x, y) = \eta_0$  are perpendicular. If  $\xi$  satisfies  $(D_x^2 + D_y^2)\xi = 0$ , then a function  $\eta$  making  $(\xi, \eta)$  a harmonic pair, exists.

deformed. The harmonic coordinate transformations deform the appearance of the target patterns to make the detection process mathematically more tractable. In the principle suggested by Theorem 3, however, these transformations are not applied to an image because they are implicitly encoded in the utilized complex filters. The deformations occur only in the idea, when designing the detection scheme and deriving the filters.

The shape concept that the generalized structure tensor utilizes depends on differential operators which do not require a binarization or an extraction of the contours. Sharing similar integral curves with the generalized structure tensor, the Lie operators of [13], [18] should be mentioned, although these studies have not provided tools on how to estimate the parameters of the integral curves, e.g., the orientation and the estimation error.

The generalized structure tensor is a powerful analytical tool that can model and estimate the position and orientation parameters of harmonic function patterns such as those illustrated by Fig. 1, explicitly. However, it is in place to point out that it was Knutsson et al. who first predicted that convolving complex images by complex filters, can result in detection of intricate patterns, though without providing an analytic model of the computed parameters, [25]. That is because they modeled the local orientation field in isolation from the underlying isocurves. By contrast, the generalized structure tensor models the isocurves by two harmonic basis curves  $\xi$  and  $\eta$ . The linear combinations of these curves define a target pattern family from the beginning, and a member of this family that is closest to the image in the least square error sense is also represented by the tensor. Recently, an efficient polynomial filtering of orientation maps has been worked out by [22] by the use of Gaussian derivatives too. The main novelties in our contribution will be explicated in greater detail in the following sections.

### 3 SYMMETRY DERIVATIVES OF GAUSSIANS

**Definitions:** We define the *first symmetry derivative* as the complex partial derivative operator

$$D_x + iD_y = \frac{\partial}{\partial x} + i \frac{\partial}{\partial y}, \quad (10)$$

which resembles the ordinary gradient in 2D. When it is applied to a scalar function  $f(x, y)$ , the result is a complex field instead of a vector field. Consequently, the first important difference is that it is possible to take the (positive integer or zero) powers of the symmetry derivative, e.g.,

$$(D_x + iD_y)^2 = (D_x^2 - D_y^2) + i(2D_x D_y) \quad (11)$$

$$(D_x + iD_y)^3 = (D_x^3 - 3D_x D_y^2) + i(3D_x^2 D_y - D_y^3) \quad (12)$$

...

Second, being a complex scalar, it is even possible to exponentiate the result of the symmetry derivative, i.e.,  $(D_x + iD_y)^n f$ , to yield nonlinear functionals:  $[(D_x + iD_y)^n f]^m$ .

The operator,  $(D_x + iD_y)^n$  will be defined as the *n*th *symmetry derivative* since its invariant patterns (those that vanish under the operator) are highly symmetric. In an analogous manner, we define, for completeness, the *first conjugate symmetry derivative* as  $D_x - iD_y = \frac{\partial}{\partial x} - i \frac{\partial}{\partial y}$  and the

*n*th *conjugate symmetry derivative* as  $(D_x - iD_y)^n$ . We will, however, only dwell on the properties of the symmetry derivatives. The extension of the results to conjugate symmetry derivatives are straightforward.

We apply the *p*th symmetry derivative to the Gaussian and define the function  $\Gamma^{\{p, \sigma^2\}}$  as

$$\Gamma^{\{p, \sigma^2\}}(x, y) = (D_x + iD_y)^p \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (13)$$

with  $\Gamma^{\{0, \sigma^2\}}$  being the ordinary Gaussian.

**Theorem 4.** *The differential operator  $D_x + iD_y$  and the scalar  $\frac{-1}{\sigma^2}(x + iy)$  operate on a Gaussian in an identical manner:*

$$(D_x + iD_y)^p \Gamma^{\{0, \sigma^2\}} = \left(\frac{-1}{\sigma^2}\right)^p (x + iy)^p \Gamma^{\{0, \sigma^2\}}. \quad (14)$$

The theorem reveals an invariance property of the Gaussians with regard to symmetry derivatives. We compare the second-order symmetry derivative with the classical Laplacian, also a second-order derivative operator, to illustrate the analytical consequences of the theorem. The Laplacian of a Gaussian

$$(D_x^2 + D_y^2) \Gamma^{\{0, \sigma^2\}} = \left(-\frac{2}{\sigma^2} + \frac{x^2 + y^2}{\sigma^4}\right) \Gamma^{\{0, \sigma^2\}} \quad (15)$$

can obviously not be obtained by a mnemonic replacement of the derivative symbol  $D_x$  with  $x$  and  $D_y$  with  $y$  in the Laplacian operator. As the Laplacian already hints, with an increased order of derivatives, the resulting polynomial factor, e.g., the one on the righthand side of (15) will resemble less and less the polynomial form of the derivation operator. Yet, it is such a form invariance that the theorem predicts when symmetry derivatives are utilized. By using the linearity of the derivation operator, the theorem can be generalized to any polynomial as follows:

**Lemma 1.** *Let the polynomial  $Q$  be defined as  $Q(q) = \sum_{n=0}^{N-1} a_n q^n$ . Then,*

$$Q(D_x + iD_y) \Gamma^{\{0, \sigma^2\}}(x, y) = Q\left(\frac{-1}{\sigma^2}(x + iy)\right) \Gamma^{\{0, \sigma^2\}}(x, y). \quad (16)$$

That the Fourier transformation of a Gaussian is also a Gaussian has been known and exploited in information sciences. It turns out that a similar invariance is valid for symmetry derivatives of Gaussians too. In the theorem below, as well as in the rest of the paper, all integrals have their integration domains as the entire 2D plane.

A proof of the following theorem is omitted because it follows by observing that derivation with regard to  $x$  corresponds to multiplication with  $i\omega_x$  in the Fourier domain and applying (14). Alternatively, Theorem 3.4 of [39] can be used to establish it.

**Theorem 5.** *The symmetry derivatives of Gaussians are Fourier transformed on themselves, i.e.,*

$$\mathcal{F}[\Gamma^{\{p, \sigma^2\}}] = \quad (17)$$

$$= \iint \Gamma^{\{p, \sigma^2\}}(x, y) e^{-i\omega_x x - i\omega_y y} dx dy \quad (18)$$

$$= 2\pi\sigma^2 \left(\frac{-i}{\sigma^2}\right)^p \Gamma^{\{p, \frac{1}{\sigma^2}\}}(\omega_x, \omega_y). \quad (19)$$

We note that, in the context of prolate spheroidal functions [38], and when constructing rotation invariant 2D filters [10], it has been observed that the (integer) symmetry order  $n$  of the function  $h(\rho) \exp(in\theta)$  where  $h$  is a one-dimensional function,  $\rho$  and  $\theta$  are polar coordinates, is preserved under the Fourier transform. To be precise, the Fourier transform of such functions are:  $H_0[h(\rho)](\omega_\rho)$ , where  $H_0$  is the Hankel transform (of order 0) of  $h$ . However, this result does not provide sufficient guidance as to how the function family  $h$  should be chosen in order to make it invariant to Fourier transform.

Another analytic property that can be used to construct efficient filters by cascading smaller filters or simply to gain further insight into steerable filters and rotation invariant filters is the addition rule under the convolution. This is stated in the following theorem.

**Theorem 6.** *The symmetry derivatives of Gaussians are closed under the convolution operator so that the order and the variance parameters add under convolution.*

$$\Gamma^{\{p_1, \sigma_1^2\}} * \Gamma^{\{p_2, \sigma_2^2\}} = \Gamma^{\{p_1+p_2, \sigma_1^2+\sigma_2^2\}}. \quad (20)$$

#### 4 MATCHING WITH THE GENERALIZED STRUCTURE TENSOR AND THE SYMMETRY DERIVATIVES

In computer vision, normally, one does not locate an edge or compute the orientation of a line by correlation with multiple templates consisting of rotated edges, incremented with a small angle. However, multiple correlations with the target pattern rotated in increments is commonly used to detect other shapes. This approach is also used to estimate the orientation of such shapes. The number of rotations of the template can be fixed a priori or, as in [7], dynamically. The precision of the estimated direction is determined by the number of the rotated templates used or by the amount of computations allowed. Although such techniques yield a generally good precision when estimating the affine parameters, that include target translation and rotation, in certain applications (e.g., our example applications), an a priori undetermined number of iterations may not be possible or be desirable due to imposed restrictions that include hardware and software resources. Furthermore, the precision and/or convergence properties remain satisfactory as long as the reference pattern and the test pattern do not violate the image constancy hypothesis severely. In other words, if the image gray tones representing the same pattern differ nonlinearly and significantly between the reference and the test image, then a good precision or a convergence may not be achieved. Our fingerprint alignment application represents a matching problem that severely violates the image constancy assumption.

An early exception to the “rotate and correlate” approach is the pattern recognition school initiated by Hu [20] who suggested the moment invariant signatures to be computed on the original image which was assumed to be real valued. Later, Reddi [36] suggested the magnitudes of complex moments to efficiently implement the moment invariants of the spatial image, mechanizing the derivation of them. The complex moments contain the rotation angle information directly encoded in their arguments as has been shown in [5]. An advantage they offer is a simple separation of the orientation parameter from the model evidence, i.e., by

taking the magnitudes of the complex moments, one obtains the moment invariants which represent the evidence. The linear rotation invariant filters suggested by [11], [15], [41] resemble the linear filters implementing the complex moments of a real image. With appropriate radial weighting functions, the rotation invariant filters can be viewed as equivalent to Reddi’s complex moments filters which in turn are equivalent to Hu’s geometric invariants. From this view point, the suggestions of [1], [37] are also related to the computation of complex moments of a real image and, hence, deliver correlates of Hu’s geometric invariants. Additionally, however, the latter authors suggest the use of normalized phases which are computed by dividing a complex moment with complex moments having lower orders, typically first-order. In the approach of [37], there is a further advantage in that the phase normalization includes more lower order complex moments increasing resilience to noise. Despite their demonstrated advantage in the context of real images, it is not a trivial matter to directly model *tensor fields* by complex moment filters or their equivalent rotation invariant filters. This is because the argument response when the complex or tensor valued image is convolved with steerable filters is not easy to interpret. By contrast, next we will use symmetry derivatives and the generalized structure tensor to model and to sample tensor fields, yielding a geometric interpretation of the argument response or equivalently the “eigenvectors” of the response tensor field.

An analytic function  $g(z)$  generates a harmonic pair via  $\xi = \Re[g]$  and  $\eta = \Im[g]$ , representing the real and the imaginary parts of  $g$ . Such pairs include the real and imaginary parts of polynomials as well as other elementary functions of complex variables, e.g.,  $\log(z)$ ,  $\sqrt{z}$ ,  $z^{1/3}$ . The next lemma, a proof of which is given in the Appendix, makes use of the symmetry derivatives to represent and to sample the generalized structure tensor. Sampled functions are denoted as  $f_k$ , i.e.,  $f_k = f(x_k, y_k)$ .

**Lemma 2.** *Consider the analytic function  $g(z)$  with  $\frac{dg}{dz} = z^{\frac{n}{2}}$  and let  $n_k$  be integer,  $0, \pm 1, \pm 2, \dots$ . Then, the discretized filter  $\Gamma_k^{\{n, \sigma_2^2\}}$  is a detector for patterns generated by the curves  $a\Re[g(z)] + b\Im[g(z)] = \text{constant}$  provided that a shifted Gaussian is assumed as interpolator and the magnitude of a symmetry derivative of a Gaussian acts as a window function. The discrete scheme*

$$I_{20}(|F(\omega_\xi, \omega_\eta)|^2) = C_n \Gamma_k^{\{n, \sigma_2^2\}} * (\Gamma_k^{\{1, \sigma_1^2\}} * f_k)^2 \quad (21)$$

$$I_{11}(|F(\omega_\xi, \omega_\eta)|^2) = C_n |\Gamma_k^{\{n, \sigma_2^2\}}| * |\Gamma_k^{\{1, \sigma_1^2\}} * f_k|^2, \quad (22)$$

where  $0 \leq n$  and  $C_n$  is a real constant, estimates the orientation parameter  $\tan^{-1}(a, b)$  as well as the error via  $I_{20}$  and  $I_{11}$  according to Theorem 3. For  $n < 0$ , the following scheme yields the analogous estimates

$$I_{20}(|F(\omega_\xi, \omega_\eta)|^2) = C_n \Gamma_k^{*\{n, \sigma_2^2\}} * (\Gamma_k^{\{1, \sigma_1^2\}} * f_k)^2 \quad (23)$$

$$I_{11}(|F(\omega_\xi, \omega_\eta)|^2) = C_n |\Gamma_k^{*\{n, \sigma_2^2\}}| * |\Gamma_k^{\{1, \sigma_1^2\}} * f_k|^2, \quad (24)$$

where  $\Gamma^{*\{n, \sigma_2^2\}} = (\Gamma^{\{n, \sigma_2^2\}})^*$ .

We note that the parameter  $C_n$  is constant with regard to  $(x_l, y_l)$  and has no implications to applications because it can be assumed to have been incorporated to the image the

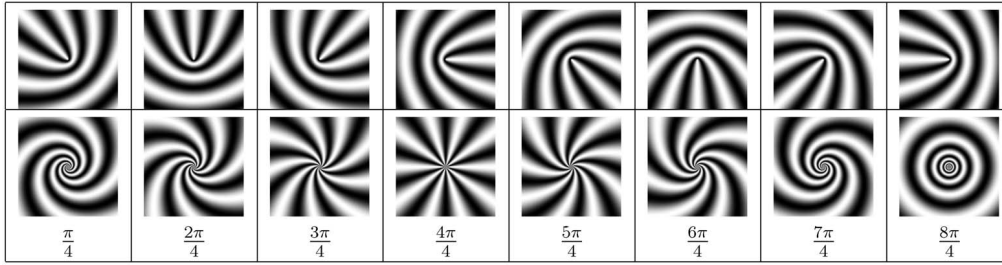


Fig. 2. The top row shows the rotated patterns generated by  $g(z) = \sqrt{z}$ , i.e., when  $n = -1$  in (25), for various angles between the parameters. The second row displays the corresponding curves for  $g(z) = \log(z)$ , i.e., when  $n = -2$ . The third row is  $\varphi$  which represents the parameter ratios used to generate the patterns. The change of  $\varphi$  represents a geometric rotation on the top row whereas it represents a change of bending in the second row.

filter is applied to. In turn, this amounts to a uniform scaling of the gray value gamut of the original image.

#### 4.1 Detectable Patterns and Their Illustration

The procedure below is due to [3]. It uses real and imaginary parts of analytic functions, which are harmonic, to reveal the detectable patterns that Lemma 2 affords via the generalized structure tensor. To that end, we integrate  $\frac{dg}{dz} = z^{\frac{n}{2}}$  to obtain the real and imaginary parts of  $g$ ,

$$g(z) = \begin{cases} \frac{1}{\frac{n}{2}+1} z^{\frac{n}{2}+1}, & \text{if } n \neq -2; \\ \log(z), & \text{if } n = -2. \end{cases} \quad (25)$$

The filter  $\Gamma^{\{n, \sigma^2\}}$  detects the patterns that are generated by real and imaginary parts of  $g(z)$ . Such patterns are shown in Fig. 1 by gray modulation

$$s(a\xi + b\eta) = \cos(a\Re[g(z)] + b\Im[g(z)]). \quad (26)$$

The 1D function  $s(t) = \cos(t)$  is chosen for illustration purposes. The filters that are tuned to detect the isocurves  $a\xi + b\eta$  are not sensitive to  $s$ , but to the angle

$$\varphi = \tan^{-1}(a, b). \quad (27)$$

The nonlinear convolution scheme of Lemma 2 estimates  $\varphi$  via the argument of  $I_{20}$  regardless of  $s$ . In Fig. 1, this angle is fixed to  $\varphi = \frac{\pi}{4}$  and  $n$  is varied between  $-4$  and  $3$ . Each  $n$  represents a separate isocurve family. By changing  $\varphi$  and keeping  $n$  fixed, the parameter pair  $(a, b)$  is rotated to  $(a', b')$ . Except for the patterns with  $n = -2$ , which we will come back to here next, this results in rotating the isocurves since for  $n \neq -2$  and  $g(z) = z^{\frac{n}{2}+1}$ , we have

$$a'\xi + b'\eta = \Re[(a' - ib')(\xi + i\eta)] = \Re[(a' - ib')g(z)] \quad (28)$$

$$= \Re[(a - ib)e^{i\varphi} z^{\frac{n}{2}+1}] = \Re[(a - ib)g(ze^{i\frac{\varphi}{2}})] \quad (29)$$

$$= a\xi' + b\eta'. \quad (30)$$

Here  $\xi'$  and  $\eta'$  are rotated versions of the harmonic pair  $\xi$  and  $\eta$ , so that  $g(z \exp^{i\varphi_0}) = \xi' + i\eta'$  for some  $\varphi_0$ . The top row of Fig. 2 displays the curves generated by (26), for increasing values of  $\varphi$  in  $(a, b) = (\cos(\varphi), \sin(\varphi))$ , to illustrate that a coefficient rotation results in a pattern rotation.

When  $n = -2$ , we obtain the isocurves via the function  $g(x + iy) = \log(|x + iy|) + i \arg(x + iy)$ , which is special in that it represents the only case when a change of the ratio between  $a$  and  $b$  does not result in a rotation of the image pattern. Instead, changing the angle  $\varphi$  bends the isocurves,  $\sin(\varphi) \log(|x + iy|) + \sin(\varphi) \arg(x + iy) = \text{constant}$ . That is,

the spirals become “tighter” or “looser” until the limit patterns, circles and radial patterns, corresponding to infinitely tight and infinitely loose spirals, are reached.

#### 4.2 Implementation and Use

There are two parameters employed by the suggested scheme that control filter sizes,  $\sigma_1$  which is the same as in the ordinary structure tensor, determining how much of the high frequencies are assumed to be noise, and  $\sigma_2$  representing the size of the neighborhood.

Equations (21) and (23) can be implemented via separable convolutions since the filters  $\Gamma_k^{\{n, \sigma_2^2\}}$  are separable for all  $n$ . The same goes for (22) and (24), provided that  $n$  is even. Consequently, for even  $n$ , both  $I_{20}$  and  $I_{11}$  can be computed with 1D filters. For odd  $n$ , only  $I_{11}$  is not separable. For such patterns, while  $I_{20}$  can be computed by the use of 1D filters, the computation of  $I_{11}$  will need one true 2D convolution or an inexact approximation of it obtainable, e.g., by the SVD decomposition of the 2D filter. Alternatively, the computational costs can still be kept small by working with small  $\sigma_2$  and Gaussian pyramids. A fingerprints application demanding detection of patterns with odd symmetry for which computation of  $I_{11}$  has been utilized in a pyramid scheme, is presented in the next section.

Lemma 2 assumes that there is a window function, the purpose of which is to limit the estimation of  $I_{20}$  and  $I_{11}$  to a neighborhood around the current image point. Apart for  $n = 0$ , straight line patterns, the local gradient direction  $\arg(\frac{dg}{dz}) = \arg(z^{\frac{n}{2}})$  is not well defined in the origin for patterns generated by (25). This is visible in Fig. 1. The factor  $|x + iy|^n$  in the window function is consequently justified since it suppresses the origin as the information provider for  $n \neq 0$ . Fig. 1 shows that the filters that are suggested by the Lemma for various patterns vanish at the origin except for the (Gaussian) one used for straight lines extraction.

As mentioned, for  $n \neq 0$ , the origins of the target patterns are singular. Because of this, with increased  $|n|$ , the continuous image of such a pattern becomes increasingly difficult to discretize in the vicinity of the origin, to the effect that the discrete image will have an appearance less faithful to the underlying continuous image near origin. As a consequence, approximating such patterns with band-limited or other regular functions, a necessity for accurate approximation of the integrals representing  $I_{20}$  and  $I_{11}$ , will be problematic because the singularity at the origin is barely or not at all accounted for already at the original image level. The continuous window function, by being close to

zero for  $n \neq 0$ , counter balances this problem, but this will not help if the image is not sufficiently densely discretized when approximating the square of a function. As  $n$  grows, this becomes a necessity for accurate approximation of the integrands of  $I_{20}$  and  $I_{11}$  containing second-order terms. This can be achieved by signal theoretically correct sampling, [12], e.g., when the square of an image on a discrete grid is needed, then the discrete image must be assured to have been over sampled by a factor 2 before pixelwise squaring is applied. Oversampling can be effectively implemented by separable filters and pyramids [9]. In our experiments however, the original sampling frequency of images we used were sufficient to detect the target patterns.

### 4.3 Steerability of the Filters and the Detected Patterns

The filter family  $h(\rho)e^{in\theta}$  is well studied in the context of real valued images, [38]. The term  $e^{in\theta}$  in this angularly separable function family has been observed to be determinant for obtaining rotation invariant linear filters by [11], and later by [41] when discussing the prolate spheroidal filtering in image analysis. The filtering scheme suggested by [26], [27] presents certain derivatives of Gaussian filters that are applied to the original image with the purpose to model the structure by rotation invariant filters. The steerable filter theory, [15], [34], introduced the steerability condition for linear spaces of the above mentioned function family by establishing that the angularly band-limited functions are steerable. In these contributions, the steerable filters have been studied for filtering real valued (gray) images. The filters suggested by Lemma 2 are rotation invariant by construction in the sense of [11], [41], and since they are linear combinations of derivatives of Gaussians, they are also closely related to those of [26], [27]. Because they also contain a single angular frequency,  $e^{in\theta}$ , they are steerable too. Below we discuss the novelties as compared to this background.

First, we study the filtering of tensor fields by means of the filter family,  $h(\rho)e^{in\theta}$ , previously studied in the context of linear filtering of real images, with the exception of the studies in [3], [22], [25]. We model the structure of an image by means of its isocurves instead of modeling its gray tones directly. Consequently, these results help to extend the application field of the mentioned filter family from originally being intended and designed for filtering of real valued images in linear schemes, to vector or tensor images obtained through nonlinear schemes via the same filter family. This is important because through tensor fields one can more directly study isocurves than by first modeling the gray tones and then studying the isocurves within the gray tones.

Second, (28), (29), and (30) show that the isocurves of the patterns that are detectable by the symmetry derivatives of Gaussians are obtained as a linear combination of the nonrotated isocurves, except  $g(z) = \log(z)$ . Yet, half of these patterns are not predicted by the steerability condition, which is novel. To be precise,  $g(z) = z^{\frac{n}{2}+1}$  does not satisfy the steerability condition of [15] when  $n$  is odd since it is not possible to expand odd powers of a square root with a limited number of (integer) angular frequencies. The same goes evidently for gray images generated by the isocurve family represented by linear combinations of real and imaginary parts of  $\log(z)$ , which are not even possible to rotate by changing the linear coefficients. Consequently, the

patterns with odd  $n$  or with  $n = -2$  are not possible to generate by weighted sums of a low number of *steerable functions*. In turn, this makes it impossible to detect the mentioned patterns by correlating the *original gray images* with steerable filters. Yet, as a consequence of Lemma 2, these patterns can be exactly generated by *analytic functions*<sup>2</sup> and detected by correlating the *structure tensor* field with steerable filters. We can conclude that, Lemma 2 shows as a byproduct, that the angular band-limitedness condition of [15] is sufficient but not necessary for steering the rotation of a 2D pattern, since there exist functions whose isocurves are steerable without that the corresponding 2D gray function or its isocurves are band-limited angularly. Furthermore, there exists a pattern family, the one with isocurves given by  $\log(z)$ , that are not steerable at all, but yet are detectable by correlating *steerable filters* with the structure tensor field.

The results of the next section can be viewed as a further build-up of the theory and practice presented in [3], [22], [25]. We model isocurves by harmonic functions as in [3] and obtain filters that detect them by means of the symmetry derivatives. Our separable filters estimating  $I_{20}$ , given in the technical report [4], are similar to those suggested by Johansson [22] to model the orientation in the image. The main novelties of the next section compared to the latter study can be summarized as 1) we enhance  $I_{20}$  that encodes the orientation of the pattern with an additional error measurement,  $I_{11}$  so that these are fully equivalent to the three (real) elements of the generalized structure tensor, and 2) through the tensor, we fit a harmonic curve family to the *isocurves* of the image that satisfies a least squares optimality criterion.

## 5 APPLICATIONS AND EXPERIMENTAL RESULTS

### 5.1 Symmetry Tracker

In vehicle crash tests, the test event is filmed with a high speed camera to quantify the impact of various parameters on human safety by tracking markers. A common marker is the "cross" which allows to quantify the planar position of an object as well as its planar rotation, see Fig. 3c. Markers have to be tracked across numerous frames (in the order of hundreds to thousands). The tracking has to be fast and robust in that the markers should not be lost from frame to frame. The rotations and translations of the objects are not constant due to the large accelerations/decelerations, while severe light conditions are common between two frames (e.g., imperfect flash synchronization). We first present the spatial model of the crosses.

The cross pattern will be detected by applying Lemma 2 and using the harmonic function  $z^{\frac{n}{2}+1}$  with  $n = 2$ . Real and imaginary parts of this function are  $\xi = x^2 - y^2$ ,  $\eta = 2xy$ , and can be used to build marker like images from any nontrivial 1D function  $s(t)$  through the substitution  $s(a\xi + b\eta)$ . By using  $s(t) = t$ , for example, we can obtain Fig. 3b, illustrating such a synthesis. The isocurves of such images will be hyperbolas,  $a\xi + b\eta = \text{constant}$ , which also include, a pair of orthogonal lines (the asymptotes) if the *constant* approaches to zero. The ideal cross markers constitute a subset of this family which are generated by choosing  $s(t)$  as the step function  $\chi$ ,

2. Analytic functions are not necessarily steerable.

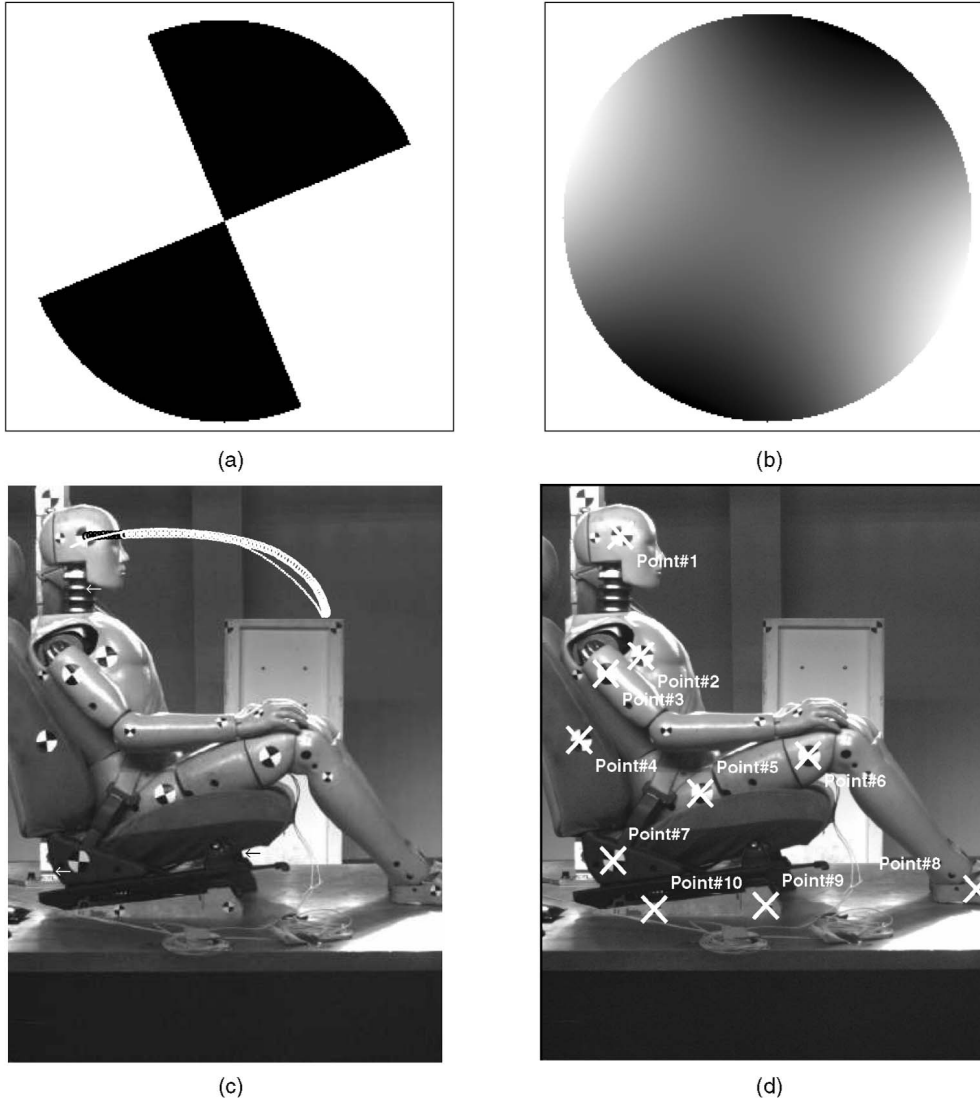


Fig. 3. (a) The ideal model with  $\varphi = \frac{\pi}{8}$ . (b) The gray tones in the disk change linearly with the function  $\sin(2\varphi)(x^2 - y^2) - \cos(2\varphi)2xy$ , where  $\varphi = \frac{\pi}{8}$ . Upon thresholding its intensity (at 0), the image in (a) is obtained. (c) The first frame of an image sequence with the identified forward, “o,” (in white or black, appearing as a thick curve due to overlap) and backward, “.” (appearing as a thin curve due to overlap), trajectory of one of the crosses, attached to the head. The three arrows (two white and one black) illustrate corner points that are not crosses. (d) The identified crosses.

$$s(t) = \chi(t) = \begin{cases} 1, & \text{if } t > 0 \\ 0, & \text{otherwise,} \end{cases} \quad (31)$$

as shown in Fig. 3a. The rotation angle of the cross is steered by the proportion of  $a$  versus  $b$ . Regardless  $s$ , the isocurves of  $s(a\xi + b\eta)$  are parallel lines in hyperbolic coordinates, which in turn yield a concentration of the power to a common line in the frequency spectrum with regard to  $\xi, \eta$  coordinates. Through Theorem 3 and Lemma 2, we can “reverse” the synthesis process and see if a given image has such a power concentration without actually knowing  $s$ . We can measure the orientation of this concentration and the goodness of fit of a line to it. Consequently, the choice of  $s$  as a step function, a linear function, a cosine function, etc., does not influence the detection process, for which all of these images belong to the same family because they have a common isocurve family. The orientation of the orthogonal asymptotes of the hyperbolas is encoded in the argument of  $I_{20}$ , whereas the goodness of the fit of a “line” in hyperbolic coordinates is encoded in the magnitudes of  $I_{20}$  and  $I_{11}$ .

We have used the rotation invariant certainty (see Theorem 3)

$$C_r = (\lambda_{max} - \lambda_{min})(\lambda_{max} + \lambda_{min}) = (\lambda_{max}^2 - \lambda_{min}^2) = |I_{20}|/I_{11}, \quad (32)$$

which has a maximum when  $\lambda_{min} = 0$ . It is straightforward to construct alternative certainty measures, e.g., the dimensionless  $C'_r = (\lambda_{max} - \lambda_{min})(\lambda_{max} + \lambda_{min}) = (\lambda_{max}^2 - \lambda_{min}^2) = |I_{20}|/I_{11}$  that becomes maximum when  $\lambda_{min} = 0$ . Our particular choice of certainty was influenced by the desire to have a measure with a dynamic range that is comparable to that of the operator suggested by [17], discussed below. The point having the largest certainty  $C_r$  will represent the position of the cross marker in our tracking algorithm which we state as follows:

**Algorithm:** Tracking consists of five steps and all steps are carried out in a region of interest, the search window.



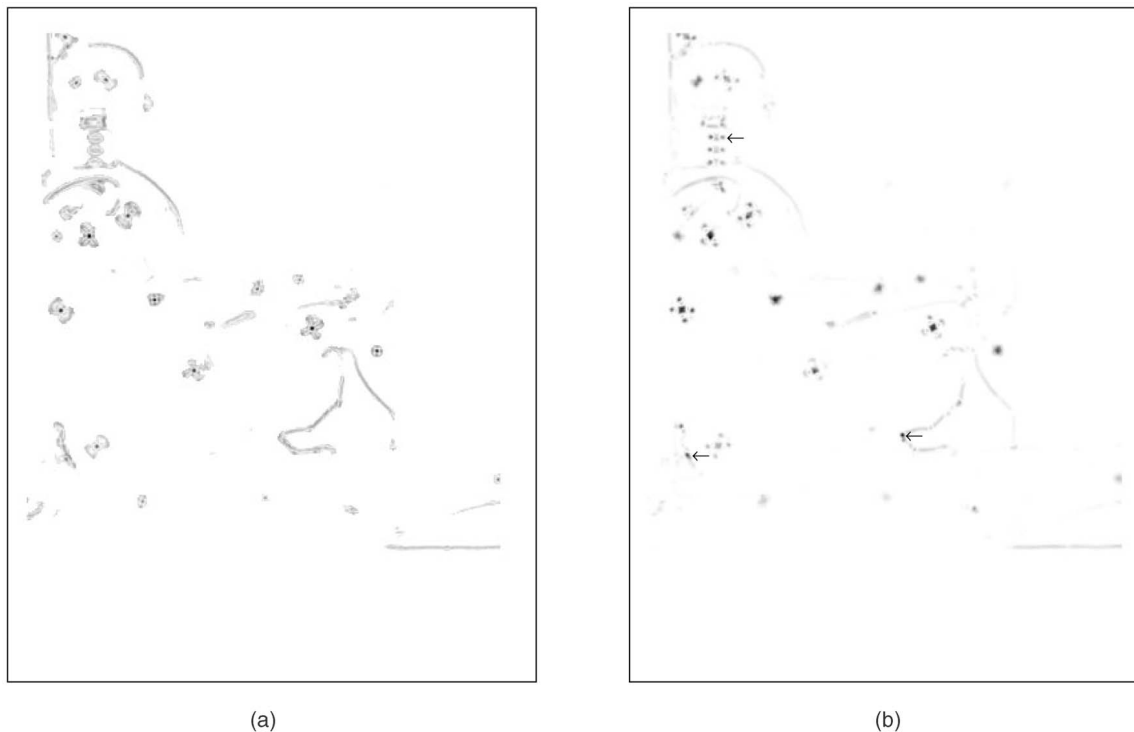


Fig. 4. (a) The certainty parameter of the symmetry tracker. (b) The response of a "corner" detector. The arrows point to example points that are not cross markers (also marked in Fig. 3c), but are "corners."

1. Compute the complex image  $h = (f_x + if_y)^2$  by separable 1D convolutions,  $\Gamma^{\{1, \sigma_1\}} * f$ , and pixel-wise complex squaring.
2. Compute the complex image  $I_{20}$ , (21), by convolving the complex image  $h$  with the complex filter of  $\Gamma^{\{n, \sigma_1^2 + \sigma_2^2\}}$ , using separable convolutions.
3. Compute the scalar (nonnegative) image  $I_{11}$  by convolving the magnitude of the complex image of Step 1, with the magnitude of the complex filter of Step 2 via separable convolutions, (22).
4. Compute the certainty image  $C_r$ , (32), using the images  $I_{20}$  and  $I_{11}$ , obtained in the previous two steps.
5. Compute the maximum of  $C_r$  in the search window to obtain the position of the marker.

Evidently,  $n = 2$  was utilized in  $\Gamma^{\{n, \sigma_1^2 + \sigma_2^2\}}$  of Steps 2 and 3. All steps are applied to all search windows. The maximum  $C_r$  in the search window was identified so that the search window could be recentered around the found maximum. Initially (in frame 1), the search windows, containing one marker each, are found automatically, so that the user only validates the results and starts the tracking. That is  $C_r$ s for the entire image are computed and markers are suggested to the human expert by thresholding. Later, to keep one marker per search window, the window sizes are automatically reduced if they overlap during tracking. The parameters determining the filters were  $\sigma_1 = 0.9$  and  $\sigma_2 = 1.3$  in the convolutions devised by Step 1 through Step 3.

*Tracking results and comparisons:* A primary motivation to automatize the tracking of the moving objects has been to save time for experts by minimizing their manual interventions while offering them a good quantification of the motion details. Consequently, the most important issue is

the robustness of the tracking with minimum manual intervention in near real time execution over many frames. For example, the cross-marker should not be lost from frame to frame, i.e., the centers of the crosses must be well identified, despite occasional but severe illumination changes due to glitches in flash synchronization. Fig. 3d shows a typical frame of the marker tracking process, superimposed to the original frame in Fig. 3c. The found trajectory of one marker (Point #1) throughout the image sequence is superimposed the first frame, Fig. 3c. By its continuity, the trajectory indicates that the marker was not lost. That the found points were also accurately positioned, were manually verified in that the distance between a found and its corresponding true point was within a tolerance threshold, which was a quarter of the radius of the currently tracked cross-marker. The displayed cross marker (Point #1) was 10 pixels in radius and its displayed motion was tracked in 294 frames. Furthermore, 100 cross markers coming from several crash tests were inspected with the same criterion on positioning accuracy as Point #1, to test the ability of tracking without losing the marker. The symmetry tracker could track all but five cross markers of 100 throughout the sequences. The used test sequences included very long image sequences (in the order of thousands of frames) as well as medium long sequences (in the order of hundreds of frames). The symmetry tracker typically lost a marker when the contrast level of a cross was extremely poor, usually due to a strong specular reflection or poor illumination, caused by the imperfections in flash light synchronization with the camera.

Fig. 4a illustrates the measurements (certainties), (32), used by our symmetry tracker, whereas Fig. 4b represents the alternative tracking measurement:



Fig. 5. The “+” and the “□” represent the Delta and Core (parabola like) points that have been automatically extracted in several fingerprint images. Top-left and bottom-left are two fingerprints of the same finger but differing significantly in quality. The certainties are 0.84 (at +) and 0.64 (at □) in top-left. The certainties are 0.73 (□), 0.22 (at +) and 0.10 (at “o”) in bottom-left.

$$C_{hs} = \lambda_{max}\lambda_{min} - 0.04(\lambda_{max} + \lambda_{min})^2. \quad (33)$$

This measure was suggested by Harris and Stephens [17] with the used eigenvalues being identical to those of the *ordinary* structure tensor proposed by Bigun and Granlund, [6], to quantify the *Cartesian* linear symmetry, see Theorems 1 and 2. We can write the identity

$$\begin{aligned} C_{hs} &= (I_{11} + |I_{20}|)(I_{11} - |I_{20}|)/4 - 0.04I_{11}^2 \\ &= (0.84I_{11}^2 - |I_{20}|^2)/4 \end{aligned} \quad (34)$$

to represent this corner detector in terms of the ordinary structure tensor. Consequently, the  $I_{20}$  and  $I_{11}$  measurements used in Figs. 4a and 4b represent different things, i.e., the second order spectral energy moments in the hyperbolic and in the Cartesian coordinates, respectively. In other words,  $I_{20}$  and  $I_{11}$  are obtained through (21) and (22) with  $n = 0$  for  $C_{hs}$  and with  $n = 2$  for  $C_r$ . Since the inequality  $|I_{20}| \leq I_{11}$  will be fulfilled with equality for linearly symmetric images, the Harris and Stephens scheme also includes a threshold to prevent  $C_{hs}$  from becoming negative.

Apart from the used curve families, there is also another rationale by which the two certainties,  $C_r$  and  $C_{hs}$  differ. Noting that the certainty for existence of linear symmetry in the image was defined by [6] as the condition  $\lambda_{min} = 0$ , it can be concluded that the Harris and Stephen contribution provides a measure for the *lack of* linear symmetry in the image. This is because  $C_{hs}$  attains its maximum when  $\lambda_{min}$  is farthest away

from 0, i.e., when  $\lambda_{min} = \lambda_{max}$ , which becomes evident when attempting to maximize  $C_{hs}$  with regard to  $\lambda_{min}$  and  $\lambda_{max}$  when the contrast energy is fixed,  $(\lambda_{max} + \lambda_{min}) = Constant$ . All harmonic patterns with  $n \neq 0$ , Fig. 1, are then valid stimuli for  $C_{hs}$  because they lack linear symmetry. For this reason, the class of patterns detected by  $C_{hs}$  includes, but not only consists of, cross-markers. Consequently,  $C_{hs}$  has an elevated risk for false acceptance of noncross-markers in comparison with  $C_r$ . Examples of such points that are falsely accepted upon accepting the Point #1 based on the strength of  $C_{hs}$ , are shown by the three arrows in Fig. 4b and in Fig. 3c. Because the symmetry tracker is more specific about the corner types that it responds to, the spreads of the detected positions in Fig. 4a are also smaller than those in Fig. 4b. This conclusion is fair because  $I_{20}$  and  $I_{11}$  used in these results were obtained by using identical parameters for  $\sigma_1$  and  $\sigma_2$  in both cases. As far as the dynamic range is concerned, both certainties were given similar conditions, in that, they were both functions of second-order polynomials in  $|I_{20}|$ , and  $I_{11}$ . We could not use the same certainty measure, because  $C_{hs}$  represents a lack of a property (linear symmetry in Cartesian coordinates), whereas  $C_r$  represents the presence of a property, (linear symmetry in hyperbolic coordinates). Because of the missing focus to a precise “corner” type, the method suggested by [17] could not be considered as an alternative in the crash-test application we reported here.

By contrast, an alternative method that was considered was *correlation* using an image of a typical cross marker as a

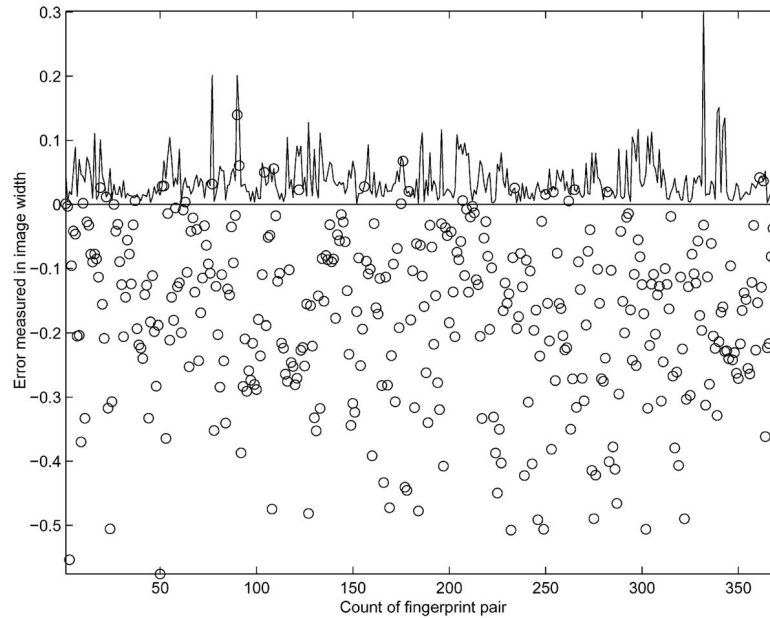


Fig. 6. Alignment errors measured as a fraction of the image width. The solid graph corresponds to the symmetry tracker. The circles correspond to the difference of the symmetry tracker error and that of the model based motion tracker with negative difference indicating inferiority of the model-based motion and vice-versa.

template. The position accuracy was inferior to that of the symmetry tracker when the rotation was considerable between two frames. This is explainable because the correlation is minimum (0, instead of maximum) when a cross rotates  $\pi/2$ . On the other side, using an iterative approach to correlation, e.g., [7], was not permissible because of hardware and time restrictions requiring 1) the number of the iterations be known and 2) more intervention demands of the iterative methods to find the search windows for each cross in the first frame since there are numerous body parts that move independently in the images. The increased manual intervention in the first frame is due to the fact that the motion is only piecewise affine (around each moving object) so that first, the moving objects must be identified. In turn, this requires motion segmentation. By contrast, the symmetry tracker identifies the crosses without the use of motion, i.e., by using only the information in a single frame. Therefore, we only compared the symmetry tracker which can be viewed as orientation field correlation, with the ordinary correlation. As expected, in terms of robustness, the correlation algorithm performed poorly. It has lost the marker during tracking in 46 cases of 100.

## 5.2 Fingerprint Alignment

In biometric authentication, alignment of two fingerprints without extraction of minutiae<sup>3</sup> has gained increased interest, e.g., [19] since this improves the subsequent person authentication (minutiae based or not) performance substantially. Besides improved accuracy, this eliminates the costly combinatorial match of fingerprint minutiae. Recently, silicon-based imaging sensors have become cheaply available. However, because sensor surfaces are decreasing, in order to accommodate them to portable devices, e.g., mobile phones, the delivered images of the fingerprints are small too. In turn, this results in fewer minutiae points that

3. Typically, a minutia point is the end of a line or bifurcation point of two lines.

are available to consumer applications which is an additional reason for why nonminutiae-based alignment techniques in biometric authentication have gained interest.

A high automation level of accurate fingerprint alignment is desirable independent of which matching technique is utilized. For robustness and precision, we suggest to automatically identify two standard landmark types: Core and Delta, see Fig. 5a. These can be modeled and detected by symmetry derivative filters in a scheme based on Lemma 2 that is similar to the five steps scheme presented in the previous section. Naturally, we used coordinate transformations which are different than the one modeling the cross marker. Furthermore, the detection was performed within a Gaussian pyramid scheme and by using the certainty

$$C_r' = \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}} = \frac{|I_{20}|}{I_{11}} \quad (35)$$

to improve the signal to noise ratio. The real and imaginary parts of the analytic functions  $z^{\pm 1/2}$ , i.e.,  $n = -1$ , and  $z^{\pm 3/2}$ , i.e.,  $n = 1$  were used to model Core and Delta, respectively, compare Figs. 1 and 5. The details are given in the Algorithm and the postprocessing paragraphs below.

**Algorithm:** *The following steps are applied sequentially.*

1. Obtain the square of the derivatives via a convolution with a  $5 \times 5$  separable filter and complex squaring:  $h_k = (\Gamma_k^{1,\sigma_1} * f_k)^2$ . Then, build a Gaussian octave pyramid of  $h_k$  image (level 1 corresponds to the original size) up to level 3. The pyramid is built to improve the signal to noise ratio.
2. Convolve the highest level of the pyramid with the filters  $\Gamma^{\{-1,1.5\}}$  for Core detection, and  $\Gamma^{\{1,1.5\}}$  for Delta detection, (both filters  $9 \times 9$ ) to obtain  $I_{20}$  for each landmark type.
3. At the top of the pyramid and for each landmark, compute the (positive) image  $I_{11}$  by convolving the magnitude of the complex image of Step 1, with the

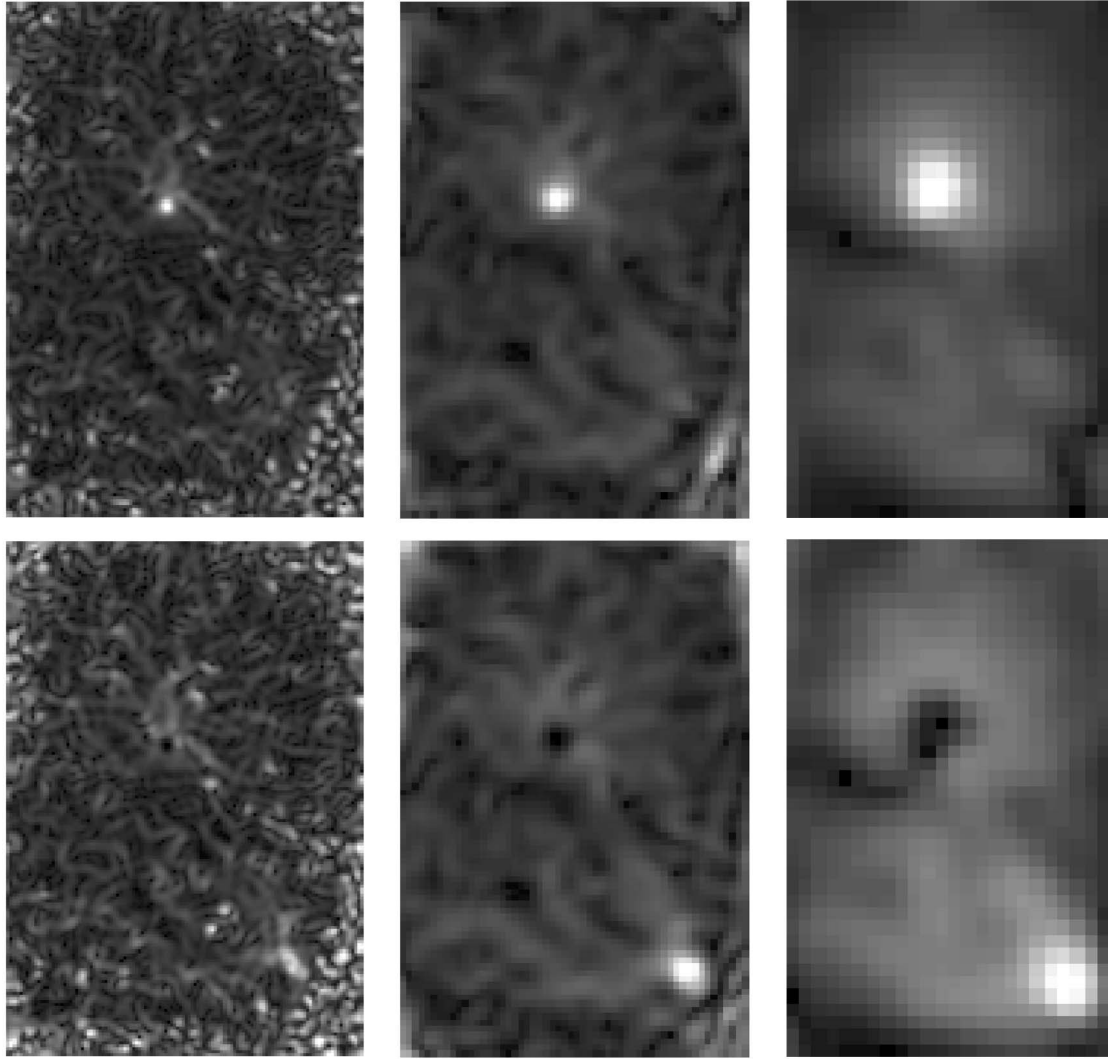


Fig. 7. The top row shows the certainty images  $C'_r$  of (35), for core patterns, at three resolutions, decreasing in octaves from left to right for Fig. 5, top left. The bottom row shows the same, but for the Delta patterns.

magnitude of the complex filter of Step 2 via convolutions, (22).

4. At the top of the pyramid and for each landmark, compute the certainty image,  $C'_r$ , by pixel-wise division according to (35) using the images,  $I_{20}$  and  $I_{11}$ , obtained in the previous two steps.
5. At the top of the pyramid and for each landmark type, compute the maximum of  $C'_r$  in the image to obtain the position of the two candidate landmarks.

**Postprocessing:** Once the landmark positions and orientations were estimated at the top of the pyramid, the position parameters were fine-tuned by projection to a lower level. In that level, by carrying out computations analogous to those in Steps 2-5 of the Algorithm, but applied in a  $13 \times 13$  window centered around the position to be fine-tuned, the maximum  $C'_r$ 's were found to update the positions of the two landmarks. The process was repeated until level 1, where we obtained the final positions of the landmarks, was reached. The same procedure was applied to the second fingerprint image to be aligned. The translation parameter between the two images to be aligned is obtained by the difference of the positions of the corresponding landmarks. The rotation parameter was obtained via the arguments of the complex scalars,  $I_{20}$ , at the

refined (maximum certainty) positions. This was done by subtracting the thus obtained two angles. If both of the landmarks were detected in both fingerprints with certainties above 0.5, we have used the translation and orientation parameters of the landmark having the highest certainty of the two.

*Alignment results and comparisons:* We report results on the publicly available FVC2000 fingerprints database, collected by [30] for benchmarking. The FVC2000 contains a total of 800 fingerprints, many having a poor quality since they were captured using a low cost capacitive sensor from 100 persons, e.g., see Fig. 5, top-left and bottom-left for two fingerprints of the same finger. The significant quality changes that can be observed in this database correspond to the actual systems in use, and stem from external variations, e.g., significant pressure variation between the imprints, humidity variations in the fingers, foreign particles such as dirt, fat, and dust, etc. The found positions of Delta and Core points are illustrated by the two images of Fig. 5, top-left and bottom-left. The example also shows the performance in case of severe noise. The Delta in the poor quality fingerprint is difficult to identify even for human observers. The Cores were detected correctly in both images whereas another (false) point was suggested

instead of the correct Delta in the poor quality fingerprint. However, despite being maximum, the certainty for this point being a true Delta point was low, 0.22. The false as well as the true point are below the used threshold of 0.5 so that this point was rejected by the system and only the point marked with "□" has been used to estimate the position and orientation parameters. Four typical detections, for average quality fingerprints, are shown in other four images of Fig. 5 as performance illustration. In Fig. 7, the refinements of Core and Delta positions within the pyramid scheme are shown. The low resolution images are resized to the original size. The images correspond to the fingerprint in the top-left of Fig. 5.

The fingerprints of FVC2000 have been translated and rotated with the parameters computed by our method toward one of the fingerprints automatically. For each such fingerprint pair (same finger), an arbitrarily chosen strong minutia point *other* than the automatically found landmark (Core and/or Delta) was manually identified in the reference as well as in the aligned image. To quantify the registration errors, we used the translational distance between the manually identified points in the automatically aligned image pair. In Fig. 6, these registration errors of our symmetry aligner,  $e_{sa}$ , are plotted in solid. The horizontal axis is the index of the fingerprint image pairs. The vertical axis is the error measured as the distance between the two manually identified minutiae in the automatically aligned image pair. The used distance unit was the fingerprint image width, 256 pixels. The images had the height of 364 pixels. The errors at the minutiae are important to reduce since such neighborhoods are later used to perform person authentication, e.g., by means of a fine-tuned correlation and/or minutiae extraction. The displayed error includes a linear amplification of the error which increases with the distance of the manually identified point from the core/delta point used when rotating the test image to achieve the alignment. In other words, a fine tuning reducing these errors significantly in the person-authentication phase but obscuring a comparison with another technique that is presented here next, is not included.

We have also tested another method which employs (affine) model-based motion estimation and robust statistics to find the alignment parameters between two images. For a description of robust motion parameter estimation techniques, we refer to [7]. Here, we have used a similar technique to the one presented in [7] using the M-estimator as regressor. Other tested regressors (LS, LMS, and LTS) were worse and are omitted here to save space. In an attempt to improve the graphical comparison, we have avoided to draw a curve for the error of this technique. Instead, we have drawn the circles in Fig. 6 at data points representing the difference of the alignment errors,  $e_{sa} - e_{me}$ , where  $e_{sa}$  drawn in solid, is the error of the symmetry aligner whereas  $e_{me}$  is the registration error of the motion aligner. Consequently, when the circles are in the negative territory of the vertical axis, the  $e_{sa}$  is smaller than  $e_{me}$ . The figure shows that nearly all reported alignment errors (0.93 percent of the total  $N = 371$  fingerprint pairs in the graph), were significantly lower with the symmetry tracker than with the model-based motion tracker. As the figure shows, in those cases, the improvement factor of the alignment,  $(e_{me} - e_{sa})/e_{sa}$  is, on average, close to 3. To the best of our knowledge, other results on *alignment* are not available on publicly available databases.

## 6 CONCLUSION

We have suggested a complex derivative operator that we called symmetry derivative. Through theorems and lemmas, we have revealed its invariance properties. Via two applications, we have illustrated the interest and the usage of the theorems in practical situations. We conclude that the symmetry derivatives of Gaussians are simple to obtain on a Cartesian grid analytically and in a closed form since the polynomial factor is the same as the one given by the derivation polynomial. Furthermore, they are invariant under Fourier Transform and Convolution which make them unique when designing separable FIR filters because 1) they do not introduce orientation bias or other artifacts on digital images (isotropic) and 2) they are concentrated in both spatial and frequency domains. The invariance of the symmetry derivatives of Gaussians to Fourier transform, together with the preservation of the scalar products (Parseval-Plancherel) allows interpretation of their quadratic forms as second-order moments of the power spectrum. In turn, this interpretation makes them an effective tool (the generalized structure tensor) for fitting an "axis" to the power spectrum directly in the spatial domain in the least square error sense.

The symmetry derivatives used in conjunction with Gaussians revealed that monomials and their square roots are particularly efficient to model and detect intricate patterns almost as efficiently as detecting lines. This is because, an estimation of the position, orientation, and error difference parameters with a fixed number of 1D filters, and with a fixed number of arithmetic operations has been achieved.

The presented results show the usefulness of symmetry derivatives 1) as an analytical tool having convenient invariants, 2) a tool with which we revealed unknown properties of steerable filters, and 3) as a way to represent and compute tensor fields, in particular the generalized structure tensor fields, for pattern recognition applications. Other potential benefactors include the theory and practice of texture analysis. Efficient texture features can be obtained by use of local spectral complex moments with orders higher than 2, which hitherto have been implemented by the use of Gabor power spectrum [5]. These higher-order features could be computed in the spatial domain more efficiently by use of symmetry derivatives.

## APPENDIX A

### PROOF OF THEOREM 4 AND LEMMA 1

We prove the theorem by induction.

1. Equation (14) holds for  $p = 1$ .
2. Assume (induction) that (14) is true for  $p = p_0$ , for some  $p_0 \geq 1$ .
3. Then,

$$\Gamma^{\{p_0+1, \sigma^2\}}(x, y) = \frac{1}{2\pi\sigma^2} (D_x + iD_y)(D_x + iD_y)^{p_0} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (36)$$

$$= \frac{1}{2\pi\sigma^2} (D_x + iD_y) \left[ \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \right], \quad (37)$$

where we indicated by brackets  $[\ ]$  the terms on which the differential operators act is obtained by applying the induction assumption (Point 2). Consequently, and by using the chain rule as well as the linearity of the partial differential operators we obtain,

$$\begin{aligned} \Gamma^{\{p_0+1,\sigma^2\}}(x,y) &= \frac{1}{2\pi\sigma^2} \left[ (D_x + iD_y) \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} \right. \\ &\quad \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &\quad \left. + \frac{1}{2\pi\sigma^2} \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} \left[ (D_x + iD_y) \right. \right. \\ &\quad \left. \left. \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \right] \right]. \end{aligned}$$

By repeated applications of the algebraic rules that govern the differential operators, we obtain

$$\begin{aligned} \Gamma^{\{p_0+1,\sigma^2\}}(x,y) &= \frac{1}{2\pi\sigma^2} \left[ D_x \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} + iD_y \right. \\ &\quad \left. \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} \right] \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &\quad + \sigma^2 \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0} \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right) \\ &\quad \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &= \frac{1}{2\pi\sigma^2} \left( p_0 \left( \frac{-1}{\sigma^2} \right) \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0-1} \right. \\ &\quad \left. + i^2 p_0 \left( \frac{-1}{\sigma^2} \right) \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0-1} \right) \\ &\quad \times \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) + \frac{1}{2\pi\sigma^2} \\ &\quad \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0+1} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &= \frac{1}{2\pi\sigma^2} \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^{p_0+1} \\ &\quad \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right). \end{aligned}$$

Consequently, when it holds for  $p = p_0$ , (14) will also hold for  $p = p_0 + 1$ .

Now, we turn to the general case, Lemma 1.

$$\begin{aligned} Q(D_x + iD_y) \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) &= \frac{1}{2\pi\sigma^2} \left[ \sum_{n=0}^{N-1} a_n \right. \\ &\quad \left. (D_x + iD_y)^n \right] \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \\ &= \frac{1}{2\pi\sigma^2} \sum_{n=0}^{N-1} a_n \left[ (D_x + iD_y)^n \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \right] \\ &= \frac{1}{2\pi\sigma^2} \sum_{n=0}^{N-1} a_n \left( \frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2} \right)^n \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \end{aligned} \quad (38)$$

$$= Q\left(\frac{-x}{\sigma^2} + i \frac{-y}{\sigma^2}\right) \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (39)$$

in which the linearity of derivative operators and (14) have been used.  $\square$

## APPENDIX B

### Proof<sup>4</sup> of Theorem 6

Convolution in the Fourier transform domain reduces to an ordinary multiplication:

$$\Gamma^{\{p_1,\sigma_1^2\}} * \Gamma^{\{p_2,\sigma_2^2\}} \leftrightarrow \mathcal{F}[\Gamma^{\{p_1,\sigma_1^2\}}] \mathcal{F}[\Gamma^{\{p_2,\sigma_2^2\}}]. \quad (40)$$

By using (19), we note that

$$\mathcal{F}[\Gamma^{\{p_1,\sigma_1^2\}}](\omega_x, \omega_y) = 2\pi \left( \frac{-i}{\sigma_1^2} \right)^{p_1} \sigma_1^2 \Gamma^{\left\{p_1, \frac{1}{\sigma_1^2}\right\}}(\omega_x, \omega_y).$$

Due to (14), we can write:

$$\begin{aligned} \mathcal{F}[\Gamma^{\{p_1,\sigma_1^2\}}] \cdot \mathcal{F}[\Gamma^{\{p_2,\sigma_2^2\}}] &= 2\pi \left( \frac{-i}{\sigma_1^2} \right)^{p_1} \sigma_1^2 \Gamma^{\left\{p_1, \frac{1}{\sigma_1^2}\right\}} \\ &\quad (\omega_x, \omega_y) \cdot 2\pi \left( \frac{-i}{\sigma_2^2} \right)^{p_2} \sigma_2^2 \Gamma^{\left\{p_2, \frac{1}{\sigma_2^2}\right\}}(\omega_x, \omega_y) \\ &= 2\pi\sigma_1^2 \left( \frac{-i}{\sigma_1^2} \right)^{p_1} (-\sigma_1^2)^{p_1} (\omega_x + i\omega_y)^{p_1} \frac{1}{2\pi\sigma_1^2} \\ &\quad \exp\left(-\sigma_1^2 \frac{\omega_x^2 + \omega_y^2}{2}\right) \times 2\pi\sigma_2^2 \left( \frac{-i}{\sigma_2^2} \right)^{p_2} (-\sigma_2^2)^{p_2} (\omega_x + i\omega_y)^{p_2} \frac{1}{2\pi\sigma_2^2} \\ &\quad \exp\left(-\sigma_2^2 \frac{\omega_x^2 + \omega_y^2}{2}\right) \\ &= \left( \frac{-i}{\sigma_1^2} \right)^{p_1} \left( \frac{-i}{\sigma_2^2} \right)^{p_2} (-\sigma_1^2)^{p_1} (-\sigma_2^2)^{p_2} (\omega_x + i\omega_y)^{p_1+p_2} \\ &\quad \exp\left(-\frac{\omega_x^2 + \omega_y^2}{2 \frac{1}{\sigma_1^2 + \sigma_2^2}}\right). \end{aligned}$$

Canceling the redundant terms allows us to write the product as

$$\begin{aligned} \mathcal{F}[\Gamma^{\{p_1,\sigma_1^2\}}] \cdot \mathcal{F}[\Gamma^{\{p_2,\sigma_2^2\}}] &= (i)^{p_1+p_2} (\omega_x + i\omega_y)^{p_1+p_2} \\ &\quad \exp\left(-\frac{\omega_x^2 + \omega_y^2}{2 \frac{1}{\sigma_1^2 + \sigma_2^2}}\right) \\ &= \frac{2\pi(\sigma_1^2 + \sigma_2^2)}{2\pi(\sigma_1^2 + \sigma_2^2)} (i)^{p_1+p_2} (\omega_x + i\omega_y)^{p_1+p_2} \exp\left(-\frac{\omega_x^2 + \omega_y^2}{2 \frac{1}{\sigma_1^2 + \sigma_2^2}}\right) \\ &\quad \frac{(-\sigma_1^2 - \sigma_2^2)^{p_1+p_2}}{(-\sigma_1^2 - \sigma_2^2)^{p_1+p_2}} \\ &= 2\pi(\sigma_1^2 + \sigma_2^2) \left( \frac{i}{-\sigma_1^2 - \sigma_2^2} \right)^{p_1+p_2} \Gamma^{\left\{p_1+p_2, \frac{1}{\sigma_1^2 + \sigma_2^2}\right\}}(\omega_x, \omega_y). \end{aligned} \quad (41)$$

Remembering (40), we now inverse Fourier transform (41) by using (19) and obtain

$$\Gamma^{\{p_1,\sigma_1^2\}} * \Gamma^{\{p_2,\sigma_2^2\}} = \Gamma^{\{p_1+p_2,\sigma_1^2+\sigma_2^2\}}.$$

$\square$

4. We can ignore nearly all steps to conclude that the theorem holds, provided that the following, which is what is proven, are granted: 1) convolution and symmetry derivatives are distributive, i.e.,  $(D_x + iD_y)[f * g] = [(D_x + iD_y)f] * g = f * [(D_x + iD_y)g]$  and 2) the Gaussian convolutions are variance additive, i.e.,  $\Gamma^{\{0,\sigma_1^2\}} * \Gamma^{\{0,\sigma_2^2\}} = \Gamma^{\{0,\sigma_1^2+\sigma_2^2\}}$ .

## APPENDIX C

### PROOF OF LEMMA 2

We use Theorem 3 to estimate  $I_{20}$ , for which  $(D_x - iD_y)\xi$  is needed.<sup>5</sup> We write the coordinates as a complex variable  $z = x + iy$  and establish a relationship between symmetry derivatives and complex derivatives as follows:

$$\begin{aligned} (D_x - iD_y)\Re[g(z)] &= D_x[\Re g(z)] - iD_y[\Re g(z)] = \Re[D_x g(z)] \\ &\quad - i\Re[D_y g(z)], \\ &= \Re\left[\frac{dg}{dz} \frac{dz}{dx}\right] - i\Re\left[\frac{dg}{dz} \frac{dz}{dy}\right] = \Re\left[\frac{dg}{dz}\right] \\ &\quad - i\Re\left[\frac{dg}{dz} i\right], \\ &= \Re\left[\frac{dg}{dz}\right] - i\Re\left[i\Re\left[\frac{dg}{dz}\right] - \Im\left[\frac{dg}{dz}\right]\right] = \Re\left[\frac{dg}{dz}\right] \\ &\quad + i\Im\left[\frac{dg}{dz}\right] = \frac{dg}{dz}. \end{aligned}$$

But,  $\frac{dg}{dz} = z^{\frac{n}{2}}$  so that we can obtain the complex exponential as:

$$e^{i \arg((D_x - iD_y)\xi^2)} = e^{i \arg\left(\frac{dg}{dz}\right)} = e^{i \arg(z^n)} = \frac{z^n}{|z^n|} = e^{in \arg(x+iy)}.$$

Consequently, the expression  $I_{20}$  in (7) of Theorem 3 reduces to

$$\begin{aligned} I_{20} &= \iint ((D_\xi + iD_\eta)f)^2 d\xi d\eta \\ &= \iint e^{in \arg(x+iy)} [(D_x + iD_y)f]^2 dx dy. \end{aligned} \quad (42)$$

We assume that  $[(D_x + iD_y)f]^2$  is discretized on a Cartesian grid and use a Gaussian as interpolator<sup>6</sup> to reconstruct it from its samples:

$$h(x, y) = 2\pi\sigma_1^2 \sum_k h_k \Gamma^{\{0, \sigma_1^2\}}(x - x_k, y - y_k). \quad (43)$$

Here,  $h_k$  represents the samples of  $h(x, y) = ((D_x + iD_y)f)^2$  and the constant  $2\pi\sigma_1^2$  serves to normalize the maximum of  $\Gamma^{\{0, \sigma_1^2\}}$  to 1. We incorporate the window function  $K_n|x + iy|^n \Gamma^{\{0, \sigma_2^2\}}(x, y)$ , where  $K_n$  is the constant<sup>7</sup> that normalizes the maximum of the window function to 1, into  $[(D_x + iD_y)f]^2$ , to estimate (42) in a neighborhood.

$$I_{20} = \iint K_n |x + iy|^n e^{in \arg(x+iy)} \Gamma^{\{0, \sigma_2^2\}}(x, y) [(D_x + iD_y)f]^2 dx dy. \quad (44)$$

By assuming  $0 \leq n$  and substituting (43) into (44), we obtain

5. Because  $\xi$  is harmonic,  $\eta$  does not represent a freedom of choice, but it is determined as soon as  $\xi$  is given.

6. This is possible by using a variety of interpolation functions, not only Gaussians. For example, the theory of band-limited functions allows such a reconstruction via the sinc functions. For the benefit of readability, we will not dwell on the details that allow Gaussians to be used as interpolation functions (or, equivalently, as smoothing functions), but will refer to the interpolation and scale space reports of [40], [42], [26], [28].

7. At  $|x + iy| = (n\sigma_2^2)^{1/2}$ , the window  $K_n|x + iy|^n \Gamma^{\{0, \sigma_2^2\}}(x, y)$  attains the value 1 when  $K_n = 2\pi\sigma_2^2 \left(\frac{e}{n\sigma_2^2}\right)^{n/2}$ .

$$\begin{aligned} \frac{I_{20}(x', y')}{(2\pi\sigma_1^2)K_n} &= \iint |x + iy|^n e^{in \arg(x+iy)} \Gamma^{\{0, \sigma_2^2\}}(x, y) \\ &\quad \left[ \sum_k h_k \Gamma^{\{0, \sigma_1^2\}}(x - x_k, y - y_k) \right] dx dy. \end{aligned}$$

Noting that

$$\begin{aligned} |x + iy|^n e^{in \arg(x+iy)} \Gamma^{\{0, \sigma_2^2\}}(x, y) &= (x + iy)^n \Gamma^{\{0, \sigma_2^2\}}(x, y) \\ &= (-\sigma_2^2)^n \Gamma^{\{n, \sigma_2^2\}}(x, y), \end{aligned}$$

where we used the definition of  $\Gamma^{\{n, \sigma_2^2\}}$  in (13) and applied Theorem 4, we can estimate  $I_{20}$  on a Cartesian grid:

$$\begin{aligned} \frac{I_{20}(x', y')}{(2\pi\sigma_1^2)K_n(-\sigma_2^2)^n} &= \sum_k h_k \iint \Gamma^{\{n, \sigma_2^2\}}(x, y) \Gamma^{\{0, \sigma_1^2\}} \\ &\quad (x' - x - x_k, y' - y - y_k) dx dy, \end{aligned} \quad (45)$$

$$\begin{aligned} &= \sum_k h(x_k, y_k) (\Gamma^{\{n, \sigma_2^2\}} * \Gamma^{\{0, \sigma_1^2\}})(x' - x_k, y' - y_k), \\ &= \sum_k h(x_k, y_k) \Gamma^{\{n, \sigma_1^2 + \sigma_2^2\}}(x' - x_k, y' - y_k). \end{aligned} \quad (46)$$

Here, (46) is obtained<sup>8</sup> by utilizing Theorem 6. Equation (46) can be computed on a Cartesian discrete grid by the substitution  $(x', y') = (x_l, y_l)$  yielding an ordinary discrete convolution of  $h$  with a sampled symmetry derivative of a Gaussian, i.e.,  $\Gamma^{\{n, \sigma_1^2 + \sigma_2^2\}}(x_l, y_l)$

$$I_{20}(x_l, y_l) = C_n (h * \Gamma^{\{n, \sigma_1^2 + \sigma_2^2\}})(x_l, y_l) \quad (47)$$

with  $C_n = (2\pi\sigma_1^2)K_n(-\sigma_2^2)^n$ .

The result for  $n < 0$  is straightforward to deduce by following the steps after (45) in an analogous manner. Likewise, the scheme of  $I_{11}$  is obtained by following the same idea as for  $I_{20}$ .  $\square$

## ACKNOWLEDGMENTS

The support of the Swedish SSF-VISIT program is gratefully acknowledged. The authors would like to thank Michael Felsberg, Linköping University, for having pointed at reference [39].

## REFERENCES

- [1] Y.S. Abu-Mostafa and D. Psaltis, "Recognitive Aspects of Moment Invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, pp. 698-706, 1984.
- [2] A.M. Bazen and S.H. Gerez, "Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 905-919, July 2002.
- [3] J. Bigun, "Recognition of Local Symmetries in Gray Value Images by Harmonic Functions," *Proc. Ninth Int'l Conf. Pattern Recognition*, pp. 345-347, Nov. 1988.
- [4] J. Bigun and T. Bigun, "Symmetry Derivatives of Gaussians Illustrated by Cross Tracking," Technical Report HH-IDE-131, Halmstad Univ., SE-30118 Halmstad, 2001.
- [5] J. Bigun and J.M.H. du Buf, "N-Folded Symmetries by Complex Moments in Gabor Space," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 1, pp. 80-87, Jan. 1994.
- [6] J. Bigun and G.H. Granlund, "Optimal Orientation Detection of Linear Symmetry," *Proc. First Int'l Conf. Computer Vision*, pp. 433-438, June 1987.

8. We note that in the proof of Theorem 6, Theorem 5 is needed so that all of the theorems of this paper are actually utilized in the proof of this lemma.

- [7] M.J. Black and P. Anandan, "A Framework for the Robust Estimation of Optical Flow," *Proc. Int'l Conf. Computer Vision*, pp. 231-236, 1993.
- [8] H. Burkhardt, "Transformationen Zur Lageinvarianten Merkmalgewinnung," PhD thesis, Habilitationsschrift, Univ. Karlsruhe, Ersch. als Fortschrittbericht, vol. 10, no. 7, der VDI-Zeitschriften, VDI-Verlag, 1979.
- [9] P. Burt, "Fast Filter Transforms for Image Processing," *Computer Graphics and Image Processing*, vol. 16, pp. 20-51, 1981.
- [10] P.E. Danielsson, Q. Lin, and Q.Z. Ye, "Efficient Detection of Second-Degree Variations in 2D and 3D Images," *Visual Comm. and Image Representation*, vol. 12, pp. 255-305, 2001.
- [11] P.E. Danielsson, "Rotation Invariant Linear Operators with Directional Response," *Proc. Fifth Int'l Conf. Pattern Recognition*, pp. 1171-1176, Dec. 1980.
- [12] D.E. Dudgeon and R.M. Mersereau, *Multi Dimensional Digital Signal Processing*, Englewood Cliffs, N.J.: Prentice Hall, 1981.
- [13] M. Ferraro and T.M. Caelli, "Relationship Between Integral Transform Invariances and Lie Group Theory," *J. Optical Soc. Am. A*, vol. 5, no. 5, pp. 738-742, 1988.
- [14] W. Forstner and E. Gulch, "A Fast Operator for Detection and Precise Location of Distinct Points, Corners and Centres of Circular Features," *Proc. Intercommission Conf. Fast Processing of Photogrammetric Data*, pp. 281-305, 1987.
- [15] W.T. Freeman and E.H. Adelson, "The Design and Use of Steerable Filters," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 9, pp. 891-906, Sept. 1991.
- [16] G.H. Granlund, "Fourier Preprocessing for Hand Print Character Recognition," *IEEE Trans. Computers*, vol. 21, pp. 195-201, 1972.
- [17] C. Harris and M. Stephens, "A Combined Corner and Edge Detector," *Proc. Fourth Alvey Vision Conf.*, pp. 147-151, Aug. 1988.
- [18] W.C. Hoffman, "The Lie Algebra of Visual Perception," *J. Math. Psychol.*, vol. 3, pp. 65-98, 1966.
- [19] L. Hong, Y. Wand, and A.K. Jain, "Fingerprint Image Enhancement: Algorithm and Performance Evaluation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777-789, Aug. 1998.
- [20] M.K. Hu, "Visual Pattern Recognition by Moment Invariants," *IRE Trans. Information Theory*, pp. 179-187, 1962.
- [21] A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, "An Identity Authentication System Using Fingerprints," *Proc. IEEE*, vol. 85, no. 9, pp. 1365-1388, 1997.
- [22] B. Johansson, "Multiscale Curvature Detection in Computer Vision," Lic. thesis no. 877; LIU-TEK-LIC-2001:14, Linköping Univ., Dept. EE (ISY), SE-581 83, Linköping, 2001.
- [23] M. Kass and A. Witkin, "Analyzing Oriented Patterns," *Computer Vision, Graphics, and Image Processing*, vol. 37, pp. 362-385, 1987.
- [24] H. Knutsson, "Representing Local Structure Using Tensors," *Proc. Sixth Scandinavian Conf. Image Analysis*, pp. 244-251, June 1989.
- [25] H. Knutsson, M. Hedlund, and G.H. Granlund, "Apparatus for Determining the Degree of Consistency of a Feature in a Region of an Image that is Divided into Discrete Picture Elements," US patent, 4.747.152, 1988.
- [26] J.J. Koenderink and A.J. van Doorn, "The Structure of Images," *Biological Cybernetics*, vol. 50, pp. 363-370, 1984.
- [27] J.J. Koenderink and A.J. van Doorn, "Receptive Field Families," *Biological Cybernetics*, vol. 63, pp. 291-298, 1990.
- [28] T. Lindeberg, "Discrete Scale-Space Theory and the Scale Space Primal Sketch," PhD thesis, Royal Inst. of Technology, NADA-CVAP S-100 44, Stockholm, 1991.
- [29] B.D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Seventh Int'l Joint Conf. Artificial Intelligence*, pp. 674-679, 1981.
- [30] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.K. Jain, "FVC 2000: Fingerprint Verification Competition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 402-412, Mar. 2002.
- [31] K.V. Mardia, *Statistics of Directional Data*. Academic Press, 1972.
- [32] D. Marr and E. Hildreth, "Theory of Edge Detection," *Proc. Royal Soc. London Bull.*, vol. 204, pp. 301-328, 1979.
- [33] G. Medioni, M.S. Lee, and C.K. Tang, *A Computational Framework for Segmentation and Grouping*. Elsevier, 2000.
- [34] P. Perona, "Steerable-Scalable Kernels for Edge Detection and Junction Analysis," *Proc. European Conf. Computer Vision*, pp. 3-18, 1992.
- [35] A.R. Rao, *A Taxonomy for Texture Description and Identification*. Springer, 1990.
- [36] S.S. Reddi, "Radial and Angular Invariants for Image Identification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 3, no. 2, pp. 240-242, 1981.
- [37] E.P. Simoncelli, "A Rotation Invariant Pattern Signature," *Proc. Int'l Conf. Image Processing*, vol. 3, pp. 185-188, 1996.
- [38] D. Slepian, "Prolate Spheroidal Wave Functions, Fourier Analysis and Uncertainty-IV: Extensions to Many Dimensions; Generalized Prolate Spheroidal Functions," *The Bell System Technical J.*, vol. 43, pp. 3009-3057, Nov. 1964.
- [39] E. Stein and G. Weiss, *Fourier Analysis on Euclidean Spaces*. Princeton, N.J.: Princeton Univ. Press, 1971.
- [40] M. Unser, A. Aldroubi, and M. Eden, "Fast B-Spline Transforms for Continuous Image Representation and Interpolation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 277-285, Mar. 1991.
- [41] R. Wilson and M. Spann, "Finite Prolate Spheroidal Sequences and their Applications II: Image Feature Description and Segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 10, pp. 193-203, Oct. 1988.
- [42] A.P. Witkin, "Scale-Space Filtering," *Proc. Eighth Int'l Joint Conf. Artificial Intelligence*, pp. 1019-1022, Aug. 1983.



**Josef Bigun** received the MSc and PhD degrees from Linköping University in 1983 and 1988, respectively. In 1988, he joined the Swiss Federal Institute of Technology in Lausanne (EPFL) where he worked until 1998 as "adjoint scientifique." He was elected professor to the signal analysis chair, his current position, at Halmstad University and Chalmers Institute of Technology in 1998. His professional interests include biometrics, texture analysis, motion analysis, and understanding of biological processing of audiovisual signals, including human face recognition. Professor Bigun is an elected fellow of the International Association for Pattern Recognition (IAPR) and the IEEE. He contributed to organization of several international conferences as cochairs or track chairs, including the initiation of the conference series Audio and Video-Based Biometric Person Authentication (AVBPA) and organizations of several ICPRs and ICIPs. He has been an editorial member of several journals in the areas of pattern recognition and image understanding.



**Tomas Bigun** received the BSc degree in mathematics from Linköping University in 1993. After his studies, he started to work in Telub AB where he worked until 1995 in the field of radio frequency planning. Then, he joined Saab Combitech AB where he worked in the field of image processing until 1998. In 1998, he joined Intenia R&D AB where he contributed to projects in the field of testing and evaluation until 2001. Since then, he has been working as method and software consultant in TietoEnator AB. During his professional career, he has contributed to several product-near software and hardware projects, as the main investigator in Swedish companies. His professional interests include human-computer interfaces and software design.



**Kenneth Nilsson** received the MS degree in electrical engineering from the Linköping University in 1981. After that, he worked in the industry from 1981-1985 as a consulting engineer in the area of industrial instrumentation, mainly in Boliden Kemi AB, Helsingborg. The industrial robots were the main field of his contributions. Since 1986, he has an appointment as an assistant professor of industrial electronics at Halmstad University. In 1995, he received the Lic.Tech. degree in computer engineering from Chalmers University of Technology. The title of the Lic. dissertation was "Time-Deterministic Communication in a Computer System for Embedded Real-Time Applications." His professional interests are in the field of digital signal and image processing, especially biometrics. In this field, he is active in teaching. He is currently pursuing PhD studies in the field of biometrics. He is a student member of the IEEE.