A THEORETICAL COMPARISON OF DIFFERENT ORIENTATION TENSORS

B. Johansson, G. Farnebäck

Computer Vision Laboratory
Dept. of Electrical Engineering
Linköping University
{bjorn,gf}@isy.liu.se

ABSTRACT

Orientation tensors is a powerful representation of local orientation. Over the years, several different approaches to estimate the tensors have appeared. The derivations of the different tensors vary to a great extent. This partly obstructs a theoretical comparison between them, which otherwise would be useful when one wants to choose the best tensor for a particular application. This paper shows that all the existing tensors can be derived using a common framework. The derivation is based on signal models and the concept of orientation functionals. The idea is to estimate a signal model and compute a suitable orientation functional in terms of the model parameters. The models used in this paper are polynomial models and quadrature models. This framework may also aid in the design of orientation tensors based on other signal models.

1. INTRODUCTION

Orientation tensors have been used in many computer vision applications. Examples are velocity estimation (optical flow) [1, 2, 3, 4], adaptive filtering [5], and as part of the process of locating local image features such as corners and junctions [6, 7].

There now exist several methods to compute the tensors in practice. The earliest one is based on the image gradient, see e.g. [8, 9, 4]. This tensor is often called the structure tensor, but this term is not exclusively used for this type of tensor. We will therefore refer to this tensor as the *gradient tensor*. The tensor in [5] is based on quadrature filters and will here be called the *quadrature tensor*. The third tensor described in [2] is based on a polynomial expansion model of the signal, hence referred to as the *polynomial tensor*.

All of the tensors give the correct orientation when applied to a simple signal (defined in section 3). They are in addition to that designed with different properties in mind. The quadrature tensor is the only one that is *phase invariant* for simple signals, which means that the norm of the tensor is invariant to the signal phase. The polynomial tensor on the other hand is designed to be positive semidefinite and *rotation equivariant* for all type of signals. Rotation equivariance means that a rotation of the signal implies a corresponding rotation of the tensor. The gradient tensor is also positive semidefinite and rotation equivariant for all types of signals, while the same properties for quadrature tensors is only guaranteed for simple signals.

The last two tensors are implicitly or explicitly based on different signal models, which have been used for other applications as well. The quadrature filters have also been used in phase estimation applications such as disparity estimation and stereo vision [5]. The polynomial expansion model has also been used for disparity estimation [10] and detection of rotational symmetries [11]. A closely related model, called the cubic facet model, has also been used for local image curvature estimation [12].

It turns out that all the tensors mentioned above can be derived using approximately the same framework. This idea is still under development, but we suggest a common recipe for designing the orientation tensors:

1. Construct a suitable orientation functional, corresponding to an orientation tensor.
2. Compute an adequate signal model.
3. Plug the signal model into the orientation functional and get an orientation tensor in terms of the model parameters.

2. SIGNAL MODELS

Assume that we have an N-dimensional signal $f(\mathbf{x})$. Introduce the scalar product

$$\langle f, b\rangle_{\mathrm{w}} = \int_{\mathcal{R}^N} w(\mathbf{x}) f(\mathbf{x}) b^*(\mathbf{x}) d\mathbf{x} \tag{1}$$

where $w(\mathbf{x})$ is a weight function and * means complex conjugate. Let $\{b_k(\mathbf{x})\}$ denote a subspace basis and $\{\tilde{b}_k(\mathbf{x})\}$ the corresponding dual basis, computed as

$$\tilde{b}_k(\mathbf{x}) = \sum_n \left(\mathbf{G}^{-1}\right)_{kn} b_n(\mathbf{x}) \tag{2}$$

where $\mathbf{G}_{kn} = \langle b_k, b_n\rangle_{\mathrm{w}}$. Assume that we want to approximate, or model, the signal with a linear combination of the

This work was supported by the Foundation for Strategic Research within the VISIT project, and the Knut and Alice Wallenberg foundation within the WITAS project

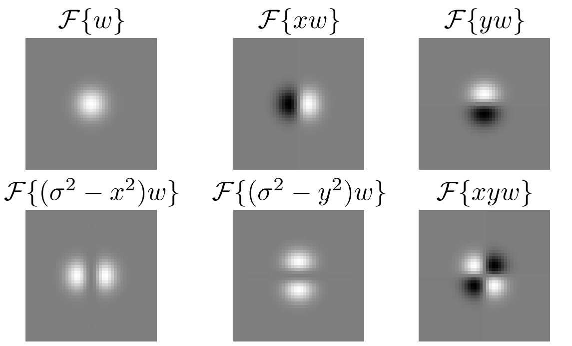

Fig. 1. Fourier transform of the filters needed to compute a second degree polynomial model of a 2D signal.

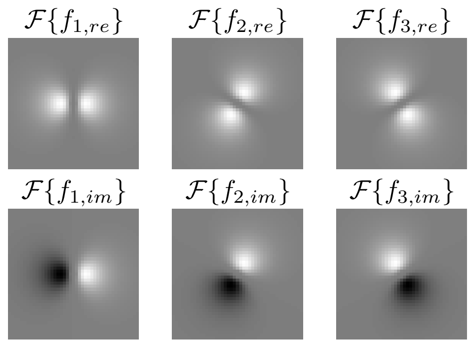

Fig. 2. Fourier transform of the filters needed to compute a quadrature filter model of a 2D signal (using the filter directions 0°, 60°, and 120°).

basis functions. This can be accomplished by minimizing a weighted least squares problem. From linear algebra we know that the solution can be written as

$$f(\mathbf{x}) \sim \sum_k \langle f, \tilde{b}_k \rangle_{\mathrm{w}} \, b_k(\mathbf{x}) = \sum_k \langle f, b_k \rangle_{\mathrm{w}} \, \tilde{b}_k(\mathbf{x}) \qquad (3)$$

Computation of the model parameters in each local region of a signal is carried out by correlating the signal with the filters $w(\mathbf{x}) b_k(\mathbf{x})$ and the resulting responses are combined according to equation 3. The weight can be seen as a spatial window controlling the size of the local region we want to model.

We will now discuss two particular signal models; the polynomial expansion model and the quadrature model.

2.1. Polynomial expansion model

In this case we have basis functions given by monomials $x_1^{p_1} x_2^{p_2} x_3^{p_3} \ldots$. We typically choose a Gaussian function with standard deviation σ as weight w. The Gaussian gives nice properties, see [2] for a thorough analysis. The standard deviation σ of the Gaussian controls the spatial size.

For an example we look at a second degree polynomial expansion of a 2D signal. Figure 1 shows an example of filters that can be used to compute the model (the basis is chosen for sake of comparison with the quadrature filters in figure 2). In particular, a second degree polynomial expansion of an N-dimensional signal f can be expressed as

$$f(\mathbf{x}) \sim \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b}^T \mathbf{x} + c \qquad (4)$$

where $\mathbf{A}$ is a symmetric $N \times N$ matrix, $\mathbf{b}$ an $N \times 1$ vector and c a scalar.

2.2. Quadrature model

Quadrature filters are defined in the Fourier domain to be zero in a half plane. For example, the filters used in [5] are expressed in the Fourier domain as $F_k(\mathbf{u}) = R(|\mathbf{u}|) D_k(\hat{\mathbf{u}})$, where

$$D_k(\mathbf{u}) = \begin{cases} (\hat{\mathbf{u}} \cdot \hat{\mathbf{n}}_k)^2 & \text{if } \hat{\mathbf{u}} \cdot \hat{\mathbf{n}}_k > 0 \\ 0 & \text{otherwise} \end{cases} \qquad (5)$$

$R(|\mathbf{u}|)$ is a radial bandpass function, usually chosen as a log-normal function $e^{-C \ln^2(|\mathbf{u}|/u_0)}$.

With these filters in mind we construct a quadrature model in the Fourier domain using the basis functions $\{D_k\}_1^K$ and the weight $w = R$. The directions $\hat{\mathbf{n}}_k$ can for example be chosen evenly distributed, e.g.

$$\hat{\mathbf{n}}_k = \begin{pmatrix} \cos \varphi_k \\ \sin \varphi_k \end{pmatrix} \quad , \quad \varphi_k = \frac{2\pi}{K}(k-1) \qquad (6)$$

Computing the scalar products $\langle F, D_k \rangle_{\mathrm{R}}$ means in practice that we correlate the signal with the filters $RD_k = F_k$, i.e. the usual quadrature filters. Let q_k denote the filter response for filter F_k. Notice that we only have to compute half of the responses if K is an even number. We assume that the signal is real valued and the filter response in a certain direction is therefore equal to the conjugated filter response in the opposite direction, i.e. $q_k = q_{k+K/2}^*$.

As an example we choose the 2D case with $K = 6$. In this case we get 3 quadrature filters. These filters are complex valued in the spatial domain and each filter therefore corresponds to 2 real valued filters, $f_k = f_{k,re} + i f_{k,im}$. $f_{k,re}$ is an even function and $f_{k,im}$ an odd function in the direction $\hat{\mathbf{n}}_k$. Figure 2 shows these 6 real valued filters in the Fourier domain for a particular choice of R. Notice that even and odd functions are matched to have the same frequency characteristic, i.e. $|\mathcal{F}\{f_{k,re}\}| = |\mathcal{F}\{f_{k,im}\}|$. After the correlations we construct the signal model using equation 3 as

$$F(\mathbf{u}) \sim \sum_k \langle F, D_k \rangle_{\mathrm{R}} \tilde{D}_k(\hat{\mathbf{u}}) = \sum_k q_k \tilde{D}_k(\hat{\mathbf{u}}) \qquad (7)$$

2.3. Comparison of models

From figures 1 and 2 we can see that the filters look qualitatively similar. The polynomial model has an extra DC-filter, but this filter is not necessary for orientation estimation since the remaining filters have zero DC (i.e. they are orthogonal to the DC-filter). As mentioned before, the even and odd quadrature filters are matched to have the same frequency characteristic. This facilitates phase invariant tasks as we will see below.

Both models can be efficiently computed, at least approximately. The polynomial filters can be made separable, which gives a low computational complexity. They can also be approximated even more efficiently using derivative filters, see [11]. The quadrature filters cannot be made Cartesian separable, but they can be approximated using filter trees, see e.g. [13].

3. LOCAL ORIENTATION TENSORS

An orientation tensor is a representation of local orientation that for N-dimensional signals takes the form of an $N \times N$ real symmetric matrix.

A *simple signal* with the orientation $\hat{\mathbf{m}}$ is defined as $f(\mathbf{x}) = h(\mathbf{x} \cdot \hat{\mathbf{m}})$ for some function h. This signal is represented by the tensor $\mathbf{T} = A\hat{\mathbf{m}}\hat{\mathbf{m}}^T$, where A is a scalar value that may encode other information than orientation, for example local signal energy. For a non-simple signal it is desired that the eigenvector with the largest eigenvalue of the tensor points out the dominant direction of the signal. A signal with no dominant direction is preferably represented by an isotropic tensor $\mathbf{T} \propto \mathbf{I}$.

There exist several methods to construct an orientation tensor in practice. Below follows a short résumé of them (in chronological order).

3.1. Gradient tensor

In [8], the gradient tensor is derived by computing a direction sensitive function closely related to the orientation functionals used in this paper. The variance, or inertia, in the direction $\hat{\mathbf{v}}$ is computed as

$$V(\hat{\mathbf{v}}) = \int_{\mathcal{R}^N} d^2(\mathbf{u}, \hat{\mathbf{v}}) |F(\mathbf{u})|^2 d\mathbf{u} \tag{8}$$

where $|F(\mathbf{u})|^2$ is interpreted as a density function and $d(\mathbf{u}, \hat{\mathbf{v}})$ is the Euclidean distance between the point $\mathbf{u}$ and the line defined by $\hat{\mathbf{v}}$. The dominant orientation is found as the direction which minimizes V. It can be shown that V can be rewritten as

$$V(\hat{\mathbf{v}}) = \hat{\mathbf{v}}^T \left(\text{trace}(\mathbf{T})\mathbf{I} - \mathbf{T}\right) \hat{\mathbf{v}} \tag{9}$$

where

$$\mathbf{T} = \int_{\mathcal{R}^N} \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T d\mathbf{x} \tag{10}$$

where ∇f is the signal gradient. The matrix $\mathbf{J} = \text{trace}(\mathbf{T})\mathbf{I} - \mathbf{T}$ is sometimes referred to as the moment of inertia matrix (in the Fourier domain). A Gaussian weight function serving as a spatial window is used in the practical implementation, i.e.

$$\mathbf{T} = \int_{\mathcal{R}^N} g(|\mathbf{x}|) \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^T d\mathbf{x} \tag{11}$$

The gradient is estimated using Gaussian derivative filters.

3.2. Quadrature tensor

This orientation tensor is thoroughly described in [5]. A number of quadrature filters with directions $\varphi_k = \frac{2\pi}{K}(k-1), k = 1, \ldots, K/2$ are correlated with the signal. The magnitude of the filter responses are then combined into an orientation tensor. The derivation of the tensor in [5] is based on the observation that if the signal is simple with the direction $\hat{\mathbf{m}}$, we get the quadrature filter response magnitudes

$$|q_k| = A(\hat{\mathbf{m}} \cdot \hat{\mathbf{n}}_k)^2 = \langle A\hat{\mathbf{m}}\hat{\mathbf{m}}^T, \hat{\mathbf{n}}_k\hat{\mathbf{n}}_k^T \rangle \tag{12}$$

where A depends on the signal energy and the bandpass function R. Hence, $|q_k|$ are scalar products between the tensor $\mathbf{T} = A\hat{\mathbf{m}}\hat{\mathbf{m}}^T$ and the basis functions $\mathbf{N}_k = \hat{\mathbf{n}}_k\hat{\mathbf{n}}_k^T$. We can therefore compute $\mathbf{T}$ using equation 3 as

$$\mathbf{T} = \sum_k |q_k| \tilde{\mathbf{N}}_k \tag{13}$$

where $\tilde{\mathbf{N}}_k$ are the dual basis vectors to $\mathbf{N}_k$. This expression is then used for non-simple signals as well.

The tensor is derived using the assumption that the signal is simple. We get a phase invariant tensor for this class of signals, due to the properties of the quadrature filters. The properties for non-simple signals are not fully investigated. For example, the tensor is not necessarily positive semidefinite for non-simple signals.

3.3. Polynomial tensor

This orientation tensor is thoroughly described in [2]. A second degree polynomial expansion is computed which gives the model in equation 4. The tensor is then constructed as

$$\mathbf{T} = \mathbf{A}\mathbf{A}^T + \gamma \mathbf{b}\mathbf{b}^T \tag{14}$$

where γ is a non-negative weight factor chosen by the user.

This tensor is designed so that the functional $\phi(\hat{\mathbf{v}}) = \hat{\mathbf{v}}^T \mathbf{T} \hat{\mathbf{v}}$ fulfills the desired properties in section 4. But a corresponding explicit functional independent of the polynomial model has not been formulated before.

4. ORIENTATION FUNCTIONALS

Here we take the view that the orientation tensor is an instance of the concept of orientation functionals. This idea was introduced in [2], and we will use it here to unify the design of orientation tensors.

An orientation functional is a mapping from the set of unit vectors to the set of non-negative real values. The value is interpreted as a measure of how well the signal locally is consistent with an orientation hypothesis in the given direction $\hat{\mathbf{v}}$. Since we do not distinguish between two opposite directions, we require that $\phi(-\hat{\mathbf{v}}) = \phi(\hat{\mathbf{v}})$. We also state some desired properties on the mapping from signal neighborhoods to the associated orientation functionals, the most important being (see [2] for a complete list):

- *Rotation equivariance*: Assume that the signal is rotated around the origin, so that $f(\mathbf{x})$ is replaced by $f(\mathbf{x}) = f(\mathbf{R}\mathbf{x})$, where $\mathbf{R}$ is a rotation matrix. Then the orientation functional ϕ associated to f should relate to ϕ by $\phi(\hat{\mathbf{v}}) = \phi(\mathbf{R}\hat{\mathbf{v}})$, i.e. be rotated in the same way.
- In directions along which the signal is constant, ϕ should be zero.
- For a simple signal in the direction $\hat{\mathbf{m}}$, ϕ should have its maximum value for $\hat{\mathbf{m}}$ and $-\hat{\mathbf{m}}$, and decrease monotonically as the angle to the closer of these two directions increases.
- The orientation functional should be invariant to the DC level.

The orientation functional can be constructed in the Fourier domain or in the spatial domain. An example of an orientation functional in the Fourier domain is

$$\phi(\hat{\mathbf{v}}) = \int_{\mathcal{R}^N} B(|\mathbf{u}|)\,|F(\mathbf{u})|^p(\hat{\mathbf{u}}\cdot\hat{\mathbf{v}})^2\,d\mathbf{u} \qquad (15)$$

where B is a radial function and p is an energy sensitivity parameter. The last factor is a directional sensitivity function, decreasing as a $\cos^2$ from the preferred direction. This particular choice makes it possible to rewrite the functional as

$$\phi(\hat{\mathbf{v}}) = \hat{\mathbf{v}}^T\mathbf{T}\hat{\mathbf{v}} \qquad (16)$$

where

$$\mathbf{T} = \int_{\mathcal{R}^N} B(|\mathbf{u}|)\,|F(\mathbf{u})|^p\hat{\mathbf{u}}\hat{\mathbf{u}}^T\,d\mathbf{u} \qquad (17)$$

In particular, if we choose $B(|\mathbf{u}|) = |\mathbf{u}|^2$ and $p = 2$ we get

$$\phi(\hat{\mathbf{v}}) = \int_{\mathcal{R}^N} |F(\mathbf{u})|^2(\mathbf{u}\cdot\hat{\mathbf{v}})^2 d\mathbf{u} \qquad (18)$$

This functional can be rewritten in the spatial domain using Parseval's relation as

$$\phi(\hat{\mathbf{v}}) = \int_{\mathcal{R}^N} \left(\nabla f(\mathbf{x})\cdot\hat{\mathbf{v}}\right)^2 d\mathbf{x} \qquad (19)$$

By adding a weight function, serving as a spatial window, we arrive at

$$\phi(\hat{\mathbf{v}}) = \int_{\mathcal{R}^N} w(|\mathbf{x}|)\left(\nabla f(\mathbf{x})\cdot\hat{\mathbf{v}}\right)^2 d\mathbf{x} = \hat{\mathbf{v}}^T\mathbf{T}\hat{\mathbf{v}} \qquad (20)$$

where

$$\mathbf{T} = \int_{\mathcal{R}^N} w(|\mathbf{x}|)\nabla f(\mathbf{x})\nabla f(\mathbf{x})^T d\mathbf{x} \qquad (21)$$

This is the same tensor expression as in equation 11.

5. DERIVATION OF THE TENSORS IN SECTION 3

We will now show that the orientation tensors described in section 3 can be derived by choosing an orientation functional and replacing the signal with a suitable model.

5.1. Gradient tensor

The tensor in section 3.1 is equivalent to the one in equation 21 when choosing w as a Gaussian function. In this case we do not insert an explicit model of the signal, except that the gradient ∇f is "modeled" by Gaussian derivative filters.

5.2. Quadrature tensor

The quadrature tensor in section 15 can be derived from the functional in equation 15 using $B = R$ and $p = 1$. But in this case we *model the magnitude* $|F(\mathbf{u})|$ instead of $F(\mathbf{u})$. To compute a model of $|F(\mathbf{u})|$ is a bit difficult in practice. It turns out though that this model can, up to a constant, be computed from the model of $F(\mathbf{u})$ if the signal is simple. Assume that we have computed a quadrature model of the signal $F(\mathbf{u})$, equation 7. Then it can be shown, assuming a simple signal, that the quadrature model of $|F(\mathbf{u})|$ can be expressed as

$$\begin{aligned} |F(\mathbf{u})| &\sim C\sum_{k=1}^{K}|q_k|\tilde{D}_k \\ &= \sum_{k=1}^{K/2}|q_k|(\tilde{D}_k(\mathbf{u}) + \tilde{D}_{k+K/2}(\mathbf{u})) \end{aligned} \qquad (22)$$

where C is a positive scalar which equals 1 for a narrow-banded signal. The last equality holds because $q_k = q^*_{k+K/2}$. This model (ignoring C) inserted into equation 17 gives

$$\begin{aligned} \mathbf{T} &= \int_{\mathcal{R}^N} R(|\mathbf{u}|)|F(\mathbf{u})|\hat{\mathbf{u}}\hat{\mathbf{u}}^T\,d\mathbf{u} \\ &\sim \sum_{k=1}^{K/2}|q_k|\int_{\mathcal{R}^N} R(|\mathbf{u}|)\left(\tilde{D}_k(\mathbf{u}) + \tilde{D}_{k+K/2}(\mathbf{u})\right)\hat{\mathbf{u}}\hat{\mathbf{u}}^T\,d\mathbf{u} \end{aligned} \qquad (23)$$

It remains to be shown that the integrals

$$\mathbf{M}_k = \int_{\mathcal{R}^N} R(|\mathbf{u}|)\left(\tilde{D}_k(\mathbf{u}) + \tilde{D}_{k+K/2}(\mathbf{u})\right)\hat{\mathbf{u}}\hat{\mathbf{u}}^T\,d\mathbf{u} \qquad (24)$$

are proportional to the dual vectors $\tilde{\mathbf{N}}_k$. This can be realized by studying the scalar product between $\mathbf{M}_k$ and $\mathbf{N}_p$ for $k, p \in [1, K/2]$, i.e.

$$\begin{aligned}
\langle \mathbf{N}_p, \mathbf{M}_k \rangle &= \hat{\mathbf{n}}_p^T \mathbf{M}_k \hat{\mathbf{n}}_p \\
&= \int_{\mathcal{R}^N} R(|\mathbf{u}|) \left(\tilde{D}_k(\mathbf{u}) + \tilde{D}_{k+K/2}(\mathbf{u}) \right) (\hat{\mathbf{u}} \cdot \hat{\mathbf{n}}_p)^2 \, d\mathbf{u} \\
&= \langle \tilde{D}_k, D_p \rangle_{\mathrm{R}} + \langle \tilde{D}_k, D_{p+K/2} \rangle_{\mathrm{R}} \\
&\quad + \langle \tilde{D}_{k+K/2}, D_p \rangle_{\mathrm{R}} + \langle \tilde{D}_{k+K/2}, D_{p+K/2} \rangle_{\mathrm{R}} \\
&= \delta(k-p) + \delta(k-p-K/2) \\
&\quad + \delta(k+K/2-p) + \delta(k-p) \\
&= 2\delta(k-p)
\end{aligned} \tag{25}$$

where δ is the discrete Dirac function. Hence, $\mathbf{M}_k$ fulfills the criteria for being a dual basis (except for a constant).

5.3. Polynomial tensor

From the polynomial model in equation 4 we can compute the signal gradient as

$$\nabla f(\mathbf{x}) \sim 2\mathbf{A}\mathbf{x} + \mathbf{b} \tag{26}$$

If we insert this estimate into the tensor expression in equation 21 we get

$$\begin{aligned}
\mathbf{T} &= \int_{\mathcal{R}^N} g(|\mathbf{x}|) \nabla f \nabla f^T d\mathbf{x} \\
&\sim \int_{\mathcal{R}^N} g(|\mathbf{x}|)(2\mathbf{A}\mathbf{x} + \mathbf{b})(2\mathbf{A}\mathbf{x} + \mathbf{b})^T d\mathbf{x} \\
&= \ldots = 4\sigma^2 \mathbf{A}\mathbf{A}^T + \mathbf{b}\mathbf{b}^T
\end{aligned} \tag{27}$$

This tensor is proportional to the tensor in section 3.3 if we choose $\gamma = \frac{1}{4\sigma^2}$.

6. CONCLUSIONS AND DISCUSSION

All three tensors in section 3 have their advantages and disadvantages, and there may not be a best choice for all applications. This framework might help to understand the properties of the different tensors and may also be of help in the design of new tensors using other signal models.

Some observations can be made from the analysis. Note for example that the gradient tensor and the polynomial tensor are derived using the same functional but different signal models. We can also see that the quadrature tensor is not necessarily positive semidefinite for non-simple signals, since the model of $|F(\mathbf{u})|$ can assume negative values. It is also not clear how this model behaves for non-simple signals. Finally, if we ignore the weight-functions we can roughly compare the tensors in the Fourier domain and say that the gradient tensor and the polynomial tensor are both based on the "density" function $|\mathbf{u}|^2 |F(\mathbf{u})|^2$, while the quadrature tensor is based on $|F(\mathbf{u})|$.

7. REFERENCES

[1] J. Bigün and G. H. Granlund, "Optical Flow Based on the Inertia Matrix of the Frequency Domain.," in *Proceedings of SSAB'88*, Lund University, Sweden, March 1988,

[2] G. Farnebäck, "Spatial Domain Methods for Orientation and Velocity Estimation," Lic. Thesis LiU-Tek-Lic-1999:13, Dept. EE, Linköping University, SE-581 83 Linköping, Sweden, March 1999,

[3] G. Farnebäck, "Very High Accuracy Velocity Estimation using Orientation Tensors, Parametric Motion, and Simultaneous Segmentation of the Motion Field," in *Proceedings of ICCV'01*, Vancouver, Canada, July 2001, vol. I,

[4] B. Jähne, H. Haussecker, and P. Geissler, Eds., *Handbook of Computer Vision and Applications*, Academic Press, 1999,

[5] G. H. Granlund and H. Knutsson, *Signal Processing for Computer Vision*, Kluwer Academic Publishers, 1995,

[6] C.G. Harris and M. Stephens, "A combined corner and edge detector," in *4th Alvey Vision Conference*, September 1988,

[7] W. Förstner, "A framework for low level feature extraction," in *Proc. of ECCV'94*, Stockholm, Sweden, May 1994, vol. II,

[8] J. Bigün, *Local Symmetry Features in Image Processing.*, Ph.D. thesis, Linköping University, Sweden, 1988,

[9] J. Bigün and G. H. Granlund, "Optimal Orientation Detection of Linear Symmetry," in *Proceedings of ICCV'87*, London, Great Britain, June 1987,

[10] G. Farnebäck, "Disparity Estimation from Local Polynomial Expansion," in *Proceedings of SSAB'01*, Norrköping, March 2001, SSAB,

[11] B. Johansson, "Multiscale Curvature Detection in Computer Vision," Lic. Thesis LiU-Tek-Lic-2001:14, Dept. EE, Linköping University, SE-581 83 Linköping, Sweden, March 2001,

[12] R. M. Haralick, "Digital step edges from zero crossing of second directional derivatives," vol. PAMI-6, no. 1, pp. 58–68,

[13] M. Andersson, J. Wiklund, and H. Knutsson, "Filter Networks," in *Proceedings of Signal and Image Processing (SIP'99)*, Nassau, Bahamas, October 1999, IASTED,