Least Squares and Robust Estimation of Local Image Structure

Joost van de Weijer and Rein van den Boomgaard

Intelligent Sensory Information Systems
Computer Science Department
University of Amsterdam
The Netherlands
{joostw,rein}@science.uva.nl

Abstract. Linear scale space methodology uses Gaussian probes at scale s to observe the differential structure. In observing the differential image structure through the Gaussian derivative probes at scale s we implicitly construct the Taylor series expansion of the smoothed image. The Gaussian facet model, as a generalization of the classic Haralick facet model, constructs a polynomial approximation of the unsmoothed image. The measured differential structure therefore is closer to the 'real' structure than the differential structure measured using Gaussian derivatives.
At the points in an image where the differential structure changes abruptly (because of discontinuities in the imaging conditions, e.g. a material change, or a depth discontinuity) both the Gaussian derivatives and the Gaussian facet model diffuse the information from both sides of the discontinuity (smoothing across the edge).
Robust estimators that are classically meant to deal with statistical outliers can also be used to deal with these 'mixed model distributions'. In this paper we introduce the robust estimators of local image structure. Starting with the Gaussian facet model where we replace the quadratic error norm with a robust (Gaussian) error norm leads to a robust Gaussian facet model.
We will show examples of using the robust differential structure estimators for luminance and color images, for zero and higher order differential structure. Furthermore we look at a 'robustified' structure tensor that forms the basis of robust orientation estimation.

1 Introduction

Linear scale-space theory of vision not only refers to the introduction of an explicit scale-parameter, it also refers to the use of differential operators to study the local structure of images. The classical way to observe the local differential image structure is to consider all Gaussian derivatives at scale s up to order N. Basically what we do is construct the Taylor series expansion of the *smoothed* image (i.e. the image observed at scale s). The Taylor polynomial thus is an approximation of the smoothed image and not of the original image.

Instead of constructing a polynomial local model of the smoothed image we can equally well construct a polynomial approximation of the unsmoothed image. Our starting point is the *image facet* model as introduced by Haralick et. al. [1]. His facet model takes a polynomial function and fits it to the data observed in a small neighborhood in the image using a linear least squares estimation procedure. The image derivatives then can be calculated as the derivatives of the fitted analytical function.

Farnebäck [2] generalizes the Haralick facet model to incorporate spatial weights in order to express the relative importance of the image samples in estimating the parameters of the polynomial function. In the classic Haralick facet model all points in the local neighborhood are considered equally important.

For spatial weighting the choice of the Gaussian kernel leads to a specially efficient implementation. Due to the fact that the derivatives of the Gaussian function are given by a polynomial (determined by the order of differentiation) times the Gaussian function itself, the coefficients in the polynomial function turn out to be a linear combination of the Gaussian derivatives.

The least squares estimation procedure considers all points in a local neighborhood, even in the situation where the local neighborhood is on the boundary of two regions in an image. The regions on either side of the boundary may well be approximated with a low-order polynomial model. The regions can be so different that their union cannot be accurately described using the same low order polynomial model. The estimation procedure then compromises between the two regions: the edge will be smoothed.

In Section 2 we generalize the Gaussian facet model to deal with those multi-model situations. Instead of using a linear least squares estimation procedure we will use a robust estimation technique. A robust estimation technique will only consider the data points from one of the regions and will disregard the data from the other region as being statistical outliers. Robust estimation of local image structure is pioneered by Besl [3]. Our work (see also [4]) differs from the work of Besl in that we consider Gaussian aperture instead of 'crisp' neighborhoods in which the polynomial function is fitted. Furthermore we introduce a fixed point iteration procedure to find the robust estimate.

In Section 3 we present a generalization of earlier work [4–6]. We derive iterative robust estimators of local image structure and we will give some examples ranging from a simple zero order Gaussian facet model to a first order facet model for color images.

In Section 4 we describe a robust estimator for a derived image quantity: the local orientation (see also [6]). To that end we consider the often used orientation estimator based on a eigen analysis of the structure tensor. Robust estimation of the orientation turns out to be quite similar, the structure tensor is replaced with a 'robustified' version in which only the points are considered that closely fit the model (i.e. the points that are not outliers).

2 Least Squares Estimation of Local Image Structure

Locally around a point $\mathbf{x}$ the image function f can be approximated with a linear combination of basis functions ϕ_i, $i = 1, \ldots, K$:

$$\hat{f} = a_1\phi_1 + \cdots + a_K\phi_K \tag{1}$$

We can rewrite this as $\hat{f} = \boldsymbol{\Phi}\mathbf{a}$ where $\boldsymbol{\Phi} = (\phi_1\, \phi_2 \cdots \phi_K)$ and $\mathbf{a} = (a_1\, a_2 \cdots a_K)^\mathsf{T}$. The least squares estimator minimizes the difference ϵ of the image f and the approximation $\hat{f}$:

$$\epsilon(\mathbf{x}) = \int_{\mathbb{R}^d} \Big(f(\mathbf{x}+\mathbf{y}) - \hat{f}(\mathbf{y})\Big)^2 W(\mathbf{y}) d\mathbf{y} \tag{2}$$

where W is the aperture function defining the locality of the model fitting. Note that the optimal fitting function $\hat{f}$ differs from position to position in the image plane. We thus have that $\hat{f}(\mathbf{y}) = \boldsymbol{\Phi}(\mathbf{y})\mathbf{a}(\mathbf{x})$, i.e. $\hat{f}(\mathbf{y}) = a_1(\mathbf{x})\phi_1(\mathbf{y}) + \cdots + a_K(\mathbf{x})\phi_K(\mathbf{y})$.

The optimal parameter vector $\mathbf{a}$ is found by projecting the function f onto the subspace spanned by the basis functions in $\boldsymbol{\Phi}$. In this function space the inner product is given by:

$$f^\mathsf{T} g \equiv \langle f, g\rangle_W = \int_{\mathbb{R}^d} f(\mathbf{x})\, g(\mathbf{x})\, W(\mathbf{x})\, d\mathbf{x} \tag{3}$$

The inner product of functions f and g will also be denoted as $f^\mathsf{T} g$.

To derive the optimal parameter vector $\mathbf{a}$ we take the derivative of the error ϵ with respect to the parameter vector $\mathbf{a}$, set it equal to zero and solve for $\mathbf{a}$. Writing ϵ in terms of the inner product results in

$$\epsilon(\mathbf{x}) = (f_{-\mathbf{x}} - \boldsymbol{\Phi}\mathbf{a})^\mathsf{T}(f_{-\mathbf{x}} - \boldsymbol{\Phi}\mathbf{a}) \tag{4}$$

where $f_{-\mathbf{x}}(\mathbf{y}) = f(\mathbf{x}+\mathbf{y})$ is the translated image $f_{-\mathbf{x}}(\mathbf{y}) = f(\mathbf{x}+\mathbf{y})$. The integral is now 'hidden' in the inner product of two functions. This can be rewritten as:

$$\epsilon(\mathbf{x}) = f_{-\mathbf{x}}^\mathsf{T} f_{-\mathbf{x}} - 2\mathbf{a}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T} f_{-\mathbf{x}} + \mathbf{a}^\mathsf{T}\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Phi}\mathbf{a} \tag{5}$$

Taking the derivative of ϵ with respect to $\mathbf{a}$ and setting this equal to 0 and solving for $\mathbf{a}$ we obtain:

$$\mathbf{a} = (\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^\mathsf{T} f_{-\mathbf{x}} = \tilde{\boldsymbol{\Phi}}^\mathsf{T} f_{-\mathbf{x}} \tag{6}$$

where $\tilde{\boldsymbol{\Phi}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}^\mathsf{T}\boldsymbol{\Phi})^{-1}$ is the *dual basis*. The functions in the dual basis, $\tilde{\boldsymbol{\Phi}} = (\,\tilde{\phi}_1 \cdots \tilde{\phi}_K\,)$, are the functions such that the inner product $\tilde{\phi}_i^\mathsf{T} f_{-\mathbf{x}}$ equals the coefficient a_i in the approximation $\hat{f} = a_1\phi_1 + \cdots + a_K\phi_K$. The dual basis functions, multiplied with the aperture function, thus are the correlation kernels needed to calculate the coefficients in the polynomial image approximation.

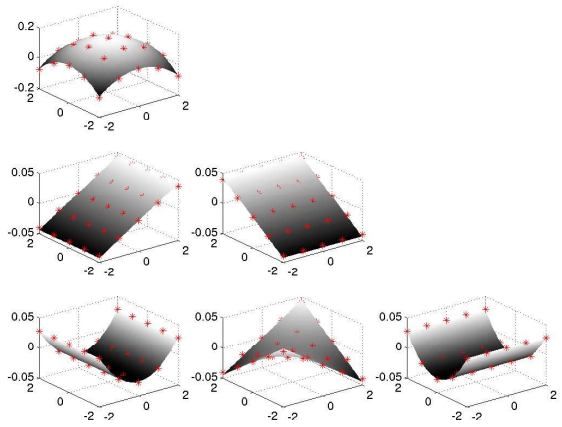

Fig. 1. The Haralick Facet Model. From left to right, top to bottom the dual basis functions are plotted. The shaded functions are the dual basis functions within a 2nd order facet model, the (red) stars correspond with the discrete dual functions. The neighborhood was taken to be of size 5×5. The scale s for the analytical kernel was set at $s = 2.42$. This value is the value to make the difference between the discrete and analytical facet models minimal. For larger neighborhoods $N \times N$ the correspondence becomes better and the analytical scale approaches $N/2$.

The classic Haralick facet model uses a uniform weight function $W(\mathbf{x}) = 1$ for $\|\mathbf{x}\|_\infty \leq s$ and $W(\mathbf{x}) = 0$ elsewhere, i.e. a 'crisp' neighborhood within an axis aligned square of size $2s \times 2s$.

For the second order polynomial basis:

$$\Phi = \left(1, x, y, \tfrac{1}{2}x^2, xy, \tfrac{1}{2}y^2\right) \tag{7}$$

the dual basis is

$$\tilde{\Phi} = \left(\tfrac{7}{8\,s^2} - \tfrac{15\,x^2}{16\,s^4} - \tfrac{15\,y^2}{16\,s^4}, \tfrac{3\,x}{4\,s^4}, \tfrac{3\,y}{4\,s^4}, \tfrac{-15}{8\,s^4} + \tfrac{45\,x^2}{8\,s^6}, \tfrac{9\,x\,y}{4\,s^6}, \tfrac{-15}{8\,s^4} + \tfrac{45\,y^2}{8\,s^6}\right) \tag{8}$$

The dual basis functions are depicted in Fig. 1. The first dual basis function (multiplied with the aperture function) is the correlation kernel needed to calculate the coefficient of the constant basis function in the approximation of the local image patch. Observe that in the Haralick facet model, the first dual basis function is not everywhere positive. Fig. 1 also shows the discrete dual basis functions, these follow from a formulation of the facet model in a discrete image space as can be found in the work of Haralick.

Within a scale-space context the most natural choice is to start with a polynomial basis and a Gaussian aperture function $W = G^s$ where G^s is the Gaussian function at scale s. Again starting with the second order polynomial basis the dual basis is a different one due to the difference in the inner product (as a consequence of a different aperture function):

$$\tilde{\Phi} = \left(2 - \tfrac{x^2}{2\,s^2} - \tfrac{y^2}{2\,s^2}, \tfrac{x}{s^2}, \tfrac{y}{s^2}, -s^{-2} + \tfrac{x^2}{s^4}, \tfrac{x\,y}{s^4}, -s^{-2} + \tfrac{y^2}{s^4}\right) \tag{9}$$

Again, a dual basis function, multiplied with the—Gaussian—aperture function is the correlation kernel needed to calculate the corresponding coefficient in the polynomial approximation of the local image patch. For the zero order coefficient the correlation kernel is a Gaussian function multiplied with a parabola: $(2 - \frac{x^2}{2\,s^2} - \frac{y^2}{2\,s^2})\,G^s(x,y)$. Again we see that the zero order coefficient in the polynomial image approximation requires a kernel with negative values.

The derivatives of the Gaussian function are equal to a polynomial function (a Hermite polynomial depending on the derivative taken) times the Gaussian function, we may write the correlation kernels associated with the dual basis functions in the Gaussian facet model as a linear combination of Gaussian derivatives. It is not hard to prove that the zero order coefficient in the second order Gaussian facet model is found by convolving the image f with the kernel:

$$G^s - \tfrac{1}{2}s^2\left(G^s_{xx} + G^s_{yy}\right) \tag{10}$$

Now we easily recognize where the negative values in the kernel come from. The term G^s is the Gaussian scale-space smoothing term. The term $-\frac{1}{2}s^2\left(G^s_{xx} + G^s_{yy}\right)$ is a well-known sharpening term: subtracting the Laplacian from the smoothed image, sharpens the image. The sharpening term is due to the fact that the Gaussian facet model approximates the original image, not the smoothed image.

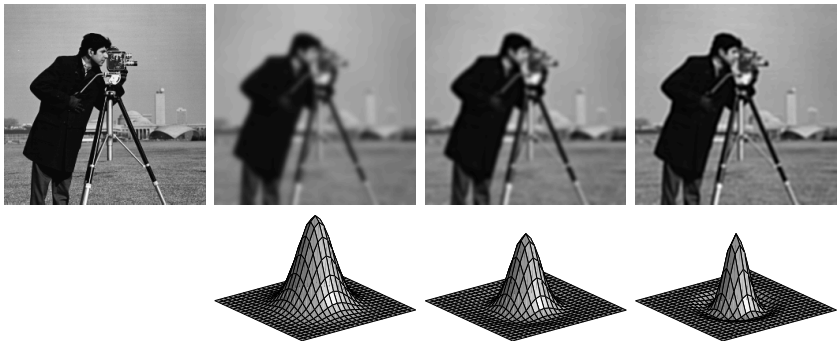

Fig. 2. Zero-order coefficient in the Gaussian Facet Model. On the first row, from left to right: the original image, and the zero order coefficients in the Gaussian facet model of order 0,2 and 6. On the second row the convolution kernel is shown that, convoluted with the original image, results in the image above it.

It turns out that this observation is true for higher order facet models as well. For a 4th order Gaussian facet model, the kernel to calculate the zero order coefficient is:

$$G^s - \tfrac{1}{2}s^2\left(G^s_{xx} + G^s_{yy}\right) + \tfrac{1}{8}s^4\left(G^s_{xxxx} + 2\,G^s_{xxyy} + G^s_{yyyy}\right) \tag{11}$$

In Fig. 2 the kernels to calculate the zero order coefficient in the Gaussian facet model of orders 0, 2 and 6 are depicted together with the convoluted images.

Apparently the N-jet of an image observed at scale s encodes details of size less then s, i.e. from the N-jet observed at scale s a lot of detail can be reconstructed.

3 Robust Estimation of Local Image Structure

Consider again the error of the Gaussian weighted least squares approximation:

$$\epsilon(\mathbf{x}) = \int_{\mathbb{R}^d} \left(f(\mathbf{x}+\mathbf{y}) - \hat{f}(\mathbf{y}) \right)^2 G^s(\mathbf{y}) d\mathbf{y} \tag{12}$$

It is well known that this error definition is not well suited for those situations were we have outliers in our measurements. In the image processing context statistical outliers are not so frequently occurring. The effect that makes least squares estimates questionable is that when collecting measurements from a neighborhood in an image these are often not well modeled using a simple (facet) model. For instance we may model local image luminance quite well with a second order polynomial model *but not near edges* where we switch from one model instantiation to another. Such multi-model situations are abundant in computer vision applications and are most often due to the nature of the imaging process where we see abrupt changes going from one object to another object.

Multi-modelity can be incorporated into sophisticated estimation procedures where we not only estimate (multi-)model parameters but also the geometry that separates the different regions (one for each model). One of the oldest examples is perhaps Hueckels edge detector [7] in which a local image patch is described with two regions separated by a straight boundary. The detector estimates this boundary and the parameters of the luminance distributions on each side of the edge.

In this paper we take a less principled approach. Instead of a multi-model approach we stick to a simpler one-model approach where we use a statistical *robust estimator* that allows us to consider part of the measurements from the local neighborhood to belong to the model we are interested in and disregard all other measurements as being 'outliers' and therefore not relevant in estimating the model parameters.

The crux of a robust estimation procedure is to rewrite the above error measure as:

$$\epsilon(\mathbf{x}) = \int_{\mathbb{R}^d} \rho(f(\mathbf{x}+\mathbf{y}) - \hat{f}(\mathbf{y}))\, G^s(\mathbf{y}) d\mathbf{y} \tag{13}$$

where ρ is the error norm. The choice $\rho(e) = e^2$ leads to the least squares estimator. Evidently measurements that are outliers to the 'true' model are weighted heavily in the total error measure. Reducing the influence of the large errors leads to *robust error norms*.

Writing $f_{-\mathbf{x}}(\mathbf{y}) = f(\mathbf{x}+\mathbf{y})$ and using the local linear model $\hat{f}(\mathbf{y}) = \Phi(\mathbf{y})\mathbf{a}(\mathbf{x})$ we obtain:

$$\epsilon(\mathbf{x}) = \int_{\mathbb{R}^d} \rho(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))\, G^s\, d\mathbf{y} \tag{14}$$

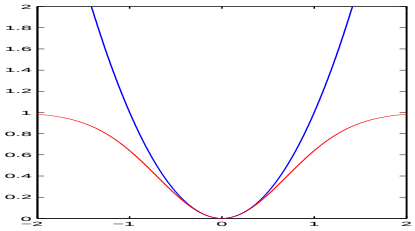

Fig. 3. Quadratic versus (robust) Gaussian error norm. The Gaussian error norm is of 'scale' $m = 0.7$.

We omitted the spatial argument $\mathbf{y}$ for ease of notation. In this paper the 'Gaussian error norm' is chosen:

$$\rho(e) = 1 - \exp\left(-\frac{e^2}{2m^2}\right) \tag{15}$$

The scale m in the error norm will be called the *model scale* to contrast it with the spatial scale s that is used in the spatial aperture function G^s. In Fig. 3 the error norm is sketched. Compared to the quadratic error norm this norm is 'clamped' at value 1. For $e \gg m$ the exact value of the error is not important any more. Gross outliers are therefore not given the weight to influence the estimation greatly.

The optimal model parameters are found by calculating the derivative of the error measure and setting this equal to zero:

$$\frac{\partial \epsilon}{\partial \mathbf{a}} = \frac{\partial}{\partial \mathbf{a}} \int_{\mathbb{R}^d} \rho(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))\, G^s \, d\mathbf{y} \tag{16}$$

$$= \frac{\partial}{\partial \mathbf{a}} \int_{\mathbb{R}^d} \left(1 - \exp\left(-\frac{(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))^2}{2\,m^2}\right)\right) G^S \, d\mathbf{y} \tag{17}$$

$$= -\frac{1}{m} \int_{\mathbb{R}^d} (f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))\, \Phi \exp\left(-\frac{(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))^2}{2\,m^2}\right) G^s \, d\mathbf{y} \tag{18}$$

Setting this derivative equal to zero and rewriting terms we obtain:

$$\int_{\mathbb{R}^d} f_{-\mathbf{x}}\, \Phi \exp\left(-\frac{(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))^2}{2\,m^2}\right) G^s \, d\mathbf{y} = \int_{\mathbb{R}^d} \Phi\mathbf{a}(\mathbf{x})\, \Phi \exp\left(-\frac{(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x}))^2}{2\,m^2}\right) G^s \, d\mathbf{y} \tag{19}$$

This can be rewritten as:

$$\int_{\mathbb{R}^d} f_{-\mathbf{x}}\, \Phi G^m \left(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x})\right) G^s \, d\mathbf{y} = \int_{\mathbb{R}^d} \Phi\mathbf{a}(\mathbf{x})\, \Phi G^m \left(f_{-\mathbf{x}} - \Phi\mathbf{a}(\mathbf{x})\right) G^s \, d\mathbf{y} \tag{20}$$

where G^m is the Gaussian function at scale m. This Gaussian function weighs the model distance, whereas the Gaussian function G^s weighs the spatial distance.

We define the operator Γ:

$$(\Gamma^m g)(\mathbf{y}) = G^m(f_{-\mathbf{x}}(\mathbf{y}) - \Phi(\mathbf{y})\mathbf{a}(\mathbf{x}))\, g(\mathbf{y}) \tag{21}$$

i.e. the point wise multiplication of the function g with the model weight function. Now Γ^m acts as a diagonal (matrix) operator in the function space. Using the vectorial notation of the inner product we can write:

$$\Phi^{\mathsf{T}} \Gamma^m f_{-\mathbf{x}} = \Phi^{\mathsf{T}} \Gamma^m \Phi\, \mathbf{a} \tag{22}$$

This looks like a familiar weighted linear least squares equation that can be solved for the value of $\mathbf{a}$. It is not, because Γ^m is dependent on $\mathbf{a}$. Solving for $\mathbf{a}$ can be done using an *iterated weighted least squares* procedure:

$$\mathbf{a}^{i+1} = \left(\Phi^{\mathsf{T}} \Gamma(\mathbf{a}^i) \Phi\right)^{-1} \Phi^{\mathsf{T}} \Gamma(\mathbf{a}^i) f_{-\mathbf{x}} \tag{23}$$

Some examples of these robust estimators may clarify matters. In the next subsection we consider the most simple of all local structure models: a locally constant model. The resulting image operator turns out to be an iterated version of the bilateral filter introduced by Tomasi and Manduchi [8].

3.1 Zero-order Image Structure

Consider a locally constant image model with only one basis function:

$$\Phi = (1) \tag{24}$$

i.e. the constant function. Eq.(23) then reduces to:

$$a_0^{i+1}(\mathbf{x}) = \frac{\int_{\mathbb{R}^d} f(\mathbf{x}+\mathbf{y})\, G^m(f(\mathbf{x}+\mathbf{y}) - a_0^i(\mathbf{x}))\, G^s(\mathbf{y})\, d\mathbf{y}}{\int_{\mathbb{R}^d} G^m(f(\mathbf{x}+\mathbf{y}) - a_0^i(\mathbf{x}))\, G^s(\mathbf{y})\, d\mathbf{y}} \tag{25}$$

This is an iterated version of the bilateral filter as introduced by Tomasi and Manduchi [8]. It is also related to the filters introduced by Smith et. al. [9]. The bilateral filter thus implements one iteration of a robust estimator with initial value $a_0^0 = f$.

In previous papers [4, 5] we have analyzed robust estimation of the zero order local image structure. Some observations made are:

- The robust estimator finds the local mode in the local luminance histogram which is smoothed with a Gaussian kernel of scale m. The local mode that is found is the local maximum in the smoothed histogram that is closest to the initial value.
- Bilateral filtering implements one iteration of the robust estimator. From mean shift analysis we know that the first step in a mean shift algorithm is a large one in the direction of the optimal value. This explains the impressive results on the bilateral filter in reducing the noise while preserving the structure of images.

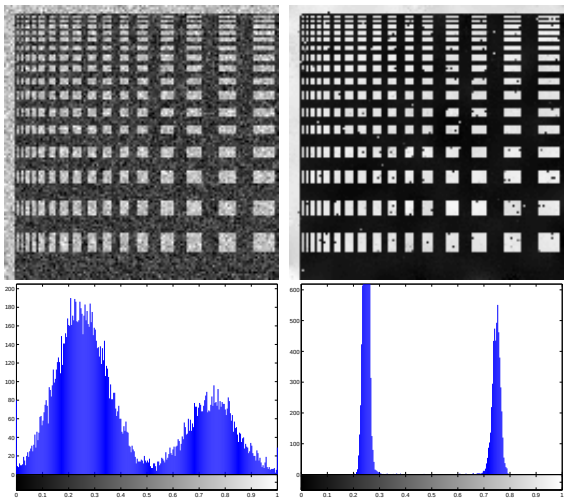

Fig. 4. Robust Estimation of Local Image Structure. On the first row a test image with noise added on the left and the result of the robust estimator based on a zero-order facet model. On the second row the histograms of the images above are depicted. Observe that the robust estimator is capable of finding the modes of both the distributions.

- The choice of an initial estimate is very important. We have found good results using the result of a linear least squares estimate as the initial estimate. In certain situations however the amount of smoothing induced by the least squares estimator sets the robust estimator at a wrong starting point leading to a local maximum in the histogram that does not correspond with the structure that we are interested in. This situation is often occurring in case the area of the structure of interest is less then the area of the 'background' (e.g. document images where there is more paper then ink visible). In such cases the image itself can be used as an initial estimate of the zero order local structure.

- The results of robust estimation of local image structure bear great resemblance to the results of non-linear diffusion. The theoretical link between robust estimation and non-linear diffusion techniques has been reported before (see [10]). The main difference with the robust estimator technique described here is that in each iteration of a non-linear diffusion algorithm the image data resulting from the previous iteration is used. In the robust estimator described here we stick to the original image data and only update the parameter to be estimated. Fig. 5 shows the differences between these two procedures.

Fig. 5. Robust Estimation and Non-linear diffusion. On the left the original image of a flower. In the middle the robust estimation of the zero order local structure and on the right the result of iteratively applying one iteration of the robust estimator, each time using the image data from the previous iteration (this procedure is very much like a non-linear diffusion process).

3.2 Higher-order Image Structure

For the image in Fig. 4 the assumption of local constant image model is a correct assumption, for most natural images such a model is an oversimplification though. Then it is better to use a higher order model for the local image structure. We start with a simple first order model for 1D functions. The local basis is:

$$\boldsymbol{\Phi} = \begin{pmatrix} 1 & x \end{pmatrix} \tag{26}$$

This leads to the matrix $\boldsymbol{\Phi}^{\mathsf{T}} \boldsymbol{\Gamma}^m \boldsymbol{\Phi}$:

$$\begin{pmatrix} \int_{\mathbb{R}} G^m(f(x+y) - a_0^i - a_1^i y) G^s(y) dy & \int_{\mathbb{R}} y\, G^m(f(x+y) - a_0^i - a_1^i y)\, G^s(y)\, dy \\ \int_{\mathbb{R}} y\, G^m(f(x+y) - a_0^i - a_1^i y)\, G^s(y)\, dy & \int_{\mathbb{R}} y^2\, G^m(f(x+y) - a_0^i - a_1^i y)\, G^s(y)\, dy \end{pmatrix} \tag{27}$$

and vector $\boldsymbol{\Phi}^{\mathsf{T}} \boldsymbol{\Gamma}^m f_{-x}$:

$$\begin{pmatrix} \int_{\mathbb{R}} f(x+y)\, G^m(f(x+y) - a_0^i - a_1^i y)\, G^s(y)\, dy \\ \int_{\mathbb{R}} y\, f(x+y)\, G^m(f(x+y) - a_0^i - a_1^i y)\, G^s(y)\, dy \end{pmatrix} \tag{28}$$

The robust estimator of the local linear model is given by Eq.(23). Fig. 6 shows a univariate 'saw-tooth' signal corrupted with additive noise. Also shown are the robust estimates based on a zero order facet model and the robust estimate based on a first order facet model. It is obvious that a robust estimator based on a local constant model is not capable of reconstructing the saw tooth signal from the noisy observations. Using a local first order model leads to a far better reconstruction.

The first order robust facet model is easily generalized to 2D functions:

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_{(00)} & \phi_{(10)} & \phi_{(01)} \end{pmatrix} \tag{29}$$

$$= \begin{pmatrix} 1 & x_1 & x_2 \end{pmatrix} \tag{30}$$

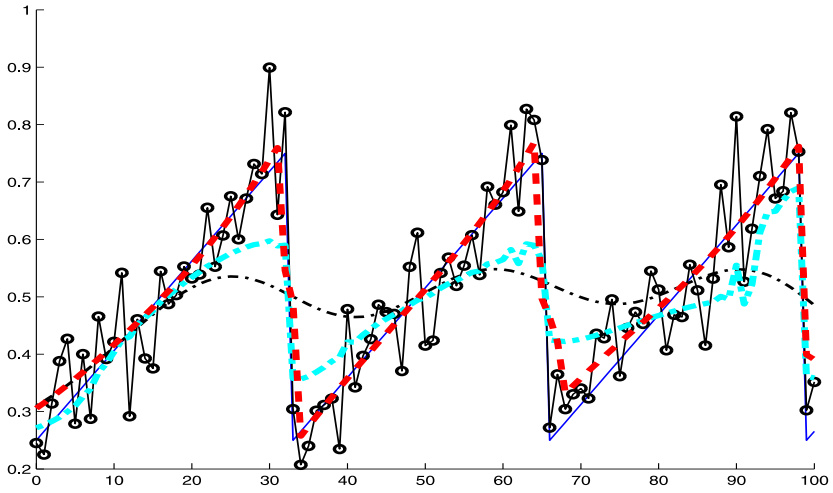

Fig. 6. Robust Estimation of Local Structure in 1D functions. A 'sawtooth' function with added noise is shown together with the Gaussian linear least squares estimate, i.e. the Gaussian smoothing (the thin 'sinusoidal' line), the robust estimate based on a zero order facet model (the dashed-dotted line) and the robust estimate based on a first order model (the thick dashed line). The spatial scale is 9 and the tonal (model) scale is 0.1. The number of iterations used is 10.

This leads to the matrix $\Phi^{\mathsf{T}} \Gamma^m \Phi$:

$$\begin{pmatrix} \int_{\mathbb{R}^2} G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_1 G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_2 G^m G^s d\mathbf{y} \\ \int_{\mathbb{R}^2} y_1 G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_1^2 G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_1 y_2 G^m G^s d\mathbf{y} \\ \int_{\mathbb{R}^2} y_2 G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_1 y_2 G^m G^s d\mathbf{y} & \int_{\mathbb{R}^2} y_2^2 G^m G^s d\mathbf{y} \end{pmatrix} \tag{31}$$

to simplify the notation we have omitted the arguments of the functions in the integrand. For the G^m-function the argument is the model error $f(\mathbf{x}+\mathbf{y}) - a_{00} - a_{10} y_1 - a_{01} y_2$. The vector $\Phi^{\mathsf{T}} \Gamma^m f_{-\mathbf{x}}$ equals

$$\begin{pmatrix} \int_{\mathbb{R}^2} f(\mathbf{x}+\mathbf{y}) G^m (f(\mathbf{x}+\mathbf{y}) - a_{00} - a_{10} y_1 - a_{01} y_2) G^s(\mathbf{y}) d\mathbf{y} \\ \int_{\mathbb{R}^2} y_1 f(\mathbf{x}+\mathbf{y}) G^m (f(\mathbf{x}+\mathbf{y}) - a_{00} - a_{10} y_1 - a_{01} y_2) G^s(\mathbf{y}) d\mathbf{y} \\ \int_{\mathbb{R}^2} y_2 f(\mathbf{x}+\mathbf{y}) G^m (f(\mathbf{x}+\mathbf{y}) - a_{00} - a_{10} y_1 - a_{01} y_2) G^s(\mathbf{y}) d\mathbf{y} \end{pmatrix} \tag{32}$$

Eq.(23) then can be used to calculate the new estimate of the optimal parameter vector $\mathbf{a}^{i+1}$.

In Fig. 7 the robust estimation of the zero order coefficient based on a first order facet model is shown. For this image the difference with a zero order facet model estimation can only be observed in regions of slowly varying luminance (like in the background).

Fig. 7. Robust Estimation of Local Image Structure. On the left the cameraman image with noise added and on the right the robust estimation of the zero order coefficient in a first order facet model.

3.3 Color Image Structure

In this section we generalize the robust facet models for scalar images to models for vectorial images. The analysis is done for color images but is valid for all vectorial images.

A color image $\mathbf{f} = (f^1\ f^2\ f^3)$ at any position $\mathbf{x}$ has three color components $f^1(\mathbf{x})$, $f^2(\mathbf{x})$ and $f^3(\mathbf{x})$. The local model for a color image using a basis

$$\Phi = (\phi_1\ \phi_2\ \cdots\ \phi_K) \tag{33}$$

is chosen as:

$$\hat{\mathbf{f}}(\mathbf{x}+\mathbf{y}) = \Phi\mathbf{A} = \Phi\left(\mathbf{a}_1\ \mathbf{a}_2\ \mathbf{a}_3\right) \tag{34}$$

where $\mathbf{A} = \left(\mathbf{a}_1\ \mathbf{a}_2\ \mathbf{a}_3\right)$ is the $K \times 3$ parameter matrix. The column $\mathbf{a}_i$ represents the parameter vector in the approximation $\hat{f}_i = \Phi\mathbf{a}_i$ of the i-th color component. Each of the color components is thus approximated as a linear combination of K basis functions. The model error is now written as:

$$\epsilon(\mathbf{x}) = \int_{\mathbb{R}^d} \rho\left(\sqrt{(f^1_{-\mathbf{x}} - \Phi\mathbf{a}_1)^2 + (f^2_{-\mathbf{x}} - \Phi\mathbf{a}_2)^2 + (f^3_{-\mathbf{x}} - \Phi\mathbf{a}_3)^2}\right) G^s(\mathbf{y})d\mathbf{y} \tag{35}$$

It is not hard to prove that in this case

$$\frac{\partial\epsilon}{\partial\mathbf{A}} = 0 \Longleftrightarrow \Phi^{\mathsf{T}}\Gamma^m\mathbf{f} = \Phi^{\mathsf{T}}\Gamma^m\Phi\mathbf{A} \tag{36}$$

where Γ^m is the 'diagonal' operator that multiplies a function point wise with the function: $G^m\left((f^1_{-\mathbf{x}} - \Phi\mathbf{a}_1)^2 + (f^2_{-\mathbf{x}} - \Phi\mathbf{a}_2)^2 + (f^3_{-\mathbf{x}} - \Phi\mathbf{a}_3)^2\right)$. As Γ^m is dependent on the parameter matrix $\mathbf{A}$ we arrive at a iterated weighted least squares estimator:

$$\mathbf{A}^{i+1} = (\Phi^{\mathsf{T}}\Gamma^m(\mathbf{A}^i)\Phi)^{-1}\Phi^{\mathsf{T}}\Gamma^m(\mathbf{A}^i)\,\mathbf{f} \tag{37}$$

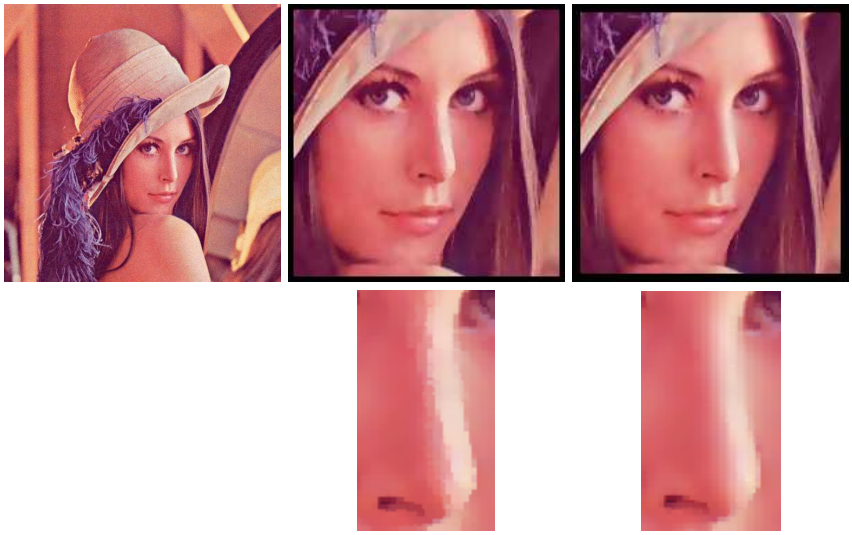

Fig. 8. Robust Estimation of Local Structure in Color Images. On the first row from left to right: the 'Lena' image with some noise added to it, the zero-order facet model based robust estimator of the color values and the robust estimator based on a first order based facet model. On the second row we show a detail from the image above.

The estimation of the robust facet model for color images is thus almost the same as for scalar images. The three color components are dealt with independently, only the error weights operator Γ^m is dependent on all three color components.

In Fig. 8 the robust estimators are shown that are based on a zero order facet model and on a first order facet model. Especially in the nose-region the first order model based robust estimator performs better then the zero order model based robust estimator.

4 Robust Estimation of Orientation

In the previous sections we have considered local image models for the image values (grey value and color). In this section we look at robust estimation of the orientation of image structures.

Oriented patterns are found in many imaging applications, e.g. in fingerprint analysis, and in geo-physical analysis of soil layers. The classical technique to estimate the orientation of the texture is to look at the set of luminance gradient vectors in a local neighborhood. In an image patch showing a stripe pattern in only one orientation we can clearly distinguish the orientation as the line cluster in gradient space perpendicular to the stripes (see Fig. 9(a-b)). A straightforward eigenvector analysis of the covariance matrix will reveal the orientation of the texture. The covariance matrix of the gradient vectors in an image neighborhood is often used to estimate the local orientation [11–14]

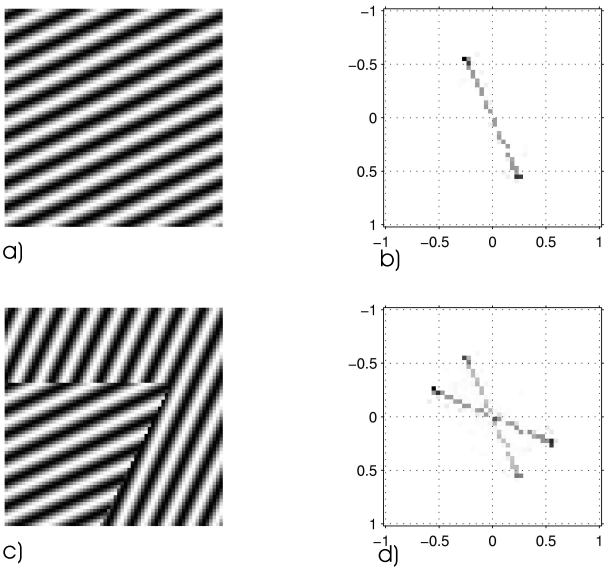

Fig. 9. Histograms of gradient vector space. In (a) an image (64×64) is shown with in (b) the histogram of all gradient vectors (where darker shades indicate that those gradient vectors occur often in the image. In (c) a composition of two differently oriented patterns is shown with corresponding histogram in (d).

In case the local neighborhood is taken from the border of two differently oriented patterns (see Fig. 9) an eigenvector analysis of the covariance matrix will mix both orientations resulting in a 'smoothing' of the orientation estimation.

In case the regions showing different textures are of sufficient size it is possible to use a post-processing step to sharpen the smoothed orientation measurements. A classical way of doing so is the Kuwahara-Nagao operator [15–17]. At a certain position in an image this operator searches for a nearby neighborhood where the (orientation) response is more homogeneous then it is at the border. That response is then used at the point of interest. In this way the neighborhoods are not allowed to cross the borders of the textured regions. In [18] we have shown that the classic Kuwahara-Nagao operator can be interpreted as a 'macroscopic' version of a PDE image evolution that combines linear diffusion (smoothing) with morphological sharpening.

Again consider the texture in Fig. 9(a). The histogram of the gradient vectors in this texture patch is shown in Fig. 9(b). Let $\mathbf{v}$ be the true orientation vector of the patch, i.e. the unit vector perpendicular to the stripes. In an ideal image patch every gradient vector should be parallel to the orientation $\mathbf{v}$. In practice they will not be parallel. The error of a gradient vector $\mathbf{g}(\mathbf{y})$ observed in a point $\mathbf{y}$ with respect to the orientation $\mathbf{v}(\mathbf{x})$ of an image patch centered at location $\mathbf{x}$ is defined as:

$$e(\mathbf{x},\mathbf{y}) = \|\mathbf{g}(\mathbf{y}) - (\mathbf{g}(\mathbf{y})^{\mathsf{T}}\mathbf{v}(\mathbf{x}))\mathbf{v}(\mathbf{x})\| \tag{38}$$

The difference $\mathbf{g}(\mathbf{y}) - (\mathbf{g}(\mathbf{y})^{\mathsf{T}}\mathbf{v}(\mathbf{x}))\mathbf{v}(\mathbf{x})$ is the projection of $\mathbf{g}$ on the normal to $\mathbf{v}$. The error $e(\mathbf{x},\mathbf{y})$ thus measures the perpendicular distance from the gradient vector $\mathbf{g}(\mathbf{y})$ to the orientation vector $\mathbf{v}(\mathbf{x})$. Integrating the squared error over all positions $\mathbf{y}$ using a soft Gaussian aperture for the neighborhood definition we

define the total error:

$$\epsilon(\mathbf{x}) = \int_{\Omega} e^2(\mathbf{x},\mathbf{y}) G^s(\mathbf{x}-\mathbf{y}) d\mathbf{y} \tag{39}$$

The error measure can be rewritten as:

$$\epsilon = \int_{\Omega} \mathbf{g}^{\mathsf{T}}\mathbf{g} G^s d\mathbf{y} - \int_{\Omega} \mathbf{v}^{\mathsf{T}}(\mathbf{g}\mathbf{g}^{\mathsf{T}})\mathbf{v} G^s d\mathbf{y}. \tag{40}$$

where we have omitted the arguments of the functions. Minimizing the error thus is equivalent with maximizing:

$$\int_{\Omega} \mathbf{v}^{\mathsf{T}}(\mathbf{g}\mathbf{g}^{\mathsf{T}})\mathbf{v} G^s d\mathbf{y}, \tag{41}$$

subject to the constraint that $\mathbf{v}^{\mathsf{T}}\mathbf{v} = 1$. Note that $\mathbf{v}$ is not dependent on $\mathbf{y}$ so that we have to maximize:

$$\mathbf{v}^{\mathsf{T}} \left(\int_{\Omega} (\mathbf{g}\mathbf{g}^{\mathsf{T}}) G^s d\mathbf{y} \right) \mathbf{v} = \mathbf{v}^{\mathsf{T}} \mu^s \mathbf{v} \tag{42}$$

where μ^s is the *structure tensor*.

Using the method of Lagrange multipliers to maximize $\mathbf{v}^{\mathsf{T}}\mu^s\mathbf{v}$ subject to the constraint that $\mathbf{v}^{\mathsf{T}}\mathbf{v} = 1$, we need to find an extremum of

$$\lambda(1 - \mathbf{v}^{\mathsf{T}}\mathbf{v}) + \mathbf{v}^{\mathsf{T}}\mu^s\mathbf{v}. \tag{43}$$

Differentiating with respect to $\mathbf{v}$ (remember that $d\mathbf{v}^{\mathsf{T}}A\mathbf{v}/d\mathbf{v} = 2A\mathbf{v}$ in case $A = A^{\mathsf{T}}$) and setting the derivative equal to zero results in:

$$\mu^s\mathbf{v} = \lambda\mathbf{v}. \tag{44}$$

The 'best' orientation thus is an eigenvector of the structure tensor. Substitution in the quadratic form then shows that we need the eigenvector corresponding to the largest eigenvalue.

The least squares orientation estimation works well in case all gradients in the ensemble of vectors in an image neighborhood all belong to the same oriented pattern. In case the image patch shows two oriented patterns the least squares estimate will mix the two orientations and give a wrong result.

A robust estimator is constructed by introducing the Gaussian error norm once again:

$$\epsilon(\mathbf{x}) = \int_{\Omega} \rho(e(\mathbf{x},\mathbf{y})) G^s(\mathbf{x}-\mathbf{y}) d\mathbf{y} \tag{45}$$

In a robust estimator large deviations from the model are not taken into account very heavily. In our application large deviations from the model are probably due to the mixing of two different linear textures (see Fig. 9(c-d)).

The error, Eq.(45), can now be rewritten as (we will omit the spatial arguments):

$$\epsilon = \int_{\Omega} \rho\left(\sqrt{\mathbf{g}^{\mathsf{T}}\mathbf{g} - \mathbf{v}^{\mathsf{T}}(\mathbf{g}\mathbf{g}^{\mathsf{T}})\mathbf{v}} \right) G^s d\mathbf{y}. \tag{46}$$

Again we use a Lagrange multiplier to minimize subject to the constraint that $\mathbf{v}^\mathsf{T}\mathbf{v} = 1$:

$$\frac{d}{d\mathbf{v}}\left(\lambda(1-\mathbf{v}^\mathsf{T}\mathbf{v}) + \int_\Omega \rho\left(\sqrt{\mathbf{g}^\mathsf{T}\mathbf{g} - \mathbf{v}^\mathsf{T}(\mathbf{g}\mathbf{g}^\mathsf{T})\mathbf{v}}\right) G^s d\mathbf{y}\right) = 0. \tag{47}$$

Using Eq.(15) as the error function leads to

$$\eta(\mathbf{v})\mathbf{v} = \lambda\mathbf{v} \tag{48}$$

where

$$\eta(\mathbf{v}) = \int_\Omega \mathbf{g}\mathbf{g}^\mathsf{T} G^m(\sqrt{\mathbf{g}^\mathsf{T}\mathbf{g} - \mathbf{v}^\mathsf{T}(\mathbf{g}\mathbf{g}^\mathsf{T})\mathbf{v}})G^s d\mathbf{y}. \tag{49}$$

The big difference with the least squares estimator is that now the matrix η is dependent on $\mathbf{v}$ (and on $\mathbf{x}$ as well). Note that η can be called a 'robustified' structure tensor in which the contribution of each gradient vector is weighted not only by its distance to the center point of the neighborhood, but also weighted according to its 'distance' to the orientation model. Weickert et. al. [19] also introduce a non linear version of the structure tensor that is close in spirit to the robust structure tensor η.

We propose the following *fixed point* iteration scheme to find a solution. Let $\mathbf{v}^i$ be the orientation vector estimate after i iterations. The estimate is then updated as the eigenvector $\mathbf{v}^{i+1}$ of the matrix $\eta(\mathbf{v}^i)$ corresponding to the largest eigenvalue, i.e. we solve:

$$\eta(\mathbf{v}^i)\mathbf{v}^{i+1} = \lambda\mathbf{v}^{i+1} \tag{50}$$

The proposed scheme is a generalization of the well-known fixed point scheme (also called *functional iteration*) to find a solution of the equation $v = F(v)$.

Note that the iterative scheme does not necessarily lead to the *global* minimum of the error. In fact often we are not even interested in that global minimum. Consider for instance the situation of a point in region A (with orientation α_1) that is surrounded by many points in region B (with orientation β). It is not to difficult to imagine a situation where the points of region B outnumber those in region A. Nevertheless we would like our algorithm to find the orientation α whereas the global minimum would correspond with orientation β. Because our algorithm starts in the initial orientation estimate and then finds the local minimum nearest to the starting point we hopefully end up in the desired *local* minimum: orientation α.

The choice for an initial estimate of the orientation vector is thus crucial in a robust estimator in case we have an image patch showing two (or more) striped patterns.

In Fig. 10 and Fig. 11 the robust estimation of orientation for a simple test image (without noise and with noise). For the robust estimation in both cases we have used the orientation in location $\mathbf{x}$ that resulted from the least squares estimator as the initial orientation vector in that point. In both cases only 5 iterations are used.

From both the noise free and the noise corrupted texture images it is evident that the robust estimation performs much better at the border of the textured regions.

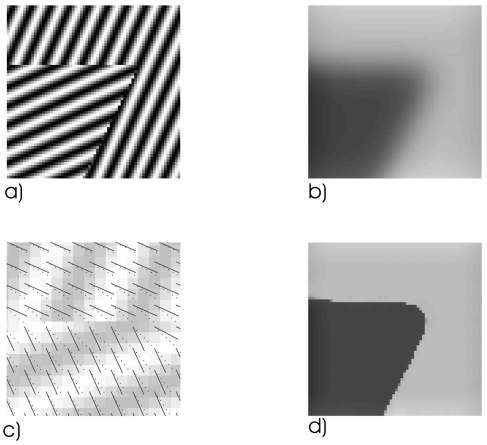

Fig. 10. Least Squares versus Robust Orientation Estimation. In (a) a generated noise free image is shown. The texture is made out of two regions each differently oriented. In (b) the orientation field $\alpha = \arctan(v_2/v_1)$ is shown that results from the least squares estimate. In (d) the orientation field is shown resulting from the robust estimation. In (c) a *detail* of the orientation vector fields for both the least squares estimation (dotted lines) and the robust estimation (solid lines) are shown.

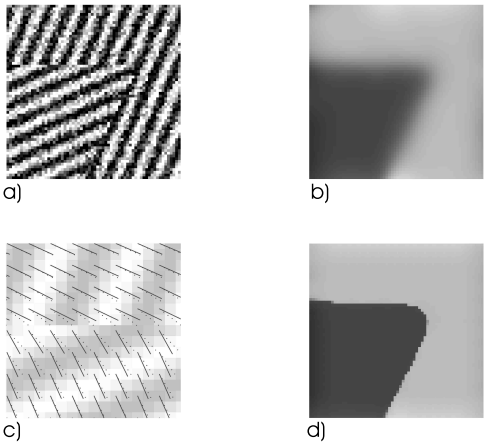

Fig. 11. Least Squares versus Robust Orientation Estimation. Same experiment as figure 10 but with noise added.

5 Conclusions

In this paper we have described robust estimators of local image structure. Our starting point was the Gaussian facet model, using a Gaussian soft aperture function to define the local neighborhood.

Using the Gaussian error norm resulted in an iterative procedure for the robust estimator that is essentially a fixed point iteration. The advantage is that very few iterations are needed in most cases (the maximum number of iterations used in the examples in this paper ranges from 5 to 10).

Through the Gaussian facet model we are able to link the robust estimation of local image structure with classical linear scale-space techniques.

References

1. Haralick, R., andT.J. Laffey, L.W.: The topographic primal sketch. IJPR (1983) 50–72
2. Farnebäck, G.: Spatial Domain Methods for orientation and velocity measurements. PhD thesis, Linköping University (1999)
3. Besl, P., Birch, J., Watson, L.: Robust window operators. Machine Vision and Applications **2** (1989) 179–191
4. van de Weijer, J., van den Boomgaard, R.: Local mode filtering. In: CVPR01. (2001) II:428–443
5. van den Boomgaard, R., van de Weijer, J.: On the equivalence of local-mode finding, robust estimation and mean-shift analysis as used in early vision tasks. In: International Conference on Pattern Recognition. (2002) 30927–30930
6. van den Boomgaard, R., van de Weijer, J.: Robust estimation of orientation for texture analysis. In: The 2nd International workshop on texture analysis and synthesis, Copenhagen (2002) 135—138
7. Heuckel, M.: An operator which locates edges in digital pictures. J. Association for computing machinery **18** (1971) 113–125
8. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Proc. of the Sixth International Conference on Computer Vision, (Bombay, India) 839–846
9. Smith, S., Brady, J.: Susan: A new approach to low-level image processing. International Journal of Computer Vision **23** (1997) 45—78
10. Black, M., Sapiro, G., Marimont, D., Heeger, D.: Robust anisotropic diffusion. IEEE Trans. Image Processing **7** (1998) 421–432
11. Kass, M., Witkin, A.: Analyzing oriented patterns. Computer Graphics and Image Processing **37** (1987) 363–385
12. Bigun, J., Granlund, G., Wiklund, J.: Multidimensional orientation estimation with application to texture analysis and optical flow. IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991) 775–789
13. Lindeberg, T., Garding, J.: Shape from texture from a multi-scale perspective. In Nagel, H.H., ed.: Proc. 4th Int. Conf. on Computer Vision. (1993) 683–691
14. Weicker, J.: Anisotropic diffusion in image processing, Stuttgart, Tuebner Verlag (1997)
15. Kuwahara, M., Hachimura, K., Eiho, S., Kinoshita, M.: Processing of riangiocardiographic images. In Preston, K., Onoe, M., eds.: Digital Processing of Biomedical Images, (Plenum Press)
16. Nagao, M., Matsuyama, T.: Edge preserving smoothing. Computer Graphics and Image Processing **9** (1979) 394–407
17. Bakker, P., van Vliet, L., Verbeek, P.: Edge preserving orientation adaptive filtering. In Boasson, M., Kaandorp, J., Tonino, J., Vosselman, M., eds.: ASCI'99, Proc. 5th Annual Conference of the Advanced School for Computing and Imaging. (1999) 207–213
18. van den Boomgaard, R.: The Kuwahara-Nagao operator decomposed in terms of a linear smoothing and a morphological sharpening. In Talbot, H., Beare, R., eds.: Proceedings of the 6th International Symposium on Mathematical Morphology, Sydney, Australia, CSIRO Publishing (2002) 283–292
19. Weickert, J., Brox, T.: Diffusion and regularization of vector- and matrix valued images. In M. Z. Nashed, O.S., ed.: Inverse Problems, Image Analysis and Medical Imaging. Volume 313. (2002) 252—268