A Geometric Functional for Gradient Approximation

Nir A. Sochen[2], Robert M. Haralick[1] and Yehoshua Y. Zeevi[2]

[1] EE Department, University of Washington
Seattle, WA 98195
haralick@ee.washington.edu
[2] EE Department, Technion - Israel Institute of Technology
Technion City, Haifa 32000, ISRAEL
sochen,zeevi@ee.technion.ac.il

Abstract. *A method for estimation of an image gradient field based on Bayesian approach which is formulated in a geometric framework, is presented. The probability configuration of the gradient field is maximized by a steepest descent method, leading to a non-linear diffusion type equation with added constraints. The derivatives are assumed to be piecewise smooth and the Beltrmi framework is used to build a specially adaptive smoothing process. Both the case where the embedding space is Euclidean and the case where non trivial geometry is induced by the tangent bundle structure are treated.*

1 Introduction

It is widely accepted that gradients are of utmost importance in early vision analysis such as image enhancement and edge detection. Several numerical recipes are known for derivatives estimation. All based on fixed square or rectangular neighborhoods of different sizes. This fixed size and shape way of estimation does not reflect the structure of images and bound to produce errors, especially near edges where the estimate on one side of the edge may wrongly influence the estimate on the other side of it. In the places where the image is relatively smooth, least square estimates of derivatives (the facet approach [2]) in large area neighborhoods will give best results [1]. But in places where the underlying image intensity surface is not smooth and therefore not fittable by a small degree bivariate polynomial, the neighborhood should be smaller and rectangular, with the long axis of the rectangle aligned with the direction of the directional derivative being estimated.

From this point of view, it is natural to suggest a varying size and shape neighborhood in order to increase both the robustness of the estimate to noise, and its correctness. Calculating directly for each point of the image its optimal neighborhood for gradient estimation is possible but cumbersome. We opt for a different approach, which uses a geometry driven diffusion that produces implicitly, and in a sub-pixel accuracy, the desirable effect.

We follow the Beltrami framework and represent the image as a two-dimensional Riemannian surface embedded in a spatial-feature space [12]. There are several ways of formulating the problem yielding different equations and different results. Taking the x and y directional derivatives as features, and adding the intensity itself, one obtains simultaneously a denoised image and its gradient estimates. One may also select as a feature space the tangent bundle itself with its induced geometry. Short time kernel can be found for the non-linear diffusion equations that are produced. This kernel is a Gaussian with locally dependent covariance metric. The solution to the equations involves therefore a convolution with an elliptic shaped Gaussian, where the size and eccentricity of the ellipse changes from point to point on the image.

The paper is organized as follows: In section 2 we review the Beltrami framework. The straitforward application for the x and y directional derivatives is presented in Section 3. We continue in Section 4 by a joint intensity and derivatives estimation. We study, in Section 5, the consequences of geometry of the tangent bundle on this estimation. Section 6 presents results and the conclusions are summarized in Section 7.

2 A Geometric Measure on Embedded Maps

According to the proposed approach, an image is represented as a two-dimensional Riemannian surface embedded in a higher dimensional spatial-feature Riemannian manifold. Let $\sigma^\mu, \quad \mu = 1, 2$, be the local coordinates on the image surface and let $X^i \quad i = 1, 2, \ldots, m$ be the coordinates on the embedding space, then, the embedding map is given by

$$(X^1(\sigma^1, \sigma^2), X^2(\sigma^1, \sigma^2), \ldots, X^m(\sigma^1, \sigma^2)). \tag{1}$$

We deal in this paper with two possibilities. In both we have $X^1 = x$ and $X^2 = y$ as the spatial dimensions of the embedding manifold. In the first possibility we have a four-dimensional spatial-feature manifold with $X^3 = I_x$ and $X^4 = I_y$ as coordinates. The second possibility includes the intensity itself as a coordinate of the feature manifold. We distinguish in this case between flat feature manifold and a curved one by taking the tangent bundle geometry into account.

Riemannian manifolds are manifolds endowed with a bi-linear positive-definite symmetric tensor which is called a *metric*. Denote by $(\Sigma, (g_{\mu\nu}))$ the image manifold and its metric, and by $(M, (h_{ij}))$ the space-feature manifold and its metric. Then, the map $\mathbf{X} : \Sigma \to M$ has the following weight [8]

$$E[X^i, g_{\mu\nu}, h_{ij}] = \int d^2\sigma \sqrt{g} g^{\mu\nu} (\partial_\mu X^i)(\partial_\nu X^j) h_{ij}(\mathbf{X}), \tag{2}$$

where the range of indices is $\mu, \nu = 1, 2$, and $i, j = 1, \ldots, m = \dim M$, and we use the Einstein summation convention: identical indices that appear one up and one down are summed over. We denote by g the determinant of $(g_{\mu\nu})$ and by $(g^{\mu\nu})$ its inverse. In the above expression $d^2\sigma\sqrt{g}$ is an area element of the image

manifold. The rest, i.e. $g^{\mu\nu}(\partial_\mu X^i)(\partial_\nu X^j)h_{ij}(\mathbf{X})$, is a generalization of L_2. It is important to note that this expression (as well as the area element) does not depend on the local coordinates one chooses.

The feature evolves in a geometric way via the steepest descent equations

$$X_t^i \equiv \frac{\partial X^i}{\partial t} = -\frac{1}{2\sqrt{g}} h^{il} \frac{\delta E}{\delta X^l}. \tag{3}$$

Note that we exploited the freedom to multiply the Euler-Lagrange equations by a strictly positive function and a positive definite matrix. This factor is the simplest one that does not change the minimization solution while giving a reparameterization invariant expression. This choice guarantees that the flow is geometric and does not depend on the parameterization.

The variational derivative of E with respect to the coordinate functions is given by

$$-\frac{1}{2\sqrt{g}} h^{il} \frac{\delta E}{\delta X^l} = \Delta_g X^i + \Gamma^i_{jk}(\partial_\mu X^j)(\partial_\nu X^k) g^{\mu\nu}, \tag{4}$$

where the operator acting on X^i in the first term is the natural generalization of the Laplacian from flat spaces to manifolds, called *the second order differential parameter of Beltrami* [7], or, in short, the *Beltrami operator*. It is given in term of the metric as

$$\Delta_g X^i = \frac{1}{\sqrt{g}} \partial_\mu (\sqrt{g} g^{\mu\nu} \partial_\nu X^i). \tag{5}$$

In the second term of Eq. (4), the Γ^i_{jk} are the Levi-Civita connection's coefficients with respect to the metric h_{ij}. they describe the geometry of the embedding space [13] as follows

$$\Gamma^i_{jk} = \frac{1}{2} h^{il} (\partial_j h_{lk} + \partial_k h_{jl} - \partial_l h_{jk}). \tag{6}$$

3 Bayesian Formulation of the Gradient Estimation

Denote by (x_r, y_s) the sampling points and by $I^0_{rs} \equiv I^0(x_r, y_s)$ the grey-levels at the sampling points.

Based on these data we want to infer the underlying function $I(x, y)$ and its gradient vector field $\mathbf{V}(x, y)$. The analysis is easier in the continuum and we refer from now on to I^0 as to a continuous function. In practice we can skip a stage and find the derivatives without referring to the underlying function. The inference is described by the *posterior probability distribution*

$$P(\mathbf{V}(x,y)|I^0(x,y)) = \frac{P(I^0(x,y)|\mathbf{V}(x,y))P(\mathbf{V}(x,y))}{P(I^0(x,y))}$$

The first term in the numerator, $P(I^0(x,y))|\mathbf{V}(x,y))$, is the probability of the sampled grey-level values given the vector field $\mathbf{V}(x, y)$. The second term is the prior distribution on vector fields assumed by our model. The denominator is independent of $\mathbf{V}$ and will therefore be ignored from now on.

Assume that $P(A|B)$ is given by a Gibbsian form :

$$P(A|B) = Ce^{-\alpha E(A,B)},$$

we get

$$-\log P(\mathbf{V}(x,y)|I^0(x,y)) = \alpha E_1(I^0(x,y), \mathbf{V}(x,y)) + \beta E_2(\mathbf{V}(x,y))$$

If we use the Euclidean L_2 norm, we get

$$\begin{aligned} E_1(I^0(x,y), \mathbf{V}(x,y)) &= \frac{1}{2}C_1 \int dxdy \left(|\mathbf{V} - \nabla I|^2\right) \\ E_2(\mathbf{V}(x,y)) &= \frac{1}{2}C_2 \int dxdy \left(|\nabla \mathbf{V}|^2\right) + E_3, \end{aligned} \tag{7}$$

where the first term is a fidelity term that forces the vector field $\mathbf{V}$ to be close enough to the gradient vector field of $I(x,y)$. The second term is a regularization term that guarantees certain smoothness properties of the solution. The third term is a constraint on the vector field to be a gradient field of a function. Its form is

$$E_3(I(x,y), \mathbf{V}(x,y) = \frac{1}{2}C_3 \int dxdy(V1_y - V2_x)^2.$$

Alternatively, we may adopt a more sophisticated regularization based on geometric ideas. This is treated in the subsequent sections.

Maximization the posterior probability amounts to the minimization of the energy. We do that with the steepest descent method which leads eventually to non-linear diffusion type equations.

4 Gradient Estimation: Euclidean Embedding Method

In this section the feature space includes the x and y intensity derivatives only. The embedding maps read

$$(X^1 = \sigma^1, X^2 = \sigma^2, X^3 = I_x(\sigma^1, \sigma^2), X^4 = I_y(\sigma^1, \sigma^2)), \tag{8}$$

where the subscripts stand for derivatives in the direction that the subscript indicates. With a slight abuse of notations we write it as $(x, y, V1, V2)$, where we identify the spatial part with x and y the same way as in Section 2 and we denote I_x by $V1$ and I_y by $V2$. We assume that these are Cartesian coordinates of $\mathbb{R}^4$ i.e. we assume that the metric of the embedding space is the identity matrix δ_{ij}.

Having explicitly the embedding maps and the metric of the embedding space we can infer the metric, and therefore all internal geometric properties, of the image surface. The metric we find this way,

$$g_{\mu\nu}(\sigma^1, \sigma^2) = h_{ij}(\mathbf{X})(\partial_\mu X^i)(\partial_\nu X^j), \tag{9}$$

is called the induced metric. Its explicit form is in our case

$$(g_{\mu\nu}(x,y)) = \begin{pmatrix} 1+V1_x^2+V2_x^2 & V1_xV1_y+V2_xV2_y \\ V1_xV1_y+V2_xV2_y & 1+V1_y^2+V2_y^2 \end{pmatrix}, \tag{10}$$

where the subscripts x and y denote directional derivatives.

The functional reads

$$\begin{aligned} E_1(I^0(x,y),\mathbf{V}(x,y)) &= \frac{1}{2}C_1\int dxdy\sqrt{g}\left(|\mathbf{V}-\nabla I^0|^2\right) \\ E_2(\mathbf{V}(x,y)) &= \frac{1}{2}C_2\int dxdy\sqrt{g}g^{\mu\nu}(\partial_\mu X^i)(\partial_\nu X^i) \\ E_3(\mathbf{V}(x,y)) &= \frac{1}{2}C_3\int dxdy\sqrt{g}\epsilon^{\mu\nu}\partial_\nu V(\mu). \end{aligned} \tag{11}$$

where $\epsilon^{\mu\nu}$ is the antisymmetric tensor. Explicitly

$$(\epsilon^{\mu\nu}) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \tag{12}$$

where there is an implicit summation over μ, and $V(\mu=1)=V1$ ($V(\mu=2)=V2$). Note the convention $\partial_1=\partial_{\sigma^1}=\partial_x$ and similarly for $2\equiv y$.

The steepest descent method results in two coupled partial differential equations (PDEs):

$$\begin{aligned} V1_t &= C_2\Delta_g V1 - C_1(V1-I_x^0) + \frac{C_3}{\sqrt{g}}\partial_y(\sqrt{g}\epsilon^{\mu\nu}\partial_\nu V(\mu)) \\ V2_t &= C_2\Delta_g V2 - C_1(V2-I_y^0) - \frac{C_3}{\sqrt{g}}\partial_x(\sqrt{g}\epsilon^{\mu\nu}\partial_\nu V(\mu)), \end{aligned} \tag{13}$$

with the initial conditions

$$\begin{aligned} V1(x,y,t=0) &= I_x^0(x,y) \\ V2(x,y,t=0) &= I_y^0(x,y). \end{aligned} \tag{14}$$

The coupling between the equations is through the non-trivial structure of the metric that involves both the x and y derivatives estimations, and through the constraint. These equations describe a flow of the two-dimensional image surface in the spatial-feature space. Each point on the surface moves in the direction of the mean-curvature vector projected on the feature coordinates. This results in an edge preserving flow, where we refer to the edge of the gradient functions and not of the intensity itself, that denoises more between regions and less across edges.

One may use for the initial conditions and for the x and y derivatives that appear in the metric and in the Laplace-Beltrami operator, simple numerical methods. The true estimate of I_x and I_y is refined through the PDEs to yield a denoised result.

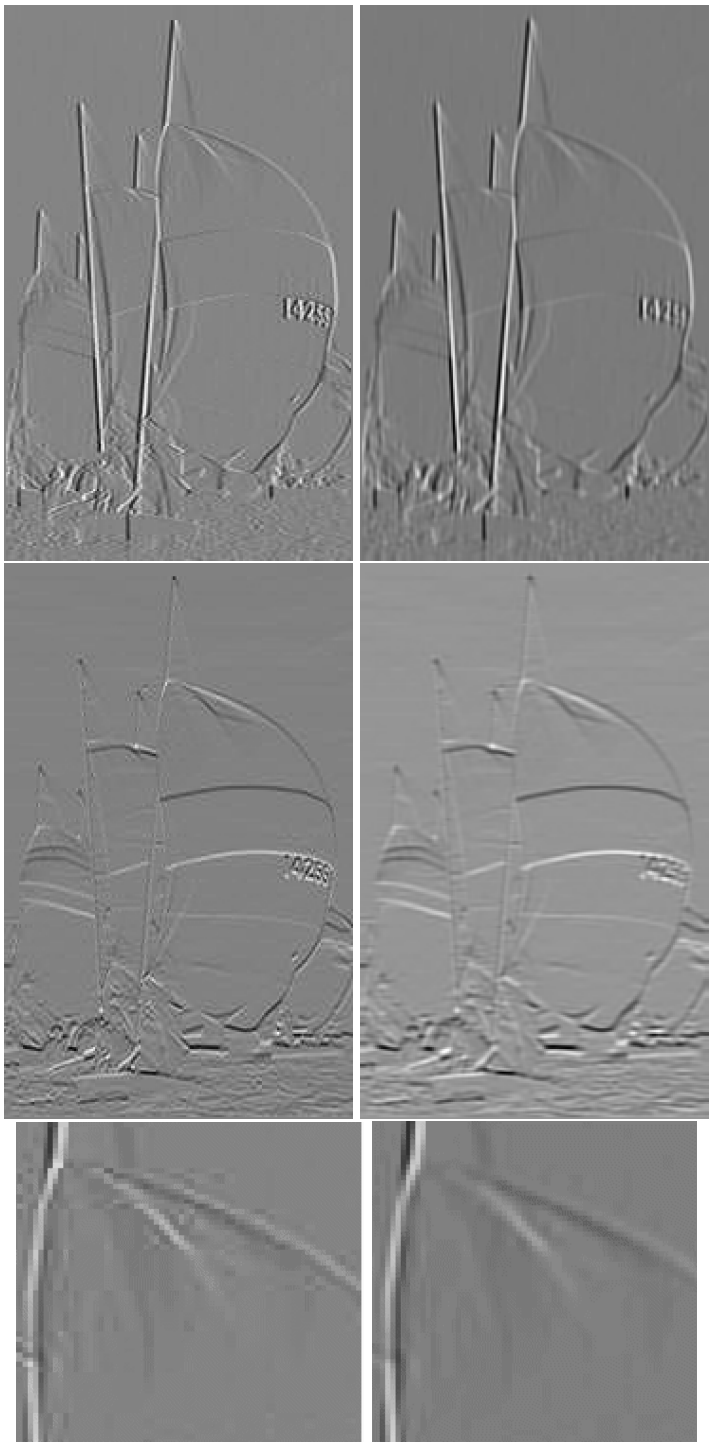

Fig. 1. Top left and right respectively: The original noisy x-derivative and its estimation by the Euclidean embedding method. Middle left and right respectively: The original noisy y-derivative and its estimation by the Euclidean embedding method. A detail from the x-derivative original and from the x-derivative estimated by the proposed method are shown in the bottom left and right respectively.

5 Gradient Estimation: Second Method

In this section we include the intensity in the feature space. The embedding map is accordingly

$$(x, y, I(x,y), V1(x,y), V2(x,y)). \tag{15}$$

We assume again that these are Cartesian coordinates of $\mathbb{R}^5$ and, therefore, $h_{ij} = \delta_{ij}$. This implies a similar induced metric:

$$(g_{\mu\nu}(x,y)) = \begin{pmatrix} 1 + I_x^2 + V1_x^2 + V2_x^2 & I_x I_y + V1_x V1_y + V2_x V2_y \\ I_x I_y + V1_x V1_y + V2_x V2_y & 1 + I_y^2 + V1_y^2 + V2_y^2 \end{pmatrix}. \tag{16}$$

The energy functionals have two more terms: The first represents the fidelity term of the denoised image, the second is an adaptive smoothing term. The functionals are

$$\begin{aligned} E_0(I(x,y), I^0(x,y)) &= \frac{1}{2} C_0 \int dxdy \sqrt{g} (I - I^0)^2 \\ E_1(I^0(x,y), \mathbf{V}(x,y)) &= \frac{1}{2} C_1 \int dxdy \sqrt{g} \left(|\mathbf{V} - \nabla I^0|^2 \right) \\ E_2(\mathbf{V}(x,y)) &= \frac{1}{2} C_2 \int dxdy \sqrt{g} g^{\mu\nu} (\partial_\mu X^i)(\partial_\nu X^i) \\ E_3(\mathbf{V}(x,y)) &= \frac{1}{2} C_3 \int dxdy \sqrt{g} \epsilon^{\mu\nu} \partial_\nu V(\mu). \end{aligned} \tag{17}$$

Since the Levi-Civita connection's coefficients are zero we get the following system as a steepest descent equations:

$$\begin{aligned} I_t &= C_2 \Delta_g I - C_0 (I - I^0) \\ V1_t &= C_2 \Delta_g V1 - C_1 (V1 - I_x^0) + \frac{C_3}{\sqrt{g}} \partial_y (\sqrt{g} \epsilon^{\mu\nu} \partial_\nu V(\mu)) \\ V2_t &= C_2 \Delta_g V2 - C_1 (V2 - I_y^0) - \frac{C_3}{\sqrt{g}} \partial_x (\sqrt{g} \epsilon^{\mu\nu} \partial_\nu V(\mu)), \end{aligned} \tag{18}$$

with the initial conditions

$$\begin{aligned} I(x,y,t=0) &= I^0(x,y) \\ V1(x,y,t=0) &= I_x^0(x,y) \\ V2(x,y,t=0) &= I_y^0(x,y), \end{aligned} \tag{19}$$

where $I^0(x,y)$ is the initial given image.

It is important to understand that $V1$ and $V2$ are estimates of I_{0x} and I_{0y} and not of the denoised I_x and I_y.

6 Derivatives Estimation: Non-Euclidean Tangent Bundle Method

We adopt here the viewpoint that the coordinates $(x, y, V1, V2)$ are coordinates of a spatial-tangent bundle. The tangent bundle of a manifold is the manifold

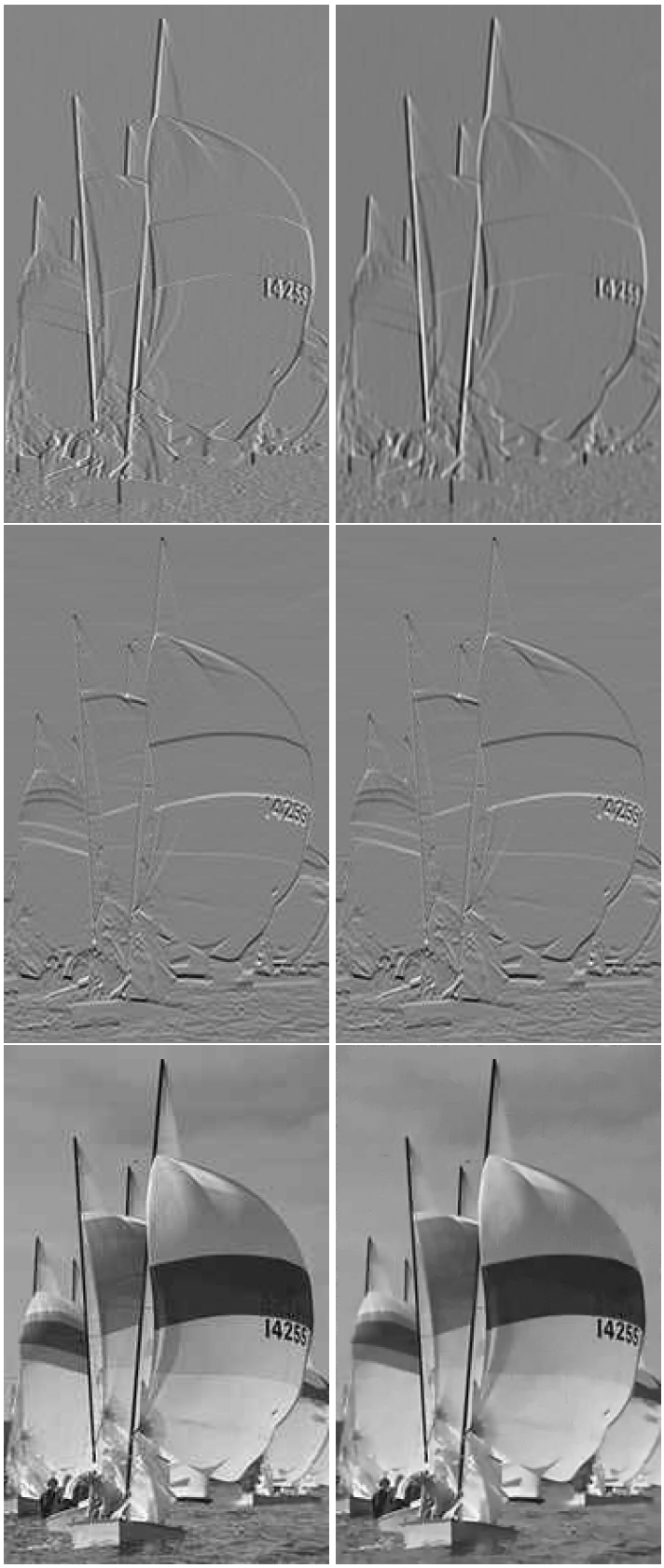

Fig. 2. Top left and right respectively: The original noisy x-derivative and its estimation by the method of Section 5. Middle left and right respectively: The original noisy y-derivative and its estimation by the method of Section 5. The original its denoised counterpart estimated by the method of section 5 are shown in the bottom left and right respectively.

and the collection of the tangent planes of the manifold from all its points. We consider $(x, y, I(x, y))$ to be the embedding maps of a two-dimensional surface embedded in $\mathbb{R}^3$. The induced metric is

$$(\tilde{g}_{\mu\nu}(x, y)) = \begin{pmatrix} 1 + I_x^2 & I_x I_y \\ I_x I_y & 1 + I_y^2 \end{pmatrix}. \tag{20}$$

Inner product of vectors in the tangent plane at the point (x, y) are calculated with respect to the metric at that point:

$$\boldsymbol{V} \cdot \boldsymbol{W} = \tilde{g}_{\mu\nu} V^\mu W^\nu. \tag{21}$$

As a consequence, the embedding space, whose coordinates are $(x, y, V1, V2)$, is not Euclidean! Due to the identification of I_x and $V1$, and of I_y with $V2$, as follows: The metric of the embedding space reads,

$$(h_{ij}(x, y)) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 + V1^2 & V1V2 \\ 0 & 0 & V1V2 & 1 + V2^2 \end{pmatrix}. \tag{22}$$

The induced metric is calculated by Eq. (9), and we have in the Euler-Lagrange equations a second term that takes into account the non-trivial geometry of the embedding space. The energy functional has in this case the following form

$$\begin{aligned} E_1(I^0(x, y), \mathbf{V}(x, y)) &= \frac{1}{2} C_1 \int dxdy \sqrt{g} \left(V^\alpha - \partial^\alpha I^0\right) \tilde{g}_{\alpha\beta} \left(V^\beta - \partial^\beta I^0\right) \\ E_2(\mathbf{V}(x, y)) &= \frac{1}{2} C_2 \int dxdy \sqrt{g} \left(g^{\alpha\beta} \partial_\alpha X^i \partial_\beta X^j h_{ij}\right) \\ E_3(\mathbf{V}(x, y)) &= \frac{1}{2} C_3 \int dxdy \sqrt{g} \epsilon^{\mu\nu} \partial_\nu V(\mu). \end{aligned} \tag{23}$$

Formally the steepest descent equations have the following form

$$\begin{aligned} V(\mu)_t &= -\frac{1}{\sqrt{g}} h^{ij} \frac{\delta E}{\delta V(\mu)} \\ &= C_2 \frac{1}{\sqrt{g}} \partial_\alpha (\sqrt{g} g^{\alpha\beta} \partial_\beta V(\mu)) + C_2 \Gamma_{jk}^{(2+\mu)} \partial_\alpha X^j \partial_\beta X^k g^{\alpha\beta} \\ &\quad + \frac{C_3}{\sqrt{g}} \epsilon^{\mu\nu} \partial_\nu (\sqrt{g} \epsilon^{\alpha\beta} \partial_\beta V(\alpha)) - C_1 (V(\mu) - \partial_\mu I^0) \end{aligned} \tag{24}$$

The induced metric is calculated by Eq. (9). A straitforward application of Eq. (4) gives the only non-vanishing Levi-Civita connection's components as follows:

$$\begin{aligned} \Gamma_{33}^3 = \Gamma_{44}^3 &= \frac{V1}{1 + V1^2 + V2^2} \\ \Gamma_{33}^4 = \Gamma_{44}^4 &= \frac{V2}{1 + V1^2 + V2^2} \end{aligned} \tag{25}$$

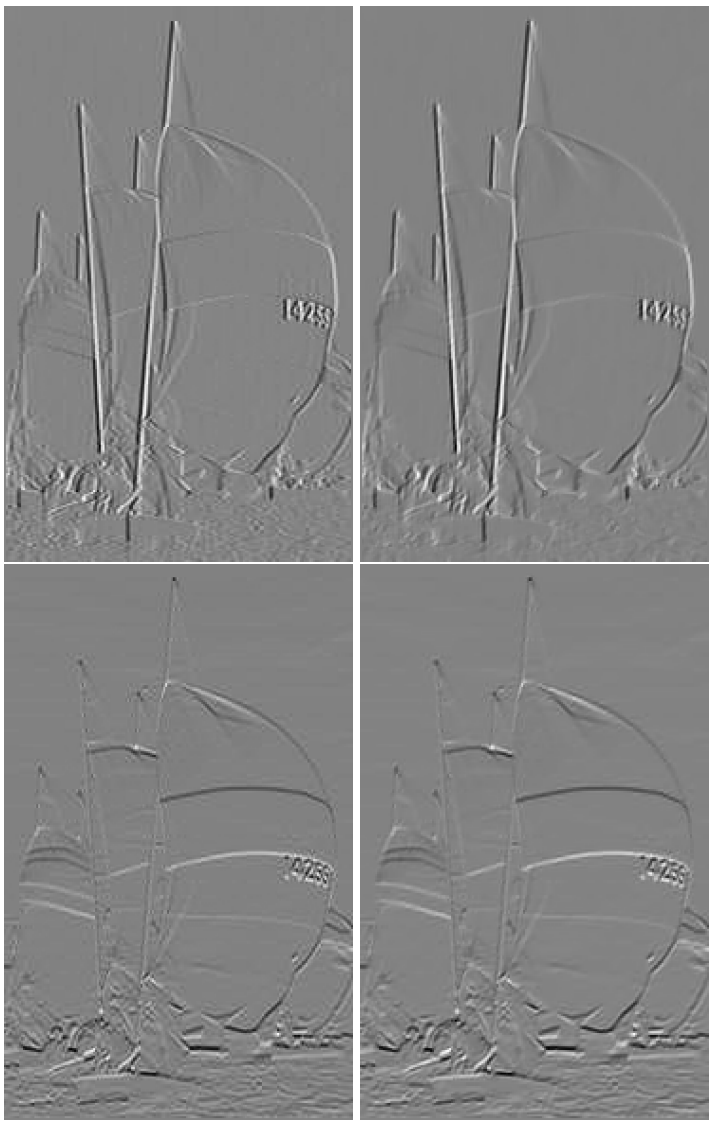

Fig. 3. Top left and right respectively: The original noisy x-derivative and its estimation by the method of Section 6. Middle left and right respectively: The original noisy y-derivative and its estimation by the method of Section 6.

The steepest descent equations are therefore:

$$
\begin{aligned}
V1_t = & \frac{1}{\sqrt{g}}\partial_\alpha(\sqrt{g}g^{\alpha\beta}\partial_\beta V1) \\
& + \frac{V1}{1+V1^2+V2^2}\left(g^{11}V(\mu)_x V(\mu)_x + 2g^{12}V(\mu)_x V(\mu)_y + g^{22}V(\mu)_y V(\mu)_y\right) \\
& + \frac{C_2}{\sqrt{g}}\partial_y(\sqrt{g}\epsilon^{\alpha\beta}\partial_\beta V(\alpha)) - C_1(V1 - I_x^0) \\
V2_t = & \frac{1}{\sqrt{g}}\partial_\alpha(\sqrt{g}g^{\alpha\beta}\partial_\beta V2) \\
& + \frac{V2}{1+V1^2+V2^2}\left(g^{11}V(\mu)_x V(\mu)_x + 2g^{12}V(\mu)_x V(\mu)_y + g^{22}V(\mu)_y V(\mu)_y\right) \\
& - \frac{C_2}{\sqrt{g}}\partial_x(\sqrt{g}\epsilon^{\alpha\beta}\partial_\beta V(\alpha)) - C_1(V2 - I_y^0),
\end{aligned} \tag{26}
$$

where there is an implicit summation over μ in the second term of each equation.

7 Experimental Results

The solutions to the PDE's were obtained by using the explicit Euler scheme where the time derivative is backward and the spatial derivatives are central. The stencil was taken as 3×3. In the first and second methods, we calculated the explicit expressions with the help of the Mathematica program. For the non-Euclidean method we executed the numerical derivation on the metric components.

So far we did not optimized the different parameters, nor the size of the time step. For the Euclidean embedding algorithm we took $C_1 = 0.5, C_2 = 1, C_3 = 8.5$ and the time step of $\Delta t = 0.005$. The estimation results, after 30 iterations are depicted on the right column of Fig. 1 . The results obtained for the joint image and derivatives algorithm, using same parameters and smaller time step after 150 iterations, are depicted in Fig. 2. Results of the non-Euclidean method with $C_1 = 0.5$, $C_2 = 1$, $C_3 = 2.5$ and a normalized range in the interval $[0, 10]$, after 1250 iterations with a very small time step, are shown in Fig. 3.

8 Summary and Conclusion

We combined in this study the Bayesian reasoning and the geometric Beltrami framework, and obtained accurate gradient estimation. three methods based on similar considerations and with different complexity were presented. The first two methods, an Euclidean embedding space was assumed while the third method involved a non-Euclidean embedding space and required another term in the equations to take this non-trivial geometry into account. The requirement that the derivatives found are the x and y derivatives of some underlying function was formulated through a Lagrange multiplier. Close inspection reveals that this requirement is fulfilled only approximately.

By inspection the edge preserving is highest in the non-Euclidean method. Yet this is a sensitive algorithm that needs fine tuning of the parameters and very small time step. The first method seems to over smooth the gradient edges. Experimental results such as those shown in this paper indicate that the joint image and derivative smoothing gives the best compromise between computational cost and quality results.

A detailed analysis and comparison with statistical based methods will appear elsewhere[3].

References

1. R M Haralick and N A Sochen and Y Y Zeevi, "The Design of Tow-Dimensional Gradient estimators Based on One-Dimensional Operators", *IEEE Trans. on Image Processing*, 5, (1996) 155-159.
2. R M Haralick and L G Shapiro *Computer and Robot Vision*, Addison-Wesley Publishing Company, New York, 1992, Chapter 8.
3. R M Haralick and N A Sochen and Y Y Zeevi *in preparation*
4. R Kimmel and R Malladi and N Sochen, "Images as Embedding Maps and Minimal Surfaces: Movies, Color, Texture, and Volumetric Medical Images", *Proc. of IEEE CVPR'97*, (1997) 350-355.
5. R Kimmel and N Sochen and R Malladi, "On the geometry of texture", Report, Berkeley Labs. UC, LBNL-39640, UC-405, November,1996.
6. R Kimmel and N Sochen and R Malladi, "From High Energy Physics to Low Level Vision", *Lecture Notes In Computer Science:* 1252, First International Conference on Scale-Space Theory in Computer Vision, Springer-Verlag, 1997, 236-247.
7. E Kreyszing, "Differential Geometry", Dover Publications, Inc., New York, 1991.
8. A M Polyakov, "Quantum geometry of bosonic strings", *Physics Letters*, **103B** (1981) 207-210.
9. M Proesmans and E Pauwels and L van Gool, "Coupled geometry-driven diffusion equations for low level vision", In Geometric–Driven Diffusion in Computer Vision, Ed. B M ter Haar Romeny, Kluwer Academic Publishers, 1994.
10. N Sochen and R Kimmel and R Malladi, "From high energy physics to low level vision", Report, LBNL, UC Berkeley, LBNL 39243, August, Presented in ONR workshop, UCLA, Sept. 5 1996.
11. N Sochen and R Kimmel and R Malladi , "A general framework for low level vision", *IEEE Trans. on Image Processing*, 7, (1998) 310-318.
12. N Sochen and Y Y Zeevi, "Images as manifolds embedded in a spatial-feature non-Euclidean space", November 1998, EE-Technion report no. 1181.
13. N Sochen and Y Y Zeevi, "Representation of colored images by manifolds embedded in higher dimensional non-Euclidean space", IEEE ICIP'98, Chicago, 1998.
14. Geometry Driven Diffusion in Computer Vision, Ed. B M ter Haar Romeny, Kluwer Academic Publishers, 1994.
15. L Rudin and S Osher and E Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D*, 60 (1991) 259-268.
16. A Yezzi, "Modified curvature motion for image smoothing and enhancement", *IEEE Trans. on Image Processing*, 7 (1998) 345-352 .