SIAM J. APPL. MATH.
Vol. 64, No. 5, pp. 1477–1508

STEREOGRAPHIC COMBING A PORCUPINE OR STUDIES ON DIRECTION DIFFUSION IN IMAGE PROCESSING*

NIR A. SOCHEN†, CHEN SAGIV†, AND RON KIMMEL‡

Abstract. This paper addresses the problem of feature enhancement in noisy images when the feature is known to be constrained to a manifold. As an example, we approach the direction denoising problem in a general dimension via the geometric Beltrami framework for image processing. The spatial-direction space is a fiber bundle in which the spatial part is the base manifold and the direction space is the fiber. The feature (direction) field is represented accordingly as a section of the spatial-feature fiber bundle. The resulting Beltrami flow is a selective smoothing process that respects the bundle's structure, i.e., the feature constraint. Direction diffusion is treated as a canonical example of a non-Euclidean feature space. The structures of the fiber spaces of interest in this paper are the unit circle S^1, the unit sphere S^2, and the unit hypersphere S^n. Applications to color analysis are discussed, and numerical experiments demonstrate again the benefits of the Beltrami framework in comparison to other feature enhancement schemes for nontrivial geometries in image processing.

Key words. anisotropic diffusion, constrained optimization, orientation diffusion, Beltrami framework

AMS subject classifications. 58J35, 58J90

DOI. 10.1137/S0036139902415518

1. Introduction.

Many objects of low-level vision are vector fields of various types. This is the case for gray-value images, color images, movies, 3D (three-dimensional) volumetric images, and disparity in stereo vision, to name just a few examples. These vector fields are traditionally considered as taking values in $\mathbb{R}^n$. Operations on these fields such as denoising, enhancement, sharpening, and segmentation are done using a variety of algorithms. Several types of vector fields are constrained in a nontrivial way. When the constraint can be expressed via the vanishing of a smooth function, e.g., a polynomial, the vector fields take their values in a non-Euclidean space. One notable example is the direction vector field which assigns a local direction to each pixel in the image. These directions are unit length vectors that span the unit n-dimensional sphere S^n. Other classes of non-Euclidean vector fields are perceptually treated color images [20] and the regularization of frames [23]. We study in this paper the n-dimensional direction vector fields and spherically constrained color models via the Beltrami framework [19].

The basic objects in the Beltrami framework are embedding maps of Riemannian manifolds. These maps embed the image manifold (a surface for a 2D image) in a fiber bundle whose base is the spatial manifold, e.g., $\mathbb{R}^2$, and the fiber is the feature manifold, e.g., $\mathbb{R}^1$, for the intensity feature alone. If we denote by F the feature manifold and assume that the image is given on a flat surface, then the spatial-feature manifold M is given locally as $M = \mathbb{R}^2 \otimes F$. In all the examples below, the

*Received by the editors October 2, 2002; accepted for publication (in revised form) November 26, 2003; published electronically June 22, 2004. We acknowledge grants from the Israel Academy of Science, Israel Ministry of Science, Research fund of the University of Tel-Aviv, and the Adamas super-center for brain research.

http://www.siam.org/journals/siap/64-5/41551.html

†Department of Applied Mathematics, University of Tel Aviv, Ramat-Aviv, Tel-Aviv 69978, Israel (sochen@math.tau.ac.il, chensagi@post.tau.ac.il).

‡Department of Computer Science, Technion - Israel Institute of Technology, Technion City, Haifa 32000, Israel (ron@cs.technion.ac.il).

fiber bundle is trivial, yet our local treatment extends to nontrivial bundles as well. Global issues of nontrivial fiber bundles are beyond the scope of this paper.

Another important ingredient of the Beltrami framework is a geometrical functional, known as the Polyakov action (or harmonic energy [1]), which is defined over the space of embedding maps. The minimization of the Polyakov action is done by an Euler–Lagrange operator that drives, through a gradient descent equation, the initial noisy feature vector fields towards a minimum of the Polyakov action. The special form of this functional favors piecewise smooth images. Jumps in the feature space (feature edges) are consequently preserved [4, 5].

Almost all works that try to minimize a functional with respect to a constraint quantity embed the constrained feature in a higher-dimensional Euclidean space and perform the minimization for the coordinates of this unconstrained space. The common wisdom is to combine a minimization of an unconstrained function and a projection on the constraint variety/manifold. The treatment of direction diffusion was recently addressed along these lines in the low-level vision community. These studies follow the well established literature in the liquid crystal community [3]. The harmonic energy functional and its minimization are subjects to intensive mathematical study as well [6, 7]. Two approaches for this problem are known: in a paper that first directly addresses this issue, Perona [13] uses a single parameter θ as an internal coordinate in S^1. The second approach [21, 22, 2] embeds the unit circle S^1 in $\mathbb{R}^2$ (the sphere S^2 in $\mathbb{R}^3$) and works with the external coordinates; see also [24] for a related effort. The first approach is problematic because of the periodicity of S^1. Averaging small angles around zero such as $\theta = \epsilon$ and $\theta = 2\pi - \epsilon$ leads to the erroneous conclusion that the average angle is $\theta = \pi$. Perona solved this problem by exponentiating the angle so that $V = e^{i\theta}$. This is actually the embedding of S^1 in $\mathbb{C}$ which is isometric to $\mathbb{R}^2$. This method is specific to a 2D embedding space where complex numbers can be used. The problem in using only one internal coordinate manifests itself in the numerical implementation of the PDE through the breaking of rotation invariance. In the second approach we have to make sure that we always stay on S^1 along the flow. This problem is known as the projection problem. It is solved in the continuum by adding a projection term. Tang, Sapiro, and Caselles [21, 22] propose the formalism of p-harmonic maps applied to the case of direction and color diffusion, and present experiments in the case $p = 2$, which corresponds to the Dirichlet integral. Moreover, they also present experiments for the case $p = 1$ as the immediate extension of the Rudin–Osher–Fatemi total variation (TV) denoising algorithm [14] to the case of general maps with values on manifolds. Nevertheless, they did not study in detail the algorithm for the $p = 1$ case. The algorithmic study for the case $p = 1$ was done by Chan and Shen [2], who also use external coordinates with a projection term and a TV measure in order to better preserve discontinuities in the vector field. This works well for the case where the codimension is one, like a circle. Yet it is difficult to generalize this approach to higher codimensions like the sphere. Moreover, the flow of the external coordinates is difficult to control numerically since numerical errors should be projected onto S^1 and since no well-defined projection exists. Recently an implicit way to define manifolds has been used in this context [1]. We concentrate in this paper on the explicit methods. A comparison between the implicit harmonic energy method and the implicit Beltrami framework can be found in [16].

We propose to work directly on the constrained manifold and to avoid the projection problem altogether. Our solution produces an adaptive smoothing process, which preserves direction discontinuities. The proposed solution works for all dimensions and codimensions, and overcomes possible parameterization singularities by introducing

several internal coordinates on different patches (charts) such that the union of the patches is the feature manifold, i.e., S^n. Adaptive smoothness is achieved by the description of the vector field as a 2D section of the $(n+2)$-dimensional spatial-feature fiber bundle manifold with S^n fibers.

The problem is formulated, in the Beltrami framework [19, 9], in terms of the embedding map

$$Y : (\Sigma, g) \to (M, h),$$

where Σ is the 2D image manifold and M, in this case, is $\mathbb{R}^n \otimes S^1$ with $n = 2$ ($n = 4$) for gray-level (color) images. The key point is the choice of *local coordinate systems* for *both* manifolds,[1] the image manifold Σ (with metric g) and the embedding manifold M (with metric h). At the same time we should verify that the geometric filter (i.e., the denoising PDE) does not depend on the specific choice of coordinates we make.

Once a local coordinate system is chosen for the embedding space and the optimization is done directly in this local coordinate system, we can never leave M and avoid the problem of projection. The difficulty represented in the problem of projection is transformed into the problem of the choice of a local coordinate system, as we describe below. Other examples of enhancement by the Beltrami framework of nonflat feature spaces, like the color perceptual space and the derivatives vector field, can be found in [20, 17].

An important issue in this approach is the numerical consideration in the choice of local coordinates. While all coordinates are equally good from analytic and geometric points of view, they are different from a numeric standpoint. A comparative study on the numerical and algorithmic accuracies of different schemes is presented here and shows that, for a range of parameters, one can get a better numerical accuracy while maintaining the edge preserving quality of the anisotropic diffusions.

This paper is organized as follows. We review the Beltrami framework and point to the relation with harmonic maps in section 2. We analyze the case of the general n-dimensional direction diffusion with hemispheric coordinate system in section 3. A stereographic coordinate system is introduced in section 4, and the appropriate equations are derived. Section 5 deals with the numerical implementation of the ideas presented in the previous sections for color image processing. Section 6 presents results on various vector fields and color images. We compare in section 7 different direction diffusion schemes from numerical and algorithmic points of view. We summarize and conclude in section 8.

2. The Beltrami framework. Let us briefly review the Beltrami geometric framework for nonlinear diffusion in computer vision [19].

2.1. Representation and Riemannian structure. An image, and many other quantities of interest in computer vision, are naturally represented via the concept of a fiber bundle. The image domain is the base manifold. In the present study it is taken as a subset of $\mathbb{R}^2$ with the canonical Cartesian coordinate system (Y^1, Y^2). It is denoted by Ω. At each point in the base manifold we attach a feature space—the fiber. The fibers at different points of the base manifold are isomorphic. The fiber space is denoted by F. The feature space, or fiber, may be a linear vector space or more interestingly a Riemannian manifold. An image (or other quantity of interest)

[1] Note the difference between this approach and the one presented in [21, 22, 2], where the image metric is flat.

is a choice of a particular point in the fiber for every point in the base manifold. Such a particular choice is called *a section* of the (trivial) fiber bundle $\Omega \otimes F$.

In general an n-dimensional (Riemannian) manifold is defined by a collection of maps from charts of the manifold to $\mathbb{R}^n$. Each chart covers part of the manifold. Their union covers the whole manifold, and the transformation of the coordinates on the intersection between any two charts is smooth. The Riemannian structure transforms in a proper way (as a tensor) under any change of the coordinate system. We denote the coordinates on the 2D section by (x^1, x^2), the coordinates on a chart of the embedding space (the fiber bundle) by $(Y^1, \ldots, Y^n)$. The embedding space is a hybrid spatial-feature space. The first two coordinates (Y^1, Y^2) are the spatial coordinates on Ω (the base manifold), and the rest $(Y^3, \ldots, Y^n)$ are the feature coordinates (the fiber's coordinates). The simplest example is a gray-value image which is represented as a 2D surface embedded in $\mathbb{R}^3$. We denote the map by $Y : \Sigma \to \mathbb{R}^3$, where Σ is a 2D section. The map Y is given in our example by $(Y^1 = x^1, Y^2 = x^2, Y^3 = I(x^1, x^2))$. We choose on this surface a Riemannian structure, namely a metric. Note that this differs from the harmonic energy functional, where the metric is taken from the base manifold and not from the section. The metric is a positive definite and a symmetric 2-tensor that may be defined through the local distance measurements

$$ds^2 = g_{11}(dx^1)^2 + 2g_{12}dx^1dx^2 + g_{22}(dx^2)^2. \tag{2.1}$$

We use the Einstein summation convention in which the above equation reads as $ds^2 = g_{\mu\nu}dx^\mu dx^\nu$, where repeated indices are summed over. We denote the inverse of the metric by $g^{\mu\nu}$.

2.2. Image metric selection: The induced metric. A reasonable assumption is that distances measured in the embedding spatial-feature fiber bundle, such as distances between pixels and differences between gray-levels, correspond directly to distances measured on the image manifold, i.e., the section. This is the assumption of isometric embedding under which we can calculate the image metric in terms of the embedding maps Y^i and the embedding space metric h_{ij}. This follows directly from the fact that the length of infinitesimal distances on the manifold can be calculated on the manifold and on the embedding space with the same result. Formally, $ds^2 = g_{\mu\nu}dx^\mu dx^\nu = h_{ij}dY^i dY^j$. By the chain rule, $dY^i = \partial_\mu Y^i dx^\mu$, we get $ds^2 = g_{\mu\nu}dx^\mu dx^\nu = h_{ij}\partial_\mu Y^i \partial_\nu Y^i dx^\mu dx^\nu$, from which we have

$$g_{\mu\nu} = h_{ij}\partial_\mu Y^i \partial_\nu Y^j. \tag{2.2}$$

As an example we take the gray-level image as a 2D image manifold embedded in the 3D Euclidean space $\mathbb{R}^3$. The embedding maps are

$$(Y^1(x^1, x^2) = x^1,\ Y^2(x^1, x^2) = x^2,\ Y^3(x^1, x^2) = \beta I(x^1, x^2)). \tag{2.3}$$

The scaling factor β defines the ratio between distances in gray-values and distances in the spatial space. It is a free parameter of the framework that interpolates between the Euclidean L_2 and L_1 types of flows, as we will see below. We choose to parameterize the image manifold by the canonical coordinate system $x^1 = x$ and $x^2 = y$. The embedding, by abuse of notation, is $(x, y, \beta I(x, y))$. The induced metric element g_{11} is calculated as follows:

$$g_{11} = h_{ij}\partial_{x^1} Y^i \partial_{x^1} Y^j = \delta_{ij}\partial_x Y^i \partial_x Y^j = \partial_x x \partial_x x + \partial_x y \partial_x y + \partial_x \beta I \partial_x \beta I = 1 + \beta^2 I_x^2. \tag{2.4}$$

Other elements are calculated in the same manner. The result is

$$G = (g_{\mu\nu}) = \begin{pmatrix} 1+\beta^2 I_x^2 & \beta^2 I_x I_y \\ \beta^2 I_x I_y & 1+\beta^2 I_y^2 \end{pmatrix}. \tag{2.5}$$

2.3. Polyakov action: A measure on the space of embedding maps. Denote by (Σ, g) the image manifold and its metric, and by (M, h) the space-feature manifold and its metric. Then the functional $S[\cdot,\cdot,\cdot]$ attaches a real number to a map $Y : \Sigma \to M$,

$$S[Y^i, g_{\mu\nu}, h_{ij}] = \int dV ||\nabla \vec{Y}||^2_{h,g}, \tag{2.6}$$

where dV is a volume element and the integration is over the Riemannian Frobenius norm[2] of the tangent map dY. In a local coordinate system the volume element is expressed by $dV = dx^1 dx^2 \sqrt{g}$ and $||\nabla \vec{Y}||^2_{h,g} = \langle \nabla Y^i, \nabla Y^j \rangle_g h_{ij} = g^{\mu\nu} \partial_\mu Y^i \partial_\nu Y^j h_{ij}$. The Polyakov action is expressed in this local system of coordinates as

$$S[Y^i, g_{\mu\nu}, h_{ij}] = \int dx^1 dx^2 \sqrt{g} g^{\mu\nu} \partial_\mu Y^i \partial_\nu Y^j h_{ij}. \tag{2.7}$$

This functional, for $m = 2$ (a 2D image manifold) and $h_{ij} = \delta_{ij}$, was proposed by Polyakov [12] in the context of high energy physics and the theory known as *string theory*. It is important to note that the image metric and the feature coordinates—i.e., intensity, color, direction, etc.—are independent variables. This functional is the natural generalization of the L_2 norm from Euclidean domains to Riemannian manifolds. The minimization of the functional with respect to the image metric can be solved analytically in the 2D case (see, for example, [18]). The minimizer is the induced metric. If we choose, a priori, the image metric induced from the metric of the embedding spatial-feature space M, then the Polyakov action is reduced to the area (volume) of the image manifold:

$$S[Y^i, h_{ij}] = 2\int dV = 2\int dx^1 dx^2 \sqrt{g} = 2\int dx^1 dx^2 \sqrt{\det(\partial_\mu Y^i \partial_\nu Y^j h_{ij})}. \tag{2.8}$$

This follows from the form of the induced metric,

$$\langle \nabla Y^i, \nabla Y^j \rangle_g h_{ij} = g^{\mu\nu} \partial_\mu Y^i \partial_\nu Y^j h_{ij} = g^{\mu\nu} g_{\mu\nu}$$

and the identity

$$g^{\mu\nu} g_{\mu\nu} = \mathrm{Tr}(G^{-1} G^T) = \mathrm{Tr}(G^{-1} G) = \mathrm{Tr}(\mathrm{Id}) = 2, \tag{2.9}$$

where $\mathrm{Tr}(X)$ denotes the trace of the matrix X.

Using standard methods in the calculus of variations (see [18]), the Euler–Lagrange equations with respect to the embedding are

$$-\frac{1}{2\sqrt{g}} h^{il} \frac{\delta S}{\delta Y^l} = \frac{1}{\sqrt{g}} \partial_\mu (\sqrt{g} g^{\mu\nu} \partial_\nu Y^i) + \Gamma^i_{jk} \langle \nabla Y^j, \nabla Y^k \rangle_g. \tag{2.10}$$

Since $(g_{\mu\nu})$ is positive definite, $g \equiv \det(g_{\mu\nu}) > 0$ for all x^μ. This factor is the simplest one that does not change the minimization solution while giving a reparameterization

[2]By Riemannian Frobenius norm we mean that the square of the elements is with respect to the Riemannian structures of the corresponding Riemannian manifolds.

invariant expression. The operator that is acting on Y^i in the first term is the natural generalization of the Laplacian from flat spaces to manifolds and is called *the second order differential operator of Beltrami* [10], or the *Beltrami operator*, and is denoted by Δ_g. The second term involves the Levi–Civita connection whose coefficients are the Christoffel symbols. The coefficients are given in terms of the metric of the embedding space

$$\Gamma^i_{jk} = \frac{1}{2} h^{il} \left(\partial_j h_{lk} + \partial_k h_{jl} - \partial_l h_{jk} \right). \tag{2.11}$$

This is the term that takes into account the fact that the image surface flows in a non-Euclidean manifold and not in $\mathbb{R}^n$.

A map that satisfies the Euler–Lagrange equations $-\frac{1}{2\sqrt{g}} h^{il} \frac{\delta S}{\delta Y^l} = 0$ is a *harmonic map*. The 1D and 2D examples are a geodesic curve on a manifold and a minimal surface.

The nonlinear diffusion or scale-space equation emerges as the gradient descent minimization flow

$$Y^i_t = \frac{\partial}{\partial t} Y^i = -\frac{1}{2\sqrt{g}} h^{il} \frac{\delta S}{\delta Y^l} = \Delta_g Y^i + \Gamma^i_{jk} \langle \nabla Y^j, \nabla Y^k \rangle_g. \tag{2.12}$$

This flow evolves a given surface towards a minimal surface, and in general it continuously changes a map towards a harmonic map.

Before closing this review of the Beltrami framework, we would like to point out a few similarities and differences between this flow and those suggested in [14, 13, 21, 2]:

1. For flat fibers:
   - We use the induced metric, while in other flows the image metric is flat. The difference comes from the fact that in our framework the image manifold is a *section* of the fiber bundle, while in the harmonic map formulation it is the *base* manifold.
   - In the case of flat and 1D fibers we get the "regularized total variation" functional. In the limit of large β the evolution equation is identical (up to $\sqrt{g}$) to the TV one. In the limit $\beta \to 0$ we get the linear diffusion case. In intermediate values we find a good compromise such that over-smoothing, on the one hand, and stair-casing, on the other hand, can be avoided. The Beltrami framework, in this case, is a one-parameter generalization of the TV scheme.
   - The multichannel functional, in the Beltrami framework, is another generalization of the TV functional. A term that depends on the direction of the gradients is added to the term that depends on their magnitude only. This provides a better adaptation of the process to the image features.
   - The Beltrami flow is degenerate (at $\nabla I \to \infty$). One can prove that discontinuities are preserved for a finite time [5].
2. For nonflat fibers:
   - The coordinates Y^i are the local coordinates of the feature space, while in the above-mentioned flows they are coordinates of a *third* manifold, i.e., $\mathbb{R}^{n+1}$, in which the feature space S^n is embedded. In other words, the fiber in the harmonic map approach is embedded in $\mathbb{R}^{n+1}$. This is not possible in general (see the Nash embedding theorem [11]).
   - The Polyakov functional is different in this case from the TV functional due to the different weighting of the magnitude of the gradients.

- The flow equation (2.12) has a clear geometric meaning. It is a mean curvature flow projected (analytically) on the fiber. This projection is an edge preserving operation [19]. It depends on ∇I in the general multichannel case and not on $|\nabla I|$ as in the harmonic map approach.

3. Hemispheric direction diffusion.

3.1. Fiber geometry. We are interested in the case where the fiber feature space is the hypersurface S^n. We choose to represent the hypersphere S^n as an n-dimensional manifold embedded in $\mathbb{R}^{n+1}$, with Cartesian coordinate system $\{U^i\}_{i=3}^{n+3}$, as the constrained hypersurface

$$\sum_{i=3}^{n+3}(U^i)^2 = 1. \tag{3.1}$$

We work in the chart, where $\{Y^i\}_{i=3}^{n+2}$ are local coordinates. On this chart, $U^i = Y^i$, $i = 3, \ldots, n+2$, and

$$U^{n+3} = \sqrt{1 - \sum_{i=3}^{n+2}(Y^i)^2}.$$

Denote the metric elements for the feature space only by $\tilde{h}_{ij}$. The metric elements and the inverse metric elements are given by

$$\tilde{h}_{ij} = \delta^{ij} + \frac{Y^i Y^j}{1 - \sum_{k=3}^{n+2}(Y^k)^2},$$

$$(\tilde{h}^{-1})_{ij} = \delta^{ij} - Y^i Y^j. \tag{3.2}$$

3.2. The induced metric. The induced metric and its inverse are accordingly

$$\begin{aligned} g_{\mu\nu} &= \delta_{\mu\nu} + \sum_{i,j=3}^{n+2} \tilde{h}_{ij}\partial_\mu Y^i \partial_\nu Y^j, \\ g^{\mu\nu} &= \frac{1}{g}\left(\delta^{\mu\nu} + \epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j=3}^{n+3}\tilde{h}_{ij}\partial_\sigma Y^i \partial_\rho Y^j\right), \\ g &= \det(g_{\mu\nu}), \\ &= 1 + \sum_{i,j=3}^{n+3}\tilde{h}_{ij}(Y_x^i Y_x^j + Y_y^i Y_y^j), \\ &\quad + \frac{1}{2}\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j,k,l=3}^{n+3}\tilde{h}_{ij}\tilde{h}_{kl}\partial_\mu Y^i \partial_\rho Y^j \partial_\nu Y^k \partial_\sigma Y^l, \end{aligned} \tag{3.3}$$

where $(g^{\mu\nu})$ is the inverse of $(g_{\mu\nu})$, g is the determinant, and $\epsilon^{\mu\nu}$ is the 2D antisymmetric tensor

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

An implicit summation on all repeated Greek indices is assumed.

3.3. The flow equations. The Levi–Civita coefficients are calculated in Appendix B with the simple result

$$\Gamma^i_{jk} = Y^i \tilde{h}_{jk}. \tag{3.4}$$

The minimization of the Polyakov action leads to the following evolution equations:

$$Y^i_t = \Delta_g Y^i + 2Y^i - Y^i \mathrm{Tr}(g^{\mu\nu}), \qquad i = 1, \ldots, n. \tag{3.5}$$

3.4. The 1D hemispheric direction diffusion.

3.4.1. The S^1 Beltrami operator. The S^1 manifold can be described as the solution to $U^2 + V^2 = 1$. We will work with two charts. One is ($Y^1 = x$, $Y^2 = y$, $Y^3 = \beta U$), and the other is ($Y^1 = x$, $Y^2 = y$, $Y^3 = \beta V$). By abuse of notation we denote the map by $(x, y, \beta Y)$. The parameter β is a scaling factor. Each one of the charts will be used in the range $Y^2 \leq 1/2$. The line element on each of the charts of the image manifold is

$$ds^2 = ds^2_{\mathbb{R}^2} + ds^2_{S^1} = dx^2 + dy^2 + \frac{\beta^2}{1 - Y^2} dY^2. \tag{3.6}$$

By using the chain rule we find

$$ds^2 = (1 + A(Y)Y_x^2)dx^2 + 2A(Y)Y_xY_y dxdy + (1 + A(Y)Y_y^2)dy^2, \tag{3.7}$$

where $A(Y) = \frac{\beta^2}{1-Y^2}$.

The induced metric is therefore

$$(g_{\mu\nu}) = \begin{pmatrix} 1 + A(Y)Y_x^2 & A(Y)Y_xY_y \\ A(Y)Y_xY_y & 1 + A(Y)Y_y^2 \end{pmatrix}, \tag{3.8}$$

and the Beltrami operator acting on Y is $\Delta_g Y = \frac{1}{\sqrt{g}} \partial_\mu (\sqrt{g} g^{\mu\nu} \partial_\nu Y)$, where $g = 1 + A(Y)(Y_x^2 + Y_y^2)$ is the determinant of $(g_{\mu\nu})$, and $(g^{\mu\nu})$ is the inverse matrix of $(g_{\mu\nu})$.

3.4.2. The Levi–Civita connection. Since the embedding space is non-Euclidean, we have to calculate the Levi–Civita connection. Remember that the metric of the embedding space is

$$(h_{ij}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & A(Y) \end{pmatrix}. \tag{3.9}$$

The Levi–Civita connection coefficients are given by the fundamental theorem of Riemannian geometry in the following formula: $\Gamma^i_{jk} = \frac{1}{2} h^{il} (\partial_j h_{lk} + \partial_k h_{jl} - \partial_l h_{jk})$, where the derivatives are taken with respect to Y^i for $i = 1, 2, 3$.

The only nonvanishing term is Γ^3_{33}, which reads

$$\Gamma^3_{33} = \frac{1}{2A(Y)} \partial_Y (A(Y)) = \frac{Y}{1 - Y^2} = Y h_{33}. \tag{3.10}$$

The second term in the Euler–Lagrange equations in this case reads $Y h_{33} ||\nabla Y||^2_g$. We can rewrite this expression using the following identities:

$$\begin{aligned} h_{33}||\nabla Y||^2_g &= (h_{11}g^{11} + h_{22}g^{22} + h_{33}\partial_\mu Y \partial_\nu Y g^{\mu\nu}) - (h_{11}g^{11} + h_{22}g^{22}) \\ &= g_{\mu\nu}g^{\mu\nu} - (g^{11} + g^{22}) = 2 - \frac{1}{g}(g_{11} + g_{22}) = 2 - \frac{1}{g}(1 + g), \end{aligned} \tag{3.11}$$

where we used the induced metric identity (2.2), and the identity (2.9), in order to rewrite

$$2 = \mathrm{Tr}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = g_{\mu\nu}g^{\mu\nu} = h_{11}g^{11} + h_{22}g^{22} + h_{33}\partial_\mu Y \partial_\nu Y g^{\mu\nu}. \tag{3.12}$$

3.4.3. The flow and the switches. The Beltrami flow is

$$Y^i_t = \Delta_g Y^i + \Gamma^i_{jk}(Y^1, Y^2, Y^3)\langle \nabla Y^j, \nabla Y^k \rangle_g \tag{3.13}$$

for $i = 3$. Only modifying the fiber values while keeping the case manifold constant is a *projection* in the direction of the fiber. This projection slows the diffusion around edges. The Beltrami flow on the two charts reads finally as

$$U_t = \Delta_g U + U\frac{g-1}{g},$$

$$V_t = \Delta_g V + V\frac{g-1}{g}. \tag{3.14}$$

In the implementation we compute the diffusion for U and V simultaneously and take the values $(U, \mathrm{sign}(V)\sqrt{1-U^2})$ for the range $U^2 \le V^2$, and $(\mathrm{sign}(U)\sqrt{1-V^2}, V)$ for the range $V^2 \le U^2$.

4. Stereographic direction diffusion.

4.1. Fiber geometry. The hemispheric parameterization requires more charts as n increases. As a result we have to work closer and closer to the singularity. As a cure for that we switch to stereographic parameterization, which demands only two charts independent of the dimension of the sphere. Moreover, we always work on the furthest point from the singularity, that is, on the equator.

Every hypersphere S^n can be isometrically embedded in $\mathbb{R}^{n+1}$. The hypersphere is realized as the place of all the points in $\mathbb{R}^{n+1}$ that satisfy the constraint $\sum_{i=1}^{n+1} U^i U^i = 1$. We denote by Y^i for $i = 1, \ldots, n$ the Cartesian coordinate system on the subspace $\mathbb{R}^n$ that passes through the equator of S^n, i.e., $\{\vec{U} \in \mathbb{R}^{n+1} | U^{n+1} = 0\}$. The stereographic transformation gives the values of Y^i as functions of the points on the north (south) hemispheres of the hypersphere. Explicitly it is given (after shifting the indices by two for notation consistent with the next sections) as

$$Y^i = \frac{U^i}{1 - U^{n+3}}, \quad i = 3, \ldots, n+2.$$

Inverting these relations, we find

$$U^i = \frac{2Y^i}{1 + \sum_{i=1}^n Y^i}, \quad i = 3, \ldots, n+2,$$

$$U^{n+3} = \frac{-1 + \sum_{i=3}^{n+2} Y^i}{1 + \sum_{i=3}^{n+2} Y^i}. \tag{4.1}$$

4.2. The induced metric. Now we can compute the induced metric of our feature space

$$h_{ij} = \sum_{k=3}^{n+3} \frac{\partial U^k \partial U^k}{\partial Y^i \partial Y^j} = \frac{4}{(1+A)^2}\delta_{ij}, \quad i, j = 3, \ldots, n+2, \tag{4.2}$$

where $A = \sum_{k=3}^{n+2} (Y^k)^2$.

4.3. The flow equations. The Levi–Civita connection can be obtained using (2.11) and (4.2). The result is

$$\Gamma^i_{jk} = \frac{4}{1+A}\left(Y^i\delta_{jk} - Y^k\delta_{ij} - Y^j\delta_{ki}\right).$$

The resulting diffusion equations are

$$Y^i_t = \Delta_g Y^i + \sum_{jk} \frac{4}{1+A}\left(Y^i\delta_{jk} - Y^k\delta_{ij} - Y^j\delta_{ki}\right)\partial_\mu Y^j \partial_\nu Y^k g^{\mu\nu}, \tag{4.3}$$

where $i = 3, \ldots, n+2$. This can be rearranged to

$$Y^i_t = \Delta_g Y^i - 4g^{\mu\nu}(\partial_\mu \log(1+A))(\partial_\nu Y^i) + (1+A)(2 - g^{11} - g^{22})Y^i. \tag{4.4}$$

4.4. 1D and 2D directions. We denote our coordinate system by the subscripts s (for south) and n (for north). The equations for the 1D case read

$$(Y_s)_t = \Delta_g Y_s - 4g^{\mu\nu}(\partial_\mu \log(1+A))(\partial_\nu Y_s) + (1+A)(2 - g^{11} - g^{22})Y_s, \tag{4.5}$$

where $A = Y_s^2$ and the induced metric is a function of Y_s. A parallel equation is written for Y_n. We solve the north and south equations simultaneously for values *smaller* than 1. At each iteration we update the values which are greater than 1 by the simple relation $Y_s = 1/Y_n$. Note that the problematic zone(s), i.e., ± 1, are as far as possible from the singularities, i.e., the poles.

The 2D case is managed similarly via

$$\begin{aligned}(Y^1_s)_t &= \Delta_g Y^1_s - 4g^{\mu\nu}(\partial_\mu \log(1+A_s))(\partial_\nu Y^1_s) + (1+A_s)(2 - g^{11} - g^{22})Y^1_s,\\ (Y^2_s)_t &= \Delta_g Y^2_s - 4g^{\mu\nu}(\partial_\mu \log(1+A_s))(\partial_\nu Y^2_s) + (1+A_s)(2 - g^{11} - g^{22})Y^2_s,\end{aligned} \tag{4.6}$$

where $A_s = (Y^1_s)^2 + (Y^2_s)^2$ and the induced metric depends on Y^1_s and Y^2_s. As in the 1D case, we solve simultaneously for the south and north patches and work with Y^i's which are smaller than 1. The update for values that are greater than 1 after the diffusion (in each iteration) is done by $Y^i_s = A_s Y^i_n$. Again the decision zone, i.e., the equator, is the most numerically stable region since it is the furthest from the poles, where singularities may appear.

5. Color diffusion. There are many coordinate systems and models of color space which try to be as close as possible to human color perception. One of the popular coordinate systems is the HSV system [15]. In this system, color is characterized by hue, saturation, and value. The saturation and value take their values in $\mathbb{R}^+$, while the hue is an angle that parameterizes S^1.

In order to denoise and enhance color images by a nonlinear diffusion process which is more adapted to human perception, we use here the HSV system. We need a special treatment of the hue coordinate in section 3.

Let us represent the image as a mapping $\mathbf{Y} : \Sigma \to \mathbb{R}^4 \times S^1$, where Σ is the 2D image surface and $\mathbb{R}^4 \times S^1$ is parameterized by the coordinates (x, y, H, S, V). As mentioned above, a diffusion process in this coordinate system is problematic. We define therefore two coordinates,

$$U = \cos H \qquad \text{and} \qquad W = \sin H,$$

and continue in a way similar to section 3. The metric of $\mathbb{R}^4 \times S^1$ on the patch where U parameterizes S^1 and $W(U)$ is nonsingular is

$$h_{ij} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & A(U) & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{5.1}$$

where $A(U) = 1/(1 - U^2)$.

The induced metric is therefore

$$\begin{aligned} ds^2 &= dx^2 + dy^2 + A(U)dU^2 + dS^2 + dV^2 \\ &= dx^2 + dy^2 + A(U)(U_x dx + U_y dy)^2 + (S_x dx + S_y dy)^2 + (V_x dx + V_y dy)^2 \\ &= (1 + A(U)U_x^2 + S_x^2 + V_x^2)dx^2 \\ &\quad + 2(A(U)U_x U_y + S_x S_y + V_x V_y)dxdy + (1 + A(U)U_y^2 + S_y^2 + V_y^2)dy^2. \end{aligned} \tag{5.2}$$

Similar expressions are obtained on the other dual patch.

The only nonvanishing Levi–Civita connection coefficient is $\Gamma^3_{33} = U h_{33}$. The resulting flow is

$$\begin{aligned} U_t &= \Delta_g U + 2U - U(g^{11} + g^{22}), \\ W_t &= \Delta_g W + 2W - W(g^{11} + g^{22}), \\ S_t &= \Delta_g S, \\ V_t &= \Delta_g V. \end{aligned} \tag{5.3}$$

Note that the switch between U and W should be applied not only to the U and W equations but also to the S and V evolution equations where, at each point, one needs to work with the metric that is defined on one of the patches.

6. Experimental results. Our first example deals with the gradient direction flow via the Beltrami framework. Figure 6.1 shows a vector field before and after the application of the flow for a given evolution time. The normalized gradient vector field extracted from the image is presented before and after the flow and shows the way the field flows into a new smooth direction transactions field.

Our second example deals with color diffusion using different color spaces. We use machine color space as our spectral model, where we first restrict the colors to one quarter of the upper hemisphere defined around the black point in the RGB space, as shown in Figure 6.2. In this example we use the hemispheric direction diffusion. The intensity, or more accurately the magnitude, is handled separately. This is a simple example since a single chart can be used as a parameterization, and indeed this simplified version was often used by others as an example.

Next, we explore a popular model that captures some of our color perception. The HSV (hue, saturation, value) model proposed in [15] is often used as a "user-oriented" color model, rather than the RGB "machine-oriented" model.

Figure 6.3 shows the classical representation of the HSV color space, in which the hue is measured as an angle, while the value (sometimes referred to as brightness) and the color saturation are mapped onto finite nonperiodic intervals. This model lands itself into a filter that operates on the spatial x, y coordinates, the value and saturation coordinates, and the hue periodic variable. Our image is now embedded in $\mathbb{R}^4 \times S^1$.

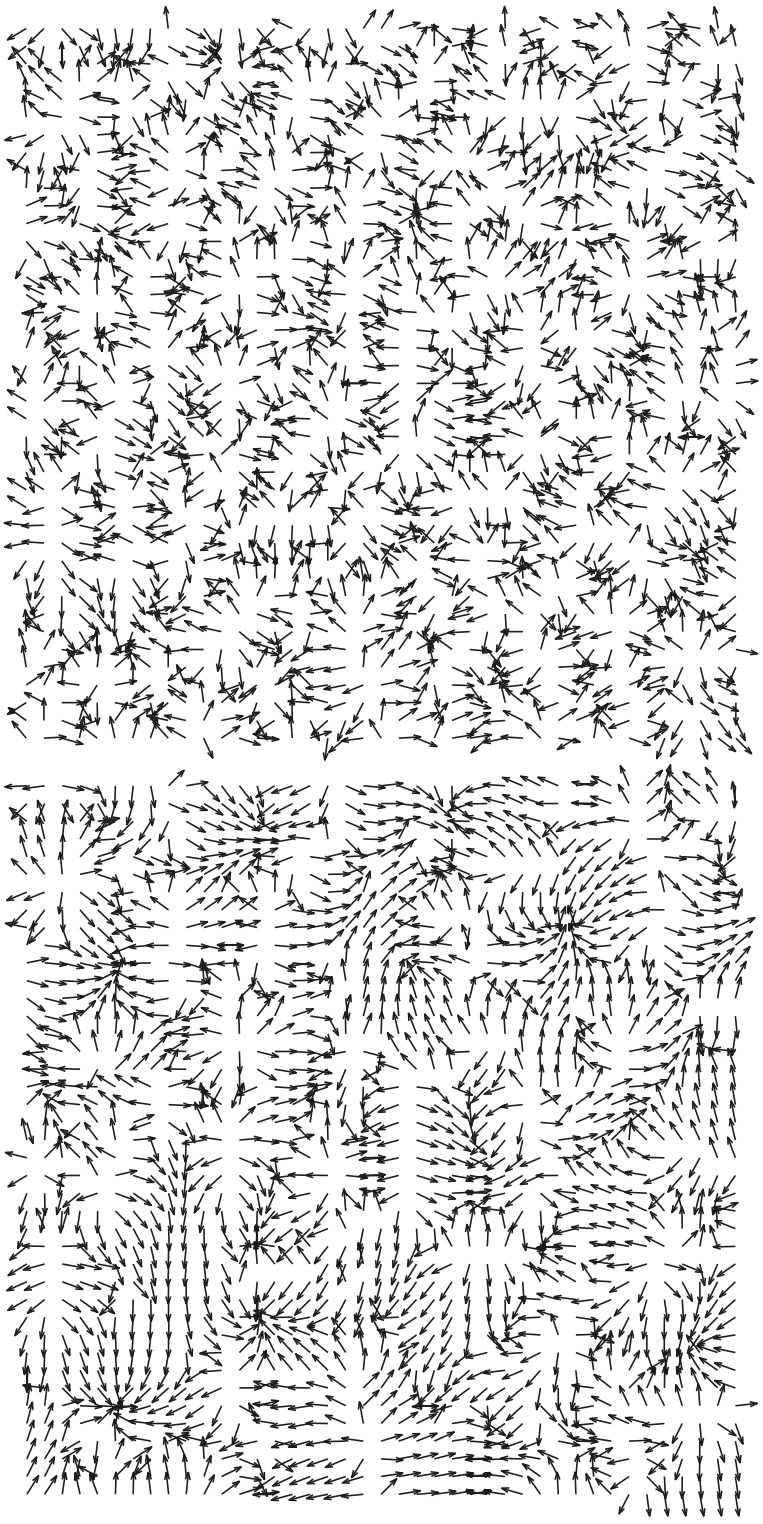

FIG. 6.1. *Two vector fields before (upper) and after (lower) the flow on* S^1.

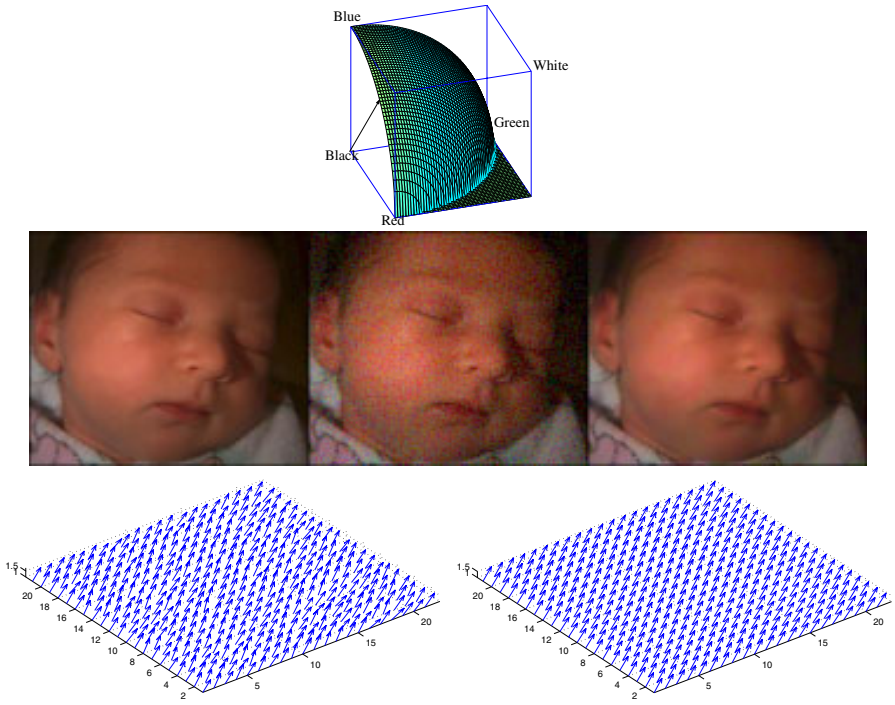

FIG. 6.2. *The colors are restricted to one quarter of the upper hemisphere defined around the black point in the RGB space.*

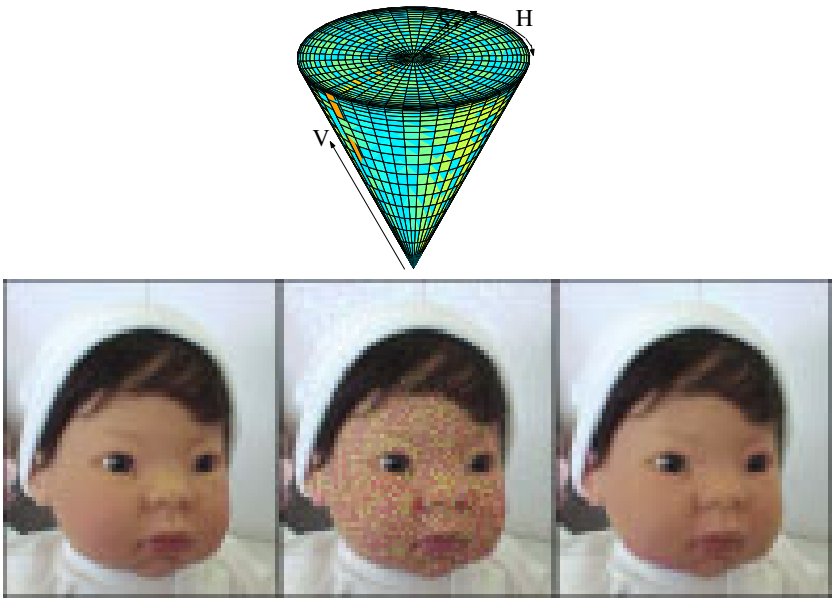

FIG. 6.3. *The HSV color model captures human color perception better than the RGB model which is the common way our machines represent colors. The original image (left), the noisy image (middle), and the filtered image (right) demonstrate the effect of the flow as a denoising filter in the HSV color space when using hemispheric coordinates.*

FIG. 6.4. *An example of stereographic direction diffusion used in the HSV color space. The original image (left), the noisy image (middle), and the filtered image (right) demonstrate the effect of the flow as a denoising filter in the HSV color space when using stereographic coordinates.*

We use the hemispheric direction diffusion for the results shown in Figure 6.3 and the stereographic direction diffusion for the results shown in Figure 6.4. For the complete set of full-size color images see http://www.math.tau.ac.il/∼sochen/Porcupine/porcupine.html.

7. Comparison to other schemes. Several schemes have been suggested to handle direction diffusion. The first to directly address this issue was Perona [13], who uses a single parameter θ as an internal coordinate. However, the periodicity of S^1 leads to erroneous values of θ. Another approach, the linear approach, was offered by Tang, Sapiro, and Caselles [21], in which the unit circle S^1 is embedded in R^2 and external coordinates are used. However, in this flow we have to actively keep our coordinates on S^1, which means that we have to project the results on the unit circle. Chan and Shen [2] studied in detail another scheme in which the evolution equation is derived according to the TV measure.

Kimmel and Sochen [8] have proposed an adaptive hemispheric smoothing scheme, which is edge preserving, based on the Beltrami framework [19]. Throughout this section this scheme is referred to as HP (hemispheric porcupine). The direction vector field is described as a 2D manifold embedded in a higher-dimensional space $M = R^2 \times S^1$. The key point in the HP scheme is the selection of local coordinate systems on the manifold, so that their union is S^1. On the other hand, the local coordinates selection is done so that the numerical error is minimized. The advantage of this scheme is that throughout the flow the coordinates are constrained to S^1. Thus, there is no need for a supplementary projection stage. We address in this work the issue of selecting the right charts to cover S^1, and an alternative stereographic coordinate system is proposed. In this paper we refer to this scheme as SP (stereographic porcupine).

In this study we compare the numerical behavior of the above-mentioned schemes, evaluate their algorithmic performance, and examine their edge preserving quality.

7.1. The evolution equations. In this subsection we mention the evolution equations for each scheme. The interested reader is referred to the original articles.

As a first step, the direction θ is embedded in R^2 via the map $\theta \to \omega = [\cos(\theta), \sin(\theta)]$. The plane is then diffused for some time t, and the result is projected back to the unit circle via the map $\omega_t = [x, y] \to \arctan(\frac{y}{x})$. This is if (x, y) is still a one unit vector. If not, then the phase of the vector is used to determine the appropriate projection; see Figure 7.1.

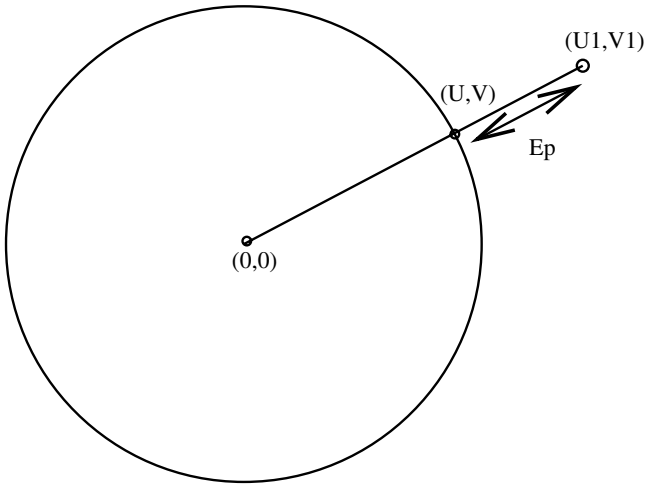

FIG. 7.1. *The projection error in the linear and TV schemes.*

Tang, Sapiro, and Caselles [21] use the following flow for an L_2 energy (which results in a linear scheme),

$$f_t = \Delta f + \| \bigtriangledown f \|^2 f, \tag{7.1}$$

where f stands for the pair (U, V).

Chan and Shen [2] use the following flow for the TV energy:

$$f_t = \operatorname{div}\left(\frac{\bigtriangledown f}{\|f\|}\right) + \| \bigtriangledown f \| f, \tag{7.2}$$

where f stands for the pair (U, V).

Kimmel and Sochen [8] use the following equation for the HP scheme:

$$u_t = \Delta_g U + U \cdot \frac{g-1}{g}, \tag{7.3}$$

$$v_t = \Delta_g V + V \cdot \frac{g-1}{g}, \tag{7.4}$$

where $g = 1 + A(U)((U_x)^2 + (U_y)^2)$ and $A(U) = \frac{1}{1-U^2}$. The SP scheme is given by the following equation:

$$Z_t = \Delta_g Z - 4g^{\mu\nu}(\partial_\mu log(1+A))(\partial_\nu Z) + (1+A)(2 - g^{11} - g^{22})Z, \tag{7.5}$$

where a stereographic coordinate system is used. Here $A = Z^2$, and Z stands for both north and south coordinates.

We remark that in the HP and SP schemes, according to the Beltrami framework, images are considered as surfaces rather than functions. The related diffusion scheme minimizes the area of the image surface. Thus, a basic concept in the Beltrami framework is the manifold's metric. In order to construct a valuable geometric measure for a direction image we have to combine the spatial coordinates with the direction

information. The simplest combination is done by introducing a scaling parameter β, so that

$$ds^2 = dx^2 + dy^2 + \beta^2 \frac{1}{1-U^2} dU^2.$$

The parameter β has dimensions $[\frac{distance}{direction}]$, and it fixes the relative scale between the size of direction information and spatial distances. The parameter β plays an important role in this study. It is a measure of the degree of coupling between the different channels in the diffusion flow. Higher values of β draw the scheme to a behavior similar to that of the TV scheme [2], and smaller values of β cause a behavior similar to that of the linear scheme [21].

Therefore, we expect both HP and SP schemes to have a numerical error and an edge preserving quality which depend on this parameter β.

7.2. Evaluation of the direction diffusion schemes. The evaluation of the different schemes offered for direction diffusion is based on two main attributes of these schemes. The first is their numerical and algorithmic accuracy, which is presented by their degree of error. The second is the edge preserving quality of the scheme. We use direction information which is synthetic. Then, random noise chosen from a uniform distribution on a predefined interval is added to the direction data, and each scheme is used to denoise the image. The numerical error of each scheme is calculated. The algorithmic error is also defined, as the deviation of the resultant direction from the original noise-free direction data. The edge preserving quality of each algorithm is examined on an artificial image which is composed of two different directions, and also on an image which combines a slowly varying direction and a large direction edge.

7.3. Definition of the numerical error. The numerical error is differently defined and calculated for each scheme. In the linear and TV schemes, the numerical error is defined as the amount of the projection needed, so that the direction information is on the unit circle. Thus, if the flow has resulted in some coordinates $(U1, V1)$ which are not necessarily on the unit circle, we take as the projected coordinates the intersection of the unit circle with the line connecting $(U1, V1)$ to the origin of axes; see Figure 7.1. The point (U, V) is given by

$$U = \frac{U1}{\sqrt{U1^2 + V1^2}}, \quad V = \frac{V1}{\sqrt{U1^2 + V1^2}}. \tag{7.6}$$

Thus, the error is clearly

$$error = \sqrt{(U1 - U)^2 + (V1 - V)^2}. \tag{7.7}$$

In the HP and SP schemes, the evaluation of the error is not straightforward, as there is no projection error; the evolving coordinates never leave the unit circle. The numerical error is therefore defined relative to the results of a similar flow in which there is no selection of a local coordinate system; thus, the coordinates (u, v) are not coupled and are not constrained to the unit circle. For the HP we denote this error by $HE_{U1,V1}$ and expect it to obtain a sharp maximum at $(-\pi, \frac{-\pi}{2}, 0, \frac{\pi}{2}, \pi)$ because one of the internal coordinates approaches 1 there and the denominator approaches infinity (see Figure 7.2). It is important to notice that *it is not an error of the hemispheric scheme*. In its minimum value, obtained between the sharp maximum points, it provides a maximum bound on the error in the HP scheme.

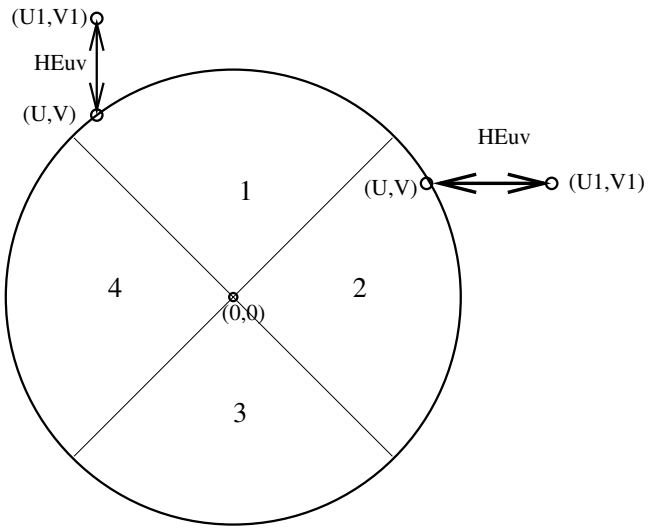

FIG. 7.2. *Artificial error in the HP scheme. In regions* 1 *and* 3 *the* U *coordinate is selected, and therefore the numerical error results from the difference between* $V1$*, which is independently calculated, and* V*, which is derived from the coordinate* U*. In regions* 2 *and* 4 *the* V *coordinate is selected, and therefore the numerical error results from the difference between* $U1$*, which is independently calculated, and* U*, which is derived from the coordinate* V*.*

For SP the definition of an error is even more complicated. Not only is there no projection error, but there are more variables for which an error term may be defined. First, u and v are obtained using the embedding $\theta \to (u, v) = [\cos(\theta), \sin(\theta)]$. Next, the stereographic coordinates Z_n and Z_s are derived, as the intersection of the line between the north (south) pole and the south (north) hemisphere. Thus, we may look at the error in Z_s and Z_n as well as in u and v. Following are the error terms used:

- $\boldsymbol{SE_{zn}}$ **and** $\boldsymbol{SE_{zs}}$**—Error terms for the stereographic coordinates.** We let Z_n and Z_s evolve independently. Then, we compare the stand-alone Z_n to the one calculated using the coupled Z_n and Z_s (where we select the local appropriate chart according to the direction). We do the same for Z_s (see Figure 7.3). We expect the error for Z_n to have a singularity at $\frac{\pi}{2}$ and the error for Z_s to have a singularity at $\frac{-\pi}{2}$. Note that SE_{zn} as defined is expected to be zero in the range $[-\pi, 0]$, and SE_{zs} as defined is expected to be zero in the range $[0, \pi]$. Since this is an error for the values of Z_n and Z_s, we need another error definition which measures the degree of error in the (u, v) coordinates.
- $\boldsymbol{SE_{UV}}$**—Error terms for** $\boldsymbol{U}$ **and** $\boldsymbol{V}$**.** It is important to evaluate the error for the (U, V) variables. We define the error term as the distance between the vector (U, V) when evaluated using the coupled Z_n and Z_s, and the vector (U, V) when using the independently calculated Z_n and Z_s (see Figure 7.4).

It is important to note that SE_z and SE_{UV} *are not errors* of the Beltrami porcupine methods. They give an indication of the actual error by noticing that *the minimum* of SE_z and SE_{UV} is *the upper bound for the Beltrami porcupine algorithm*. This is so since the most unreliable numeric regions are exactly the regions where the minimum in the $SE_{Z,UV}$ is obtained. The actual error in other areas is smaller since we do not trust one of the components that leads to a greater error. Thus, a small value of an error may indicate that using the appropriate local chart

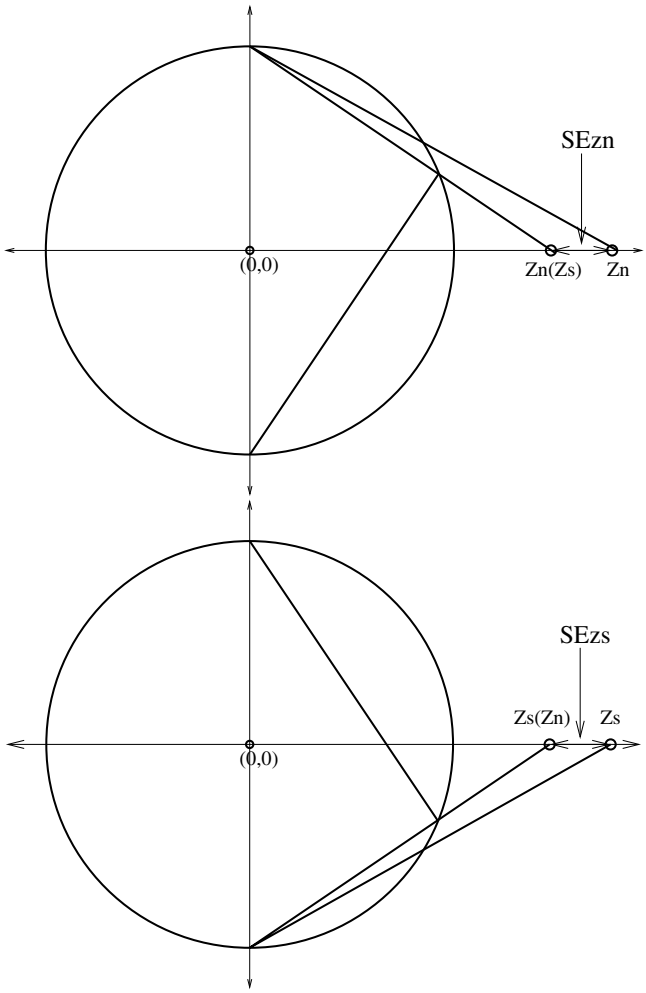

FIG. 7.3. *Artificial error in the Z_n (upper) and Z_s (lower) variables in the SP scheme.*

is not as important as it is when the error is larger. The higher the error, the more important it is to use the right local chart.

7.3.1. Definition of the algorithmic error. The definition of the algorithmic error is the same for all schemes. It is simply the deviation of the direction following diffusion from the noise-free direction, which is originally given. While the numerical error gives an indication of the stability of the method, the algorithmic error deals with performance: how close the resultant direction is to the actual one. The algorithmic error is defined as follows:

$$E = \sqrt{(\cos(\theta) - \cos(\theta_1))^2 + (\sin(\theta) - \sin(\theta_1))^2},$$

where θ is the original noise-free angle and θ_1 is the resultant angle following the diffusion scheme.

7.3.2. Definition of an edge preservation quality. An important quality of any diffusion scheme is its edge preserving ability. The first test image used to examine

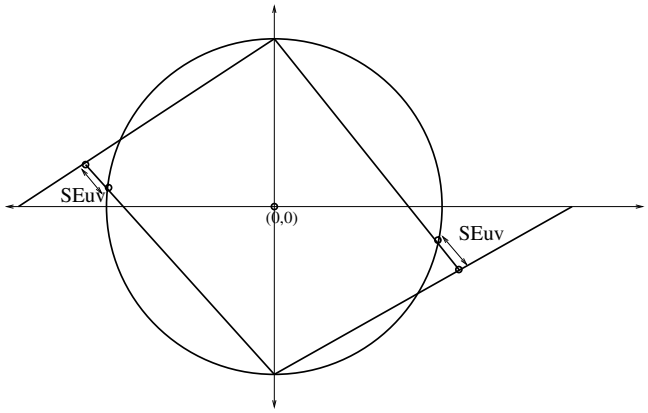

FIG. 7.4. *Artificial error in the* (U, V) *coordinates in the SP scheme.*

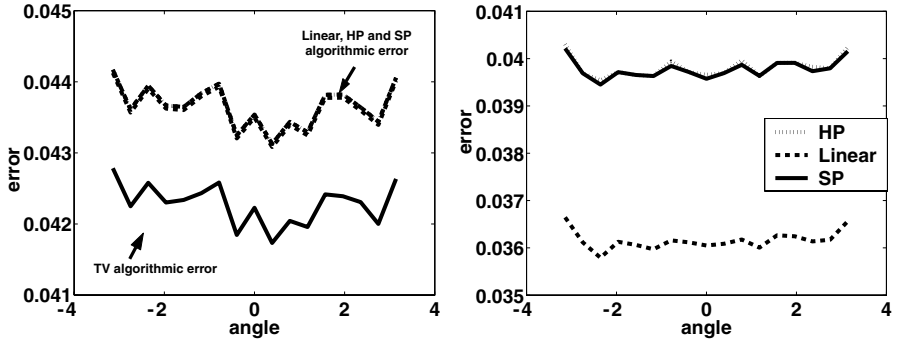

FIG. 7.5. *Algorithmic error for the linear, TV, HP, and SP schemes. Left: the four schemes together, using a small time step,* $dt = 0.00001$. *Right: the HP, SP, and linear schemes, using a larger time step,* $dt = 0.001$.

edge preservation is composed of two different directions. We apply each tested scheme to this image. We expect that the TV-based method will preserve edges better than the linear-based approach. As for the porcupine methods, we expect edge preservation quality to depend on the parameter β. The second test image is composed of two significantly different directions, where each direction is slowly varying. Using this test image, we may compare the edge preserving quality with the handling of the slowly varying data.

7.4. Comparison results and discussion. In this section we present the results of the numerical errors, algorithmic errors, and edge preserving performance.

In the test we go over S^1 from $-\pi$ to π using an equal step size. For each angle, random noise entries, chosen from a uniform distribution, are added to the vector field.

In Figure 7.5(left) we present the algorithmic error for the four schemes using a time step $dt = 0.00001$. All errors lie within the same range. However, the best performance is presented by the TV scheme, while the linear, HP, and SP approaches seem to have the same performance. In Figure 7.5(right) we used a larger time step, $dt = 0.001$, to observe the different behavior of the linear, HP, and SP schemes. The linear scheme has the smallest algorithmic error among the three schemes, and the HP and SP schemes seem to have the same algorithmic performance.

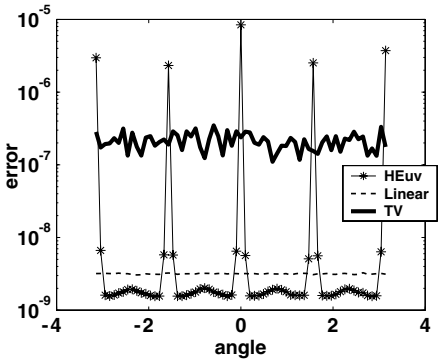

FIG. 7.6. *Numerical error for the TV, linear, and HP schemes. In this test we go over* S^1 *from* $-\pi$ *to* π *using an equal step size of* $\frac{\pi}{32}$.

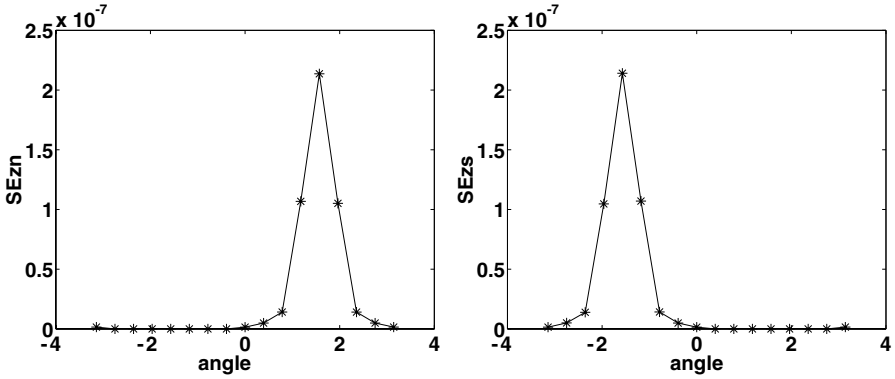

FIG. 7.7. *Left: numerical error for* Z_n. *Here we go over* S^1 *from* $-\pi$ *to* π *using an equal step size of* $\frac{\pi}{8}$. *As expected, it has a singular point at* $\frac{\pi}{2}$. *Right: numerical error for* Z_s. *As expected, it has a singular point at* $\frac{-\pi}{2}$.

Figure 7.6 compares the numerical errors of the HP, TV, and linear schemes. A logarithmic scale is used, as the error of the TV scheme is two orders of magnitude higher than the error of the linear and HP schemes! The HP error term has a periodic behavior, and it is very large at the singular points, $(-\pi, \frac{-\pi}{2}, 0, \frac{\pi}{2}, \pi)$. Away from the singular points, the HP error is slightly smaller than the linear scheme error, and the TV error is significantly higher than the HP error. However, as we approach the singularities, the HP error increases, and there the linear scheme's error is smaller.

In Figure 7.7 we show the numerical errors of Z_n and Z_s in the SP scheme. As expected, the errors have sharp maxima at $\frac{\pi}{2}$ and $\frac{-\pi}{2}$, respectively.

Another definition for the numerical error of the SP scheme was given, SE_{UV}, in which we refer to the (U, V) variables rather than the (Z_n, Z_s) variables. In Figure 7.8 this error is presented: the differences between the values of (U, V) when calculated using a coupled scheme for (Z_n, Z_s) and when calculated using an independent scheme for (Z_n, Z_s) are shown. It is interesting to note that this error has a periodic behavior, with maximum values at $(\frac{-\pi}{2}, \frac{\pi}{2})$, as can be expected.

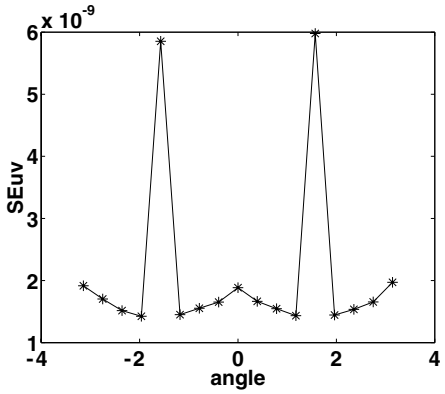

FIG. 7.8. *The differences between the values of* (u, v) *when calculated using the* (Z_n, Z_s) *coupled scheme and when independently calculated using* (Z_n, Z_s). *Here we go over* S^1 *from* $-\pi$ *to* π *using an equal step size of* $\frac{\pi}{8}$.

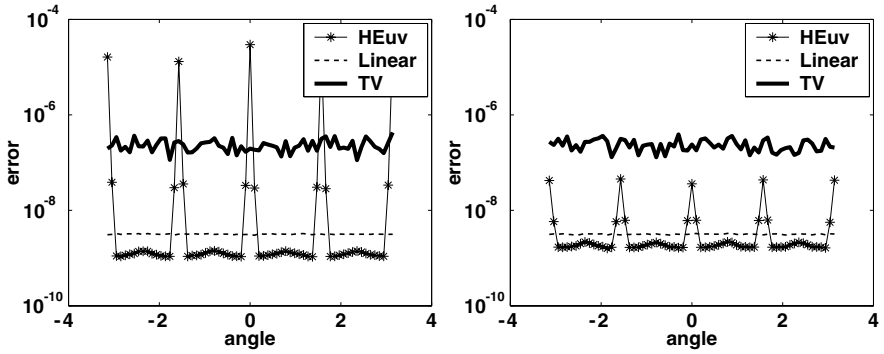

FIG. 7.9. *Numerical error for HP for* $\beta = 10$ *(left) and for* $\beta = 0$ *(right). Here we go over* S^1 *from* $-\pi$ *to* π *using an equal step size of* $\frac{\pi}{32}$.

The next step is to examine the dependence of the numerical error in the HP and SP schemes on the scaling parameter β. Figure 7.9 shows the numerical error for the HP scheme for a larger value of β (left) and for a smaller value of β (right). The scale used for presenting these results is again logarithmic. Away from the singular points, larger values of β produce smaller errors. In the vicinity of the singular points, the error increases when β increases.

The same goes for the SP scheme. In Figures 7.10 and 7.11 we present the results with respect to the three error measures we have defined for the SP scheme. The scale used for presenting the results is logarithmic. In Figure 7.10 the results for a larger value of β are presented, and in Figure 7.11 the results for a smaller value of β are presented.

When $\beta = 100$, the values of SE_{zn} in the range $[0, \pi]$ and away from the singularity at $\frac{\pi}{2}$ lie between the numerical errors of the linear and TV schemes. The error decreases as we move away from $\frac{\pi}{2}$ and is even smaller than the linear scheme error as we move closer to 0 and π. In the range $[-\pi, 0]$, SE_{zn} is equal to zero. SE_{zs} presents a mirror behavior. SE_{UV} is smaller than the numerical errors of the TV and linear schemes. It obtains maximum values at $\pm\frac{\pi}{2}$. When $\beta = 0$, the values of SE_{zn} in the

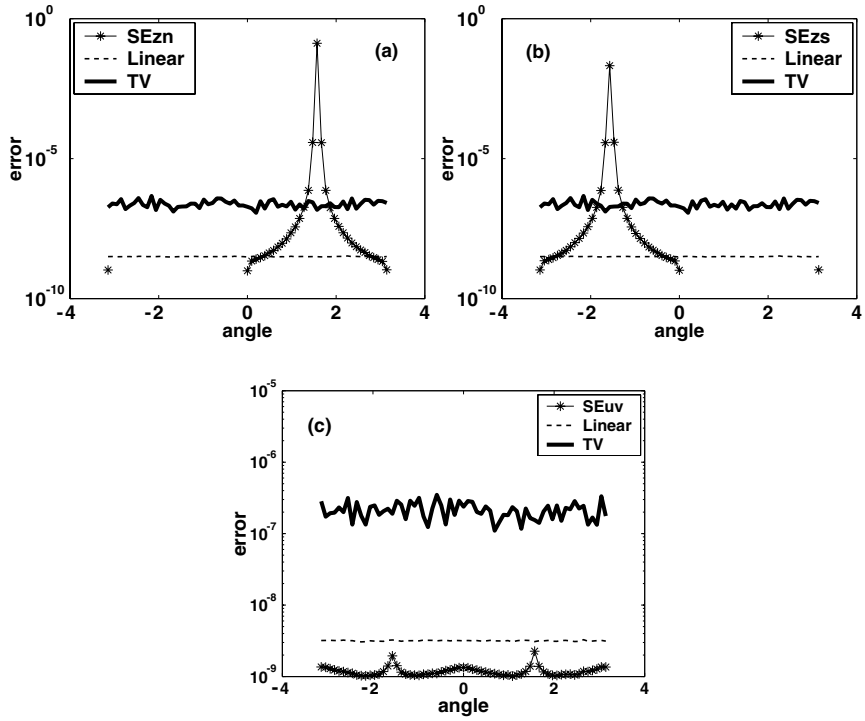

FIG. 7.10. *Numerical errors of the SP scheme for a large value of* $\beta = 100$. (a) *The numerical error of* Z_n. (b) *The numerical error of* Z_s. (c) *The numerical error of the* (U, V) *variables.*

range $[0, \pi]$ and away from the singularity at $\frac{\pi}{2}$ are a little bigger than those obtained for $\beta = 100$. Again, SE_{zs} presents a mirror behavior. In this case, SE_{UV}, away from the singular points $\pm\frac{\pi}{2}$, is higher than the one obtained for $\beta = 100$. Note that the error values in the vicinity of the singularities are much higher for the lower value of β.

Next, we examine the edge preserving quality of each direction diffusion scheme. The following synthetic data was generated so that there is a difference of $\frac{\pi}{2}$ radians between the left and right sides of the noise-free image. Random noise entries, chosen from a uniform distribution in the range $[\frac{-\pi}{9}, \frac{\pi}{9}]$, are added to the noise-free data, and each scheme is applied to the image. The noise-free and noisy initial images are shown in Figure 7.12. The diffusion results are presented for all schemes, while for the HP and SP approaches we show the results for both smaller and higher values of the parameter β. In Figure 7.13 the results for the linear and the TV schemes are presented. It is interesting to note that the linear scheme is less edge preserving than the TV scheme, as can be expected. In Figures 7.14 and 7.15 the results for the HP and SP schemes are also presented. Here, we note the dependence of the results on the value of the parameter β. We can go from linear to TV behavior simply by adjusting the value of β. If we examine the relationship between the numerical errors of the TV and linear schemes (see Figure 7.6), and their edge preserving quality, we note that while the linear scheme offers a low numerical error, it is less edge preserving, and while the TV scheme better preserves edges, it has a significantly higher numerical error. For the HP and SP schemes, both the numerical errors (see Figures 7.9, 7.10, 7.11) and the edge preserving quality depend on the parameter β. We may find a

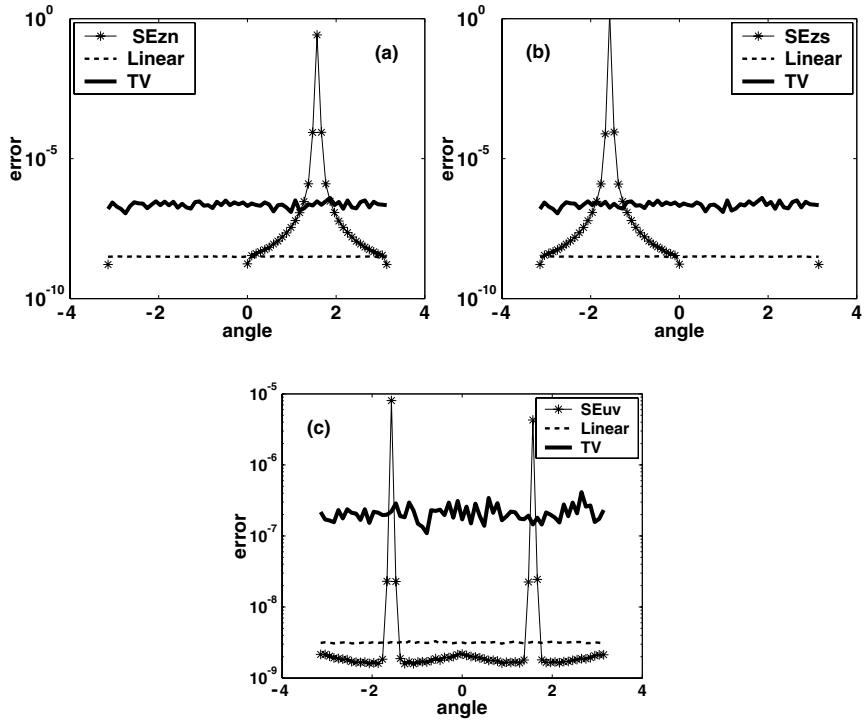

FIG. 7.11. *Numerical errors for the SP scheme for a small value of* $\beta = 0$. (a) *The numerical error of* Z_n. (b) *The numerical error of* Z_s. (c) *The numerical error of the* (U, V) *variables.*

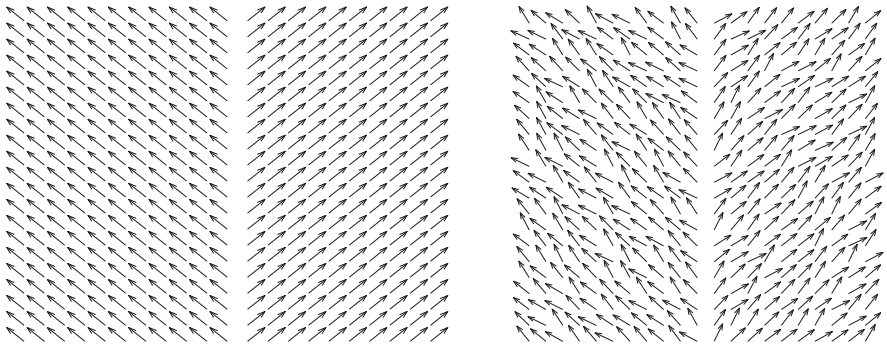

FIG. 7.12. *The original noise-free image (left) and the image after random noise was added (right).*

value of β in the HP and SP schemes so that we obtain a numerical error which is in the order of the linear scheme's error and an edge preserving quality which is comparable to that of the TV scheme.

Another example for exploring the edge preserving quality of each scheme is the direction fan example. The test image (Figure 7.15) is composed of a major gradient in directions in the image's center and a slowly varying angle as we move away from the center. The direction information is presented both by arrows (Figure 7.16 (left)) and by a color image, representing the angles (Figure 7.16 (right)). Random noise

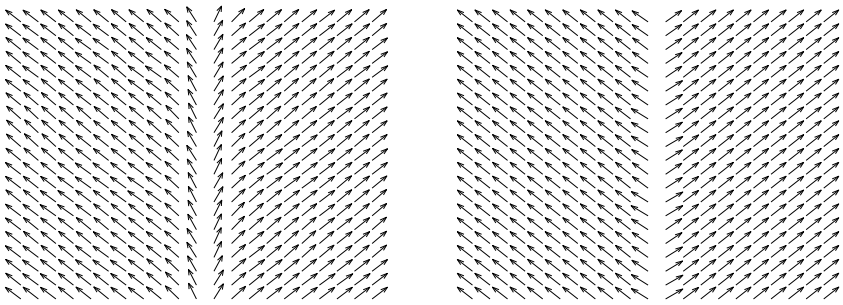

FIG. 7.13. *Left: the result of linear diffusion, with* 10,000 *iterations and a time step equal to* 0.0001. *Right: the result of TV diffusion, with* 100,000 *iterations and a time step equal to* 0.00001.

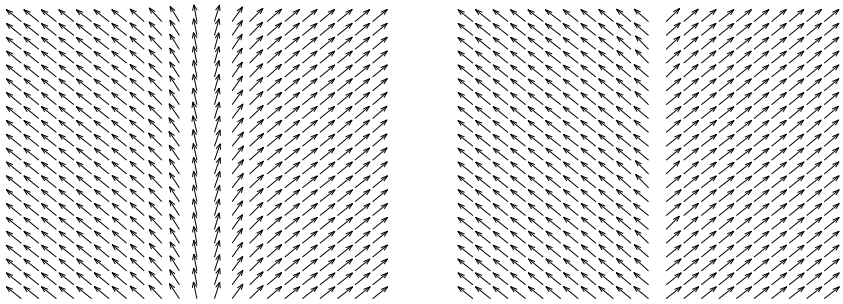

FIG. 7.14. *Left: the result of HP diffusion for* $\beta = 0$. *Right: the result of HP diffusion for* $\beta = 10$. *These results were obtained following* 10,000 *iterations with time step equal to* 0.001.

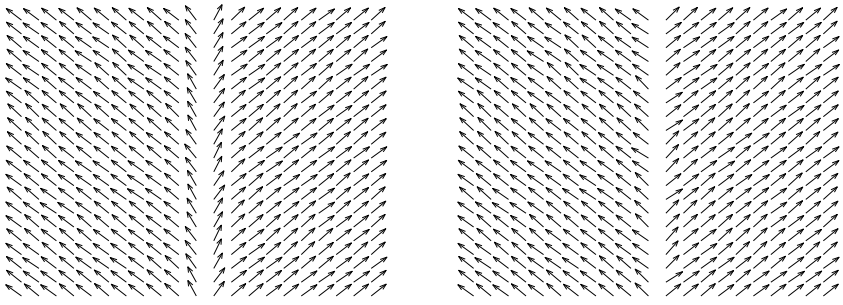

FIG. 7.15. *Left: the result of SP diffusion for* $\beta = 0$. *Right: the result of SP diffusion for* $\beta = 100$. *These results were obtained following* 10,000 *iterations with time step equal to* 0.0001.

entries, chosen from a uniform distribution in the range $[\frac{-\pi}{9}, \frac{\pi}{9}]$, are added to the noise-free data, and a noisy direction image is obtained (Figure 7.17). Next, each scheme is applied to the image with the time step, number of iterations, and value of β (for the HP and SP schemes) that produce the best results. When applying the linear scheme, the edge is blurred while the amount of noise is still significant (Figure 7.18). The TV approach results in a sharper boundary relative to the linear

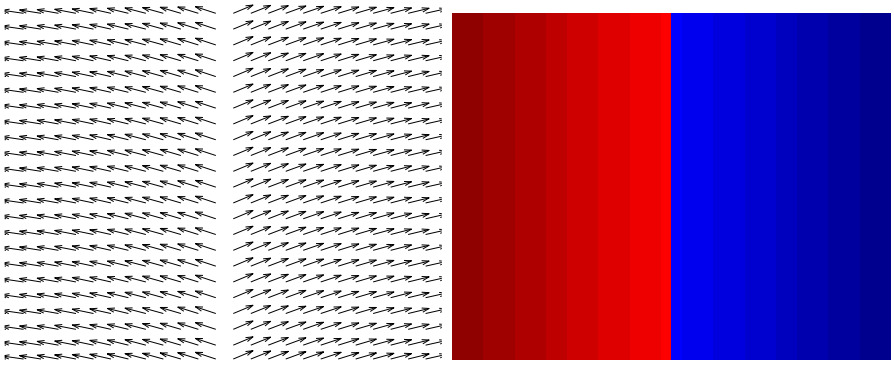

FIG. 7.16. *The noise-free direction fan image, represented by arrows (left) and as a color image (right).*

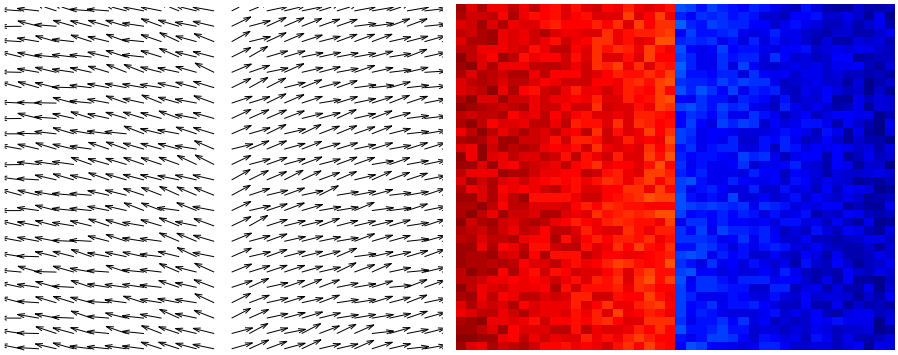

FIG. 7.17. *The noisy direction fan image, represented by arrows (left) and as a color image (right).*

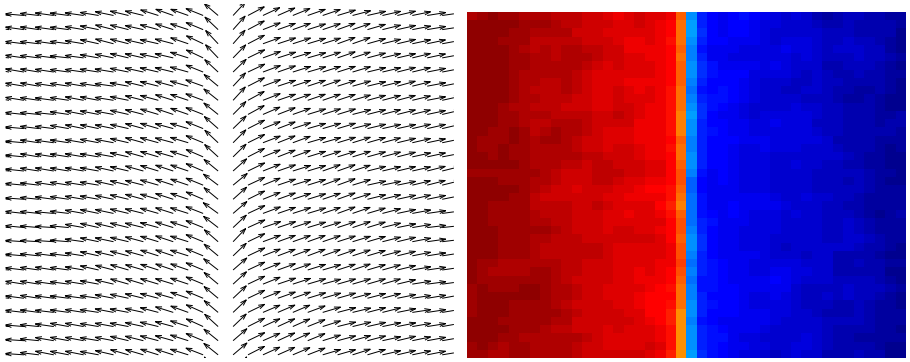

FIG. 7.18. *The result of linear diffusion following* 10,000 *iterations with time step* 0.0001*, represented by arrows (left) and as a color image (right).*

scheme, but if we examine the smoothed direction, we note a stair-casing effect; thus the smaller changes in direction are ignored (Figure 7.19).

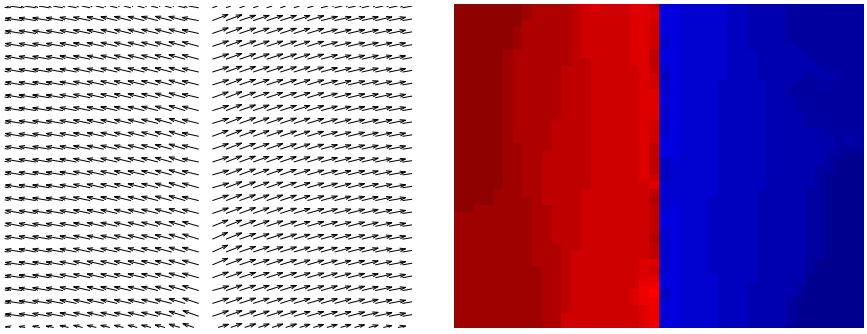

FIG. 7.19. *The result of TV diffusion following* 100,000 *iterations with time step* 0.00001*, represented by arrows (left) and as a color image (right).*

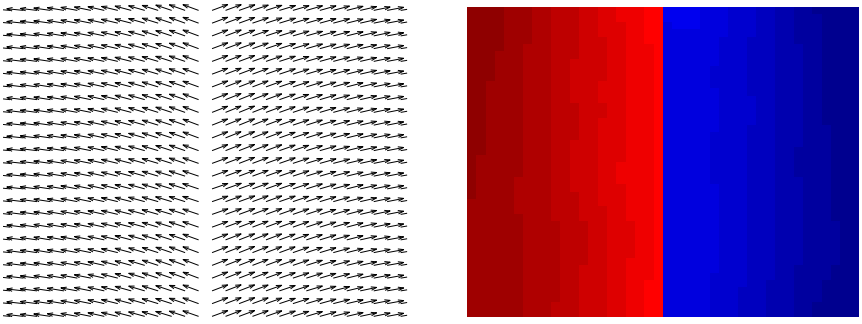

FIG. 7.20. *The result of HP diffusion following* 1,000 *iterations with time step* 0.01*. The value of* β *is* 1.5*. Representation by arrows (left) and as a color image (right).*

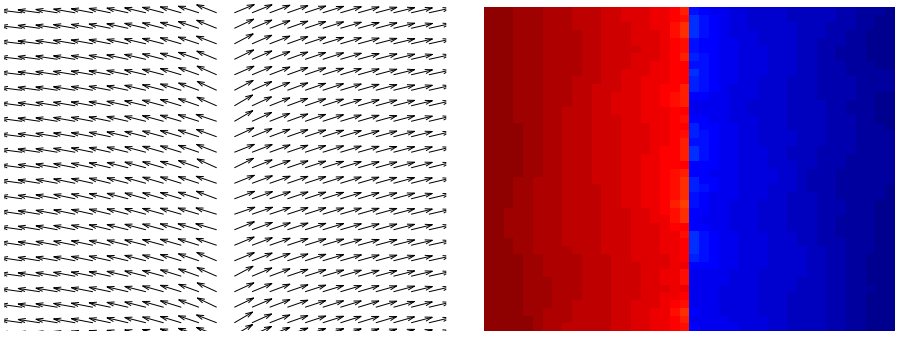

FIG. 7.21. *The result of SP diffusion following* 1,000 *iterations with time step* 0.01*. The value of* β *is* 10*. Representation by arrows (left) and as a color image (right).*

The HP scheme produces good results, as it keeps a sharp boundary and restores the original slowly changing behavior of the original direction data (Figure 7.20). The SP scheme produces similar results to those for the HP scheme, but as can be seen, some noise is still present (Figure 7.21).

8. Concluding remarks. There are some important issues in the process of denoising a constrained feature field. The first is to make the process compatible with the constraint in such a way that the latter is never violated along the flow. The second is the type of regularization which is applied in order to preserve significant discontinuities of the feature field while removing noise. The third is the numeric and algorithmic accuracy of the algorithms.

These issues are treated in this paper via the Beltrami framework. First a Riemannian structure, i.e., a metric, is introduced on the feature manifold, and several local coordinate systems are chosen to *intrinsically* describe the constrained feature manifold. The diffusion process acts on these coordinates, and the compatibility with the constraint is achieved through the intrinsic nature of the coordinate system. The difficulty in working on a non-Euclidean space transforms itself to the need to locally choose the best coordinate system to work with.

The preservation of significant discontinuities is dealt with by using the induced metric and the corresponding Laplace–Beltrami operator acting on feature coordinates only. This operation is in fact a projection of the mean curvature, in the normal(s) direction(s) to the surface, to the feature direction(s). This projection slows the diffusion process along significant (supported) discontinuities while letting the process proceed in the homogeneous regions at a normal speed.

The result of this algorithm is an adaptive smoothing process for a constrained feature space in every dimension and codimension. As an example we have shown how our geometric model coupled with a proper choice of charts handles the direction diffusion problem. This is a new application of the Beltrami framework, proposed in [18]. We tested the new model on vector fields restricted to the unit circle S^1, and hybrid spaces like the HSV color space. The integration of the spatial coordinates with the color coordinates yields a selective smoothing filter for images in which some of the coordinates are restricted to a circle.

Moreover, it is shown that even when algorithms are analytically equivalent, they may differ in their accuracy (numerical and algorithmic). It is shown that the hemispheric and stereographic coordinate systems present an advantage in the sense that a parameter β can be found, i.e., $\beta = 10$, or 100, respectively, such that the edge preserving quality is as good as that for the TV algorithm, while the numerical error is two orders of magnitude smaller!

Appendix A. The Levi–Civita method for S^2. Using (3.9) and the general formula

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

we get, for example,

$$\begin{aligned}
\Gamma^3_{33} &= \frac{1}{2} h^{3l} \left(2\partial_3 h_{l3} - \partial_l h_{33} \right) = \frac{1}{2} \left(h^{33} \partial_3 h_{33} + 2h^{34} \partial_3 h_{34} - h^{34} \partial_4 h_{33} \right) \\
&= \frac{1}{2} \Bigg[(1-U^2) \frac{\partial}{\partial U} \left(\frac{1-V^2}{1-U^2-V^2} \right) - 2UV \frac{\partial}{\partial U} \left(\frac{UV}{1-U^2-V^2} \right) \\
&\qquad + UV \frac{\partial}{\partial V} \left(\frac{1-V^2}{1-U^2-V^2} \right) \Bigg],
\end{aligned}$$

and a straightforward calculation gives

$$\Gamma^3_{33} = \frac{U(1-V^2)}{1-U^2-V^2} = Uh_{33}. \tag{A.2}$$

Appendix B. The S^n diffusion flow. The hypersphere S^n is presented as an n-dimensional manifold embedded in $\mathbb{R}^{n+1}$ as the constrained hypersurface

$$\sum_{i=1}^{n+1}(U^i)^2 = 1.$$

We work in the chart where $\{U^i\}_{i=1}^n$ are local coordinates. On this chart, $U^{n+1} = \sqrt{1-\sum_{i=1}^n (U^i)^2}$.

THEOREM B.1. *The local S^n metric elements are*

$$\tilde{h}_{ij} = \delta^{ij} + \frac{U^iU^j}{1-\sum_{s=1}^n (U^s)^2}.$$

Proof. The hypersphere is embedded isometrically in $\mathbb{R}^{n+1}$. We use the induced metric technique as follows:

$$ds^2 = \sum_{i=1}^n (dU^i)^2 + (dU^{n+1})^2. \tag{B.1}$$

The U^{n+1} coordinate is a function of all the others, and as such we can apply the chain rule to get

$$dU^{n+1} = \sum_{i=1}^n \frac{\partial U^{n+1}}{\partial U^i} dU^i = -\sum_{i=1}^n \frac{U^i}{\sqrt{1-\sum_{s=1}^n (U^s)^2}} dU^i.$$

Using this expression in (B.1), we get

$$\begin{aligned} ds^2 &= \sum_{i,j=1}^n \tilde{h}_{ij} dU^i dU^j \\ &= \sum_{i=1}^n (dU^i)^2 + \left(-\sum_{i=1}^n \frac{U^i}{\sqrt{1-\sum_{s=1}^n (U^s)^2}} dU^i\right)\left(-\sum_{j=1}^n \frac{U^j}{\sqrt{1-\sum_{s=1}^n (U^s)^2}} dU^j\right) \\ &= \sum_{i,j=1}^n \delta_{ij} dU^i dU^j + \sum_{i,j=1}^n \frac{U^iU^j}{1-\sum_{s=1}^n (U^s)^2} dU^i dU^j \\ &= \sum_{i,j=1}^n \left(\delta_{ij} + \frac{U^iU^j}{1-\sum_{s=1}^n (U^s)^2}\right) dU^i dU^j, \end{aligned} \tag{B.2}$$

from which the assertion follows. ∎

THEOREM B.2. *The local S^n inverse metric elements are*

$$\tilde{h}^{-1}_{ij} = \delta^{ij} - U^iU^j.$$

Proof. By direct calculation,

$$\begin{aligned}\sum_{j=1}^{n}\tilde{h}_{ij}\tilde{h}^{-1}_{jk} &= \sum_{j=1}^{n}\left(\delta^{ij}+\frac{U^iU^j}{1-\sum_{s=1}^{n}(U^s)^2}\right)\left(\delta^{jk}-U^jU^k\right)\\ &= \delta^{ik}-U^iU^k+\frac{U^iU^k}{1-\sum_{s=1}^{n}(U^s)^2}-\sum_{j=1}^{n}\frac{U^i(U^j)^2U^k}{1-\sum_{s=1}^{n}(U^s)^2}\\ &= \delta^{ik}. \qquad \text{(B.3)}\end{aligned}$$

One can check similarly that

$$\sum_{j=1}^{n}\tilde{h}^{-1}_{ij}\tilde{h}_{jk}=\delta^{ik}. \qquad \square$$

THEOREM B.3. *The induced metric, and its inverse, are accordingly*

$$\begin{aligned}g_{\mu\nu} &= \delta_{\mu\nu}+\sum_{i,j=1}^{n}\tilde{h}_{ij}U^i_\mu U^j_\nu,\\ g^{\mu\nu} &= \frac{1}{g}\left(\delta^{\mu\nu}+\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j=1}^{n}\tilde{h}_{ij}U^i_\sigma U^j_\rho\right),\\ g &= \det(g_{\mu\nu})\\ &= 1+\sum_{i,j=1}^{n}\tilde{h}_{ij}(U^i_xU^j_x+U^i_yU^j_y)+\frac{1}{2}\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j,k,l=1}^{n}\tilde{h}_{ij}\tilde{h}_{kl}U^i_\mu U^j_\nu U^k_\rho U^l_\sigma, \qquad \text{(B.4)}\end{aligned}$$

where $(g^{\mu\nu})$ is the inverse of $(g_{\mu\nu})$, g is the determinant, and $\epsilon^{\mu\nu}$ is the 2D antisymmetric tensor

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

An implicit summation on all repeated Greek indices is assumed.

Proof. The calculation of the metric element is done directly by the induced metric identity

$$\begin{aligned}ds^2 = g_{\mu\nu}dx^\mu dx^\nu &= dx^2+dy^2+\sum_{i,j=1}^{n}\tilde{h}_{ij}dU^idU^j\\ &= \delta_{\mu\nu}dx^\mu dx^\nu+\sum_{ij}\tilde{h}_{ij}U^i_\mu U^j_\nu dx^\mu dx^\nu, \qquad \text{(B.5)}\end{aligned}$$

from which we extract the metric coefficients. The metric is a 2×2 matrix whose determinant is $g=g_{11}g_{22}-g_{12}^2=\epsilon^{\mu\nu}g_{1\mu}g_{2\nu}$. Using the explicit form of the metric, we get

$$
\begin{aligned}
g &= \left(1+\sum_{ij}\tilde{h}_{ij}U_x^iU_x^j\right)\left(1+\sum_{kl}\tilde{h}_{kl}U_y^kU_y^l\right)-\left(\sum_{ij}\tilde{h}_{ij}U_x^iU_y^j\right)\left(\sum_{kl}\tilde{h}_{kl}U_x^kU_y^l\right)\\
&= 1+\sum_{ij}\tilde{h}_{ij}(U_x^iU_x^j+U_y^iU_y^j)+\sum_{ijkl}\tilde{h}_{ij}\tilde{h}_{kl}U_x^i(U_x^jU_y^k-U_y^jU_x^k)U_y^l\\
&= 1+\sum_{ij}\tilde{h}_{ij}(U_x^iU_x^j+U_y^iU_y^j)+\epsilon^{\mu\nu}\sum_{ijkl}\tilde{h}_{ij}\tilde{h}_{kl}U_x^iU_\mu^jU_\nu^kU_y^l\\
&= 1+\sum_{ij}\tilde{h}_{ij}(U_x^iU_x^j+U_y^iU_y^j)\\
&\quad+\frac{1}{2}\epsilon^{\mu\nu}\sum_{ijkl}\tilde{h}_{ij}\tilde{h}_{kl}U_x^iU_\mu^jU_\nu^kU_y^l-\frac{1}{2}\epsilon^{\mu\nu}\sum_{ijkl}\tilde{h}_{ij}\tilde{h}_{kl}U_y^iU_\mu^jU_\nu^kU_x^l \qquad \text{(B.6)}\\
&= 1+\sum_{ij}\tilde{h}_{ij}(U_x^iU_x^j+U_y^iU_y^j)+\frac{1}{2}\epsilon^{\mu\nu}\epsilon^{\sigma\rho}\sum_{ijkl}\tilde{h}_{ij}\tilde{h}_{kl}U_\sigma^iU_\mu^jU_\nu^kU_\rho^l.
\end{aligned}
$$

Finally, we prove the formula for the inverse metric

$$
\begin{aligned}
g^{\mu\nu}g_{\nu\lambda} &= \frac{1}{g}\left(\delta^{\mu\nu}+\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j=1}^{n}\tilde{h}_{ij}U_\sigma^iU_\rho^j\right)\left(\delta_{\nu\lambda}+\sum_{i,j=1}^{n}\tilde{h}_{ij}U_\nu^iU_\lambda^j\right)\\
&= \frac{1}{g}\left(\delta_\lambda^\mu+\epsilon^{\mu\sigma}\epsilon^{\lambda\rho}\sum_{i,j=1}^{n}\tilde{h}_{ij}U_\sigma^iU_\rho^j+\sum_{k,l=1}^{n}\tilde{h}_{kl}U_\mu^kU_\lambda^l\right.\\
&\qquad\left.+\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j,k,l=1}^{n}\tilde{h}_{ij}\tilde{h}_{kl}U_\sigma^iU_\rho^jU_\nu^kU_\lambda^l\right) \qquad \text{(B.7)}\\
&= \frac{1}{g}\left(\delta_\lambda^\mu+\sum_{i,j=1}^{n}\tilde{h}_{ij}(U_x^iU_x^j+U_y^iU_y^j)\delta_\lambda^\mu+\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j,k,l=1}^{n}\tilde{h}_{ij}\tilde{h}_{kl}U_\sigma^iU_\rho^jU_\nu^kU_\lambda^l\right),
\end{aligned}
$$

where the last equality comes from a case-by-case analysis. Remember that $\lambda,\nu\in\{1,2\}$, and take, for example, $\mu=\lambda-1=1$. In this case we get

$$
\begin{aligned}
\sum_{\sigma,\rho=1}^{2}\epsilon^{1\sigma}\epsilon^{2\rho}\sum_{i,j=1}^{n}\tilde{h}_{ij}U_\sigma^iU_\rho^j+\sum_{k,l=1}^{n}\tilde{h}_{kl}U_x^kU_y^l &= \epsilon^{12}\epsilon^{21}\sum_{i,j=1}^{n}\tilde{h}_{ij}U_y^iU_x^j+\sum_{i,j=1}^{n}\tilde{h}_{ij}U_x^iU_y^a\\
&= -\sum_{i,j=1}^{n}\tilde{h}_{ij}U_y^iU_x^j+\sum_{i,j=1}^{n}\tilde{h}_{ij}U_x^jU_y^i=0, \qquad \text{(B.8)}
\end{aligned}
$$

where we have used the fact that the metric is a symmetric tensor. Other cases are analyzed in a similar manner. The third term is also analyzed on a case-by-case basis, and the result, as the reader can verify, is

$$
\epsilon^{\mu\sigma}\epsilon^{\nu\rho}\sum_{i,j,k,l=1}^{n}\tilde{h}_{ij}\tilde{h}_{kl}U_\sigma^iU_\rho^jU_\nu^kU_\lambda^l=\frac{1}{2}\delta_\lambda^\mu\epsilon^{\alpha\beta}\epsilon^{\nu\rho}\sum_{i,j,k,l=1}^{n}\tilde{h}_{ij}\tilde{h}_{kl}U_\alpha^iU_\nu^jU_\rho^kU_\beta^l.
$$

The whole expression in the parentheses in B.7 is, therefore, $\delta_\lambda^\mu g$, which completes our proof. ∎

The last piece of information needed for our machinery is the explicit form of the Levi–Civita coefficients.

THEOREM B.4. *The Levi–Civita coefficients are*

$$\Gamma^i_{jk} = U^i \tilde{h}_{jk}. \tag{B.9}$$

Proof. From the formula (2.11) we get

$$\begin{aligned}\Gamma^i_{jk} &= \frac{1}{2}\sum_l h^{-1}_{il}\left(\partial_j h_{lk} + \partial_k h_{jl} - \partial_l h_{jk}\right)\\ &= \frac{1}{2}\sum_l (\delta^{il} - U^i U^l)\left(\partial_j\left(\frac{U^l U^k}{1-\sum_s (U^s)^2}\right) + \partial_k\left(\frac{U^j U^l}{1-\sum_s (U^s)^2}\right) - \partial_l\left(\frac{U^j U^k}{1-\sum_s (U^s)^2}\right)\right).\end{aligned}$$

Let us compute the first term, for example,

$$\partial_j\left(\frac{U^l U^k}{1-\sum_s (U^s)^2}\right) = \frac{\delta^{jl}U^k}{1-\sum_s (U^s)^2} + \frac{\delta^{jk}U^l}{1-\sum_s (U^s)^2} + \frac{2U^j U^l U^k}{(1-\sum_s (U^s)^2)^2}. \tag{B.10}$$

Summing up the three terms, we get

$$\begin{aligned}\Gamma^i_{jk} = \frac{1}{2}\sum_l (\delta^{il} - U^i U^l)\Bigg(&\frac{\delta^{jl}U^k}{1-\sum_s (U^s)^2} + \frac{\delta^{kj}U^l}{1-\sum_s (U^s)^2} + \frac{2U^j U^l U^k}{(1-\sum_s (U^s)^2)^2}\\ &+ \frac{\delta^{kj}U^l}{1-\sum_s (U^s)^2} + \frac{\delta^{lk}U^j}{1-\sum_s (U^s)^2} + \frac{2U^j U^l U^k}{(1-\sum_s (U^s)^2)^2}\\ &- \frac{\delta^{lk}U^j}{1-\sum_s (U^s)^2} - \frac{\delta^{jl}U^k}{1-\sum_s (U^s)^2} - \frac{2U^j U^l U^k}{(1-\sum_s (U^s)^2)^2}\Bigg).\end{aligned}$$

Now simple algebra gives

$$\begin{aligned}\Gamma^i_{jk} &= \frac{1}{1-\sum_s (U^s)^2}\sum_l (\delta^{il} - U^i U^l)\left(\delta^{kj}U^l + \frac{U^j U^l U^k}{1-\sum_s (U^s)^2}\right)\\ &= \frac{1}{1-\sum_s (U^s)^2}\left(U^i - U^i\sum_l (U^l U^l)\right)\left(\delta^{kj} + \frac{U^j U^k}{1-\sum_s (U^s)^2}\right) = U^i \tilde{h}_{jk}. \qquad \square\end{aligned}$$

Acknowledgments. We thank Alfred Bruckstein from the Technion Israel for stimulating discussions on diffusion and averaging, and on color analysis. We also thank Guillermo Sapiro from the University of Minnesota for sharing with us his ideas and results on direction diffusion.

REFERENCES

[1] M. BERTALMÍO, L. T. CHENG, S. OSHER, AND G. SAPIRO, *Variational problems and partial differential equations on implicit surfaces*, J. Comput. Phys., 174 (2001), pp. 759–780.

[2] T. CHAN AND J. SHEN, *Variational restoration of nonflat image features: Models and algorithms*, SIAM J. Appl. Math., 61 (2000), pp. 1338–1361.

[3] R. COHEN, R. M. HARDT, D. KINDERLEHRER, S. Y. LIN, AND M. LUSKIN, *Minimum energy configurations for liquid crystals: Computational results*, in Theory and Applications of Liquid Crystals, J. L. Ericksen and D. Kinderlehrer, eds., IMA Vol. Math. Appl. 5, Springer, New York, 1987, pp. 99–121.

[4] L. Dascal and N. Sochen, *A maximum principle for the Beltrami color flow*, manuscript.
[5] L. Dascal, S. Kamin, and N. Sochen, *Existence and Uniqueness of the Weak Solutions to the Beltrami Flow*, in preparation.
[6] J. Eells and L. Lamarie, *A report on harmonic maps*, Bull. London Math. Soc., 10 (1978), pp. 1–68.
[7] J. Eells and L. Lamarie, *Another report on harmonic maps*, Bull. London Math. Soc., 20 (1988), pp. 385–524.
[8] R. Kimmel and N. Sochen, *Orientation diffusion or How to comb a porcupine*, J. Visual Communication and Image Representation, 13 (2002), pp. 238–248.
[9] R. Kimmel, R. Malladi, and N. Sochen, *Images as embedded maps and minimal surfaces: Movies, color, texture, and volumetric medical images*, Int. J. Computer Vision, 39 (2000), pp. 111–129.
[10] E. Kreyszig, *Differential Geometry*, Dover Publications, New York, 1991.
[11] J. Nash, *The imbedding problem for Riemannian manifolds*, Ann. Math., 63 (1965), pp. 20–63.
[12] A. M. Polyakov, *Quantum geometry of bosonic strings*, Phys. Lett. B, 103 (1981), pp. 207–210.
[13] P. Perona, *Orientation diffusion*, IEEE Trans. Image Process., 7 (1998), pp. 457–467.
[14] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation-based noise removal algorithms*, Phys. D, 60 (1991), pp. 259–268.
[15] A. R. Smith, *Color gamut transform pairs*, in SIGGRAPH: ACM Special Interest Group on Computer Graphics and Interactive Techniques, ACM Press, NY, 1979, pp. 12–19.
[16] N. Sochen, R. Deriche, and L. Lopez-Perez, *The Beltrami Flow over Manifolds*, Technical report TR-4897, INRIA Sophia-Antipolis, Sophia Antipolis, France, 2003.
[17] N. Sochen, R. M. Haralick, and Y. Y. Zeevi, *A Geometric Functional for Derivatives Approximation*, in Proceedings of the 2nd International Conference on Scale-Space Theories in Computer Vision, Lecture Notes in Comput. Sci. 1682, Springer-Verlag, Berlin, 1999, pp. 507–512.
[18] N. Sochen, R. Kimmel, and R. Malladi, *From High Energy Physics to Low Level Vision*, Technical report, LBNL 39243, University of California at Berkeley, Berkeley, CA, 1996.
[19] N. Sochen, R. Kimmel, and R. Malladi, *A general framework for low level vision*, IEEE Trans. Image Process., 7 (1998), pp. 310–318.
[20] N. Sochen and Y. Y. Zeevi, *Representation of colored images by manifolds embedded in higher dimensional non-Euclidean space*, in Proceedings of the International Conference on Image Processing, Chicago, 1998, IEEE, Los Alamitos, CA, pp. 166–170.
[21] B. Tang, G. Sapiro, and V. Caselles, *Direction diffusion*, in Proceedings of the International Conference on Computer Vision, Corfu, Greece, 1999, Vol. 2, pp. 1245–1252.
[22] B. Tang, G. Sapiro, and V. Caselles, *Diffusion of general data on non-flat manifolds via harmonic maps theory: The direction diffusion case*, Int. J. Computer Vision, 36 (2000), pp. 149–161.
[23] D. Tchumperlé and R. Deriche, *Regularization of orthonormal vector sets using coupled PDE's*, in Proceedings of the 2001 IEEE Workshop on Variational and Level Set Methods in Computer Vision (VLSM'01), Vancouver, 2001, IEEE, Los Alamitos, CA, pp. 3–10.
[24] J. Weickert, *Coherence-enhancing diffusion of colour images*, Image and Vision Computing, 17 (1999), pp. 201–212.