The Texture Gradient Equation for Recovering Shape from Texture

Maureen Clerc and Stéphane Mallat, *Member*, *IEEE*

Abstract—This paper studies the recovery of shape from texture under perspective projection. We regard *Shape from Texture* as a statistical estimation problem, the texture being the realization of a stochastic process. We introduce *warplets*, which generalize wavelets over the 2D affine group. At fine scales, the warpogram of the image obeys a transport equation, called *Texture Gradient Equation*. In order to recover the 3D shape of the surface, one must estimate the *deformation gradient*, which measures metric changes in the image. This is made possible by imposing a notion of homogeneity for the original texture, according to which the deformation gradient is equal to the velocity of the Texture Gradient Equation. By measuring the warplet transform of the image at different scales, we obtain a deformation gradient estimator.

Index Terms—Shape from texture, texture gradient, warplets, wavelets.

1 Introduction

When observing a static monocular image, we perceive the 3D structure of a scene through a combination of shape cues, especially shading, occlusion, and texture. *Shape from Texture*, first introduced 50 years ago by Gibson [1], studies the recovery of the 3D coordinates of a surface in a scene, by analyzing the distortion of its texture projected in an image [2], [3], [4], [5], [6]. The *Shape from Texture* problem is generally broken down into two independent steps. The first step is to measure the texture distortion in the image and the second is to recover the surface coordinates from this texture distortion.

Texture can be modeled either deterministically or stochastically. Although structural, or geometry-based methods allow the recovery of 3D surface coordinates for deterministic textures, stochastic models encompass a wider class of textures [4], [5], [7]. Measuring the distortion of stochastic textures requires local spectral measurements, obtained by convolving the image with waveforms which are localized in space as well as in spatial frequency. The local filtering most commonly used is based on the localized Fourier transform [8], [5], and wavelets have also recently been introduced for *Shape from Texture* [3], [9].

Traditionally, one measures the texture distortion by assuming a property on the original texture (for instance, its homogeneity, its isotropy, or its spectral content), and comparing the properties of the texture in the observed image to the prior information on the original texture. A differential analysis consists of measuring the relative distortion of the texture within the observed image, without reference to the original texture. In [5], the relative texture distortion between neighboring texture patches is approximated by an affine transform, and measured with a local Fourier transform.

Unlike local Fourier functions, wavelets have the property of migrating in position and scale under a 1D affine transform, which leads to a simpler and more precise estimation of the deformation. In two dimensions, to maintain this migration property, it is necessary to generalize wavelets into *warplets*, whose "scale" is no longer a scalar but a 2×2 warping matrix. The observed textured image is modeled as the realization of a stochastic process. The texture distortion can locally be approximated by a 2D affine transform, and the variance of the warplet coefficients, called the *warpogram*, thus undergoes a transport in the position-warping parameter space. This fundamental transport equation obeyed by the warpogram is called *Texture Gradient Equation*. It can be regarded as the analog of the Optical Flow Equation for motion estimation [10]. Whereas the optical flow velocity is related to the projection of 3D velocity vectors of objects in the scene, here the texture gradient velocity is related to the 3D coordinates of the surface where the texture lies.

To recover the 3D surface coordinates from the texture variations in the image, one must assume that the texture has some form of spatial homogeneity on the surface, so that the texture variations are only produced by the projective geometry. Perceptual results indicate that departure from isotropy is also an important cue in shape from texture, leading to biased slant estimates when the original texture is actually anisotropic [11]. Here, we address *Shape from Texture* without supposing any isotropy property. As natural though it may appear from a perceptual point of view, texture homogeneity on a general nondevelopable surface is very difficult to state mathematically.

- *M. Clerc is with the Centre d'Enseignement et de Recherche en Mathématiques, Informatique et Calcul Scientifique, Ecole Nationale des Ponts et Chaussées, Champs sur Marne 77455, Marne-la-Vallée Cedex 2, France. E-mail: maureen@cermics.enpc.fr.*
- *S. Mallat is with the Centre de Mathématiques Appliquées, Ecole Polytechnique, 91128 Palaiseau Cedex, France. E-mail: mallat@cmapx.polytechnique.fr.*

Manuscript received 6 Dec. 2000; revised 30 July 2001; accepted 28 Aug. 2001.
Recommended for acceptance by A. Khotanzad.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 113245.

We can therefore distinguish two independent theoretical subproblems nested in the recovery of 3D surface coordinates from the texture gradient. The first one is purely geometrical and amounts to understanding how the change of metric between the 3D surface and the image plane, due to the projection (either orthographic or perspective) relates to the surface coordinates. We call *deformation gradient* the relative change of this metric within the observed image. For instance, a planar surface viewed under an orthographic projection has a deformation gradient equal to zero. This is not true under a perspective projection, because the foreshortening is not the same throughout the image. The geometrical issues pertaining to *Shape from Texture* have been formalized by Gårding [2] and further analyzed by Malik and Rosenholtz [5], who establish the relationship between the deformation gradient and local surface shape parameters. The 3D coordinates of the surface can then easily be inferred, up to a scaling factor. The second subproblem concerns texture modeling. We mentioned that a homogeneity condition must be imposed so that the texture gradient is only produced by the geometrical deformation gradient. If this is the case, we show that the velocity term of the *Texture Gradient Equation* is equal to the deformation gradient, which can thus be calculated. The solution of the Shape from Texture problem can then be computed in two steps:

- Estimating the deformation gradient from the *Texture Gradient Equation*.
- Measuring the 3D surface coordinates from the deformation gradient.

The paper is organized as follows: In Section 2, we detail the model used for *Shape from Texture*. We focus our attention on developable surfaces, for which the texture homogeneity condition can be stated quite simply. In Section 3.1, we introduce the *Texture Gradient Equation* in the 1D case, after observing that wavelets migrate in the position-scale parameter space under an affine transform. Section 3.2 establishes the *Texture Gradient Equation* in 2D. Wavelets are now replaced by warplets, which are especially designed to migrate in the position-warping parameter space under a 2D affine transform. In Section 4, we analyze the statistical issues involved with the Texture Gradient Equation. The consistency of the deformation gradient estimator is proved in 1D, and the corresponding algorithm is illustrated with numerical results. Section 5 presents our Shape from Texture algorithm, with examples on photographs. Last, we propose a new homogeneity condition, based on the *Texture Gradient Equation*, which generalizes the homogeneity condition of Section 2 to general surfaces. This paper is oriented towards modeling and algorithms; although we state some mathematical results, we refer to [12] for their detailed proofs.

2 Shape from Texture Model

We assume that the surface has a Lambertian reflectance distribution. This supposes the texture to be "painted" on the surface and to have neither rugosity nor self-occlusions. With a Lambertian assumption and under perspective projection, the image intensity at position x in the image is related to the reflectance $\tilde{R}$ of the surface in the scene by

$$I(x) = a(x)\,\tilde{R}(p(x)), \tag{1}$$

where $a(x)$ is a multiplicative shading term and $p(x)$ is the perspective backprojection (Fig. 1). For instance, if the light is coming from a point source in direction $\vec{s}$ and if $\vec{n}$ is the surface normal, then $a(x) = \vec{n}(p(x)) \cdot \vec{s}(p(x))$ [13], [14].

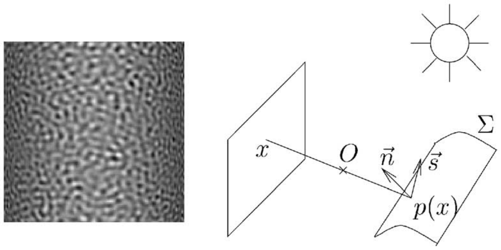


Fig. 1. Perspective image of a textured cylinder (left). Image formation (right): each position x in the image corresponds to a point $p(x)$ at the intersection between the surface Σ and the light ray connecting x and the optical center O. Vectors $\vec{n}$ and $\vec{s}$, respectively, represent the surface normal and the light source direction.

We use a stochastic model: The surface reflectance $\tilde{R}$ is the realization of a random process, supported on $\Sigma \subset \mathbb{R}^3$, and taking its values in $\mathbb{R}$. The image intensity I is also a random process, supported on $\mathbb{R}^2$. As explained in the introduction, we need to make a homogeneity assumption on the texture so that its gradient in the image is only a consequence of the geometrical deformation gradient. We first study the Lambertian model (1) when Σ is a developable surface, in which case the homogeneity condition on the texture $\tilde{R}$ can be stated in relatively simple terms.

A developable surface Σ (i.e., with zero Gaussian curvature) can be unfolded isometrically into a portion of a plane [15], thus defining a mapping from each position $p(x) \in \Sigma$ onto $d(x) \in \mathbb{R}^2$. A 2D stochastic process R on $\mathbb{R}^2$ can then be defined by $R(d(x)) = \tilde{R}(p(x))$. In the developable case, model (1) therefore simply becomes

$$I(x) = a(x)\,R(d(x)).$$

We say that the original texture $\tilde{R}$ is homogeneous if R is a wide-sense stationary process, i.e., satisfies $\mathbf{E}\{R(x)\} = \mathbf{E}\{R(0)\}$, and

$$\mathbf{E}\{R(x)\,R(x+\tau)\} = C(\tau). \tag{2}$$

In this case,

$$\mathbf{E}\big\{I^2(x)\big\} = a^2(x)\,\mathbf{E}\big\{R^2(d(x))\big\} = a^2(x)\,C(0).$$

The shading term $a(x)$ can thus be estimated up to a multiplicative constant from the second moment of the image $\mathbf{E}\big\{I^2(x)\big\}$. *Shape from Shading* studies shape recovery from this shading term only [16]. Here, we concentrate on the texture distortion and, hence, compensate for illumination changes. The estimation of $\mathbf{E}\big\{I^2(x)\big\}$ from a single realization of $I(x)$ is performed with a local averaging computed by convolving I^2 with with a Gaussian filter. We then calculate $\mathbf{E}\big\{I^2(x)\big\}^{-1/2} I(x)$ to remove the shading term $a(x)$. The image resulting from this local contrast

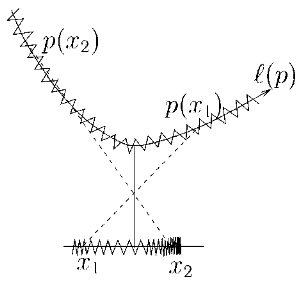


Fig. 2. A point x in the image backprojects to a point $p(x)$ on Σ, whose arc-length is $d(x) = \ell(p(x))$. The zig-zag line represents a stationary texture covering Σ. When projected onto the image, it gives rise to a nonstationary process $I(x) = R(d(x))$.

normalization is still denoted $I(x)$ for convenience. The model therefore simplifies to

$$I(x) = R(d(x)). \tag{3}$$

We assume that the surface Σ is $\mathbf{C}^3$ and, in particular, does not contain any occluding contour. Hence, $d(x)$ is $\mathbf{C}^3$ and invertible.

The Jacobian matrix of $d(x)$ in an orthonormal basis $(\vec{x}_1, \vec{x}_2)$ of $\mathbb{R}^2$ is given by

$$J_d(x) = \begin{pmatrix} \frac{\partial d_1(x)}{\partial x_1} & \frac{\partial d_1(x)}{\partial x_2} \\ \frac{\partial d_2(x)}{\partial x_1} & \frac{\partial d_2(x)}{\partial x_2} \end{pmatrix}. \tag{4}$$

Since Σ is developable and can be isometrically unfolded into a portion of a plane, $J_d(x)$ represents the change of metric between the surface Σ and the image plane. We call *deformation gradient* the relative variations of the Jacobian in directions x_1 and x_2. The deformation gradient is thus represented by the two matrices, for $k = 1, 2$:

$$J_d(x)^{-1} \partial_{x_k} J_d(x) = \begin{pmatrix} \frac{\partial d_1(x)}{\partial x_1} & \frac{\partial d_1(x)}{\partial x_2} \\ \frac{\partial d_2(x)}{\partial x_1} & \frac{\partial d_2(x)}{\partial x_2} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial^2 d_1(x)}{\partial x_k \partial x_1} & \frac{\partial^2 d_1(x)}{\partial x_k \partial x_2} \\ \frac{\partial^2 d_2(x)}{\partial x_k \partial x_1} & \frac{\partial^2 d_2(x)}{\partial x_k \partial x_2} \end{pmatrix}. \tag{5}$$

As we shall see in Section 5.2, it is possible to recover the three-dimensional coordinates of Σ (up to a scaling factor) from the deformation gradient. The main difficulty is to estimate this deformation gradient given one realization of $I(x) = R(d(x))$.

3 Texture Gradient Equation

3.1 In 1D: Scalogram Migration

For the sake of simplicity, let us start with a 1D *Shape from Texture* problem, in which the shape Σ to be recovered is a curve. Let R denote the "reflectance" of Σ, parameterized by arc-length ℓ: $R(\ell)$ is assumed stationary and is depicted in Fig. 2 by a regular zig-zag line along the curve. In a 1D perspective model, a pixel at position x in the image backprojects onto a position $p(x)$ on Σ, whose arc-length is $\ell(p(x))$. The image $I(x)$ can therefore be viewed as the deformation of a stationary process R by $d(x) = \ell(p(x))$:

$$I(x) = R(d(x)).$$

Let ψ be a function with zero average, whose support is in $[-1, 1]$. A local analysis of the image is performed by computing the inner product of $I(x)$ with

$$\psi_{u,s}(x) = \frac{1}{s}\, \psi\Big(\frac{x-u}{s}\Big), \tag{6}$$

whose support is in $[u - s, u + s]$. This inner product $\langle I, \psi_{u,s} \rangle$ is called a *wavelet coefficient* of I at position u and scale s [17] and we call *scalogram* of I the variance of this wavelet coefficient:

$$w(u, s) = \mathbf{E}\{\left|\langle I, \psi_{u,s} \rangle\right|^2\}.$$

If R is stationary, then we easily verify that, for a fixed scale s, its scalogram is independent of u:

$$\frac{d}{du}\, \mathbf{E}\{\left|\langle R, \psi_{u,s} \rangle\right|^2\} = 0. \tag{7}$$

In Fig. 3a, the scalogram of R is displayed in a gray-level image as a function of u (horizontal axis) and of $\log s$ (vertical axis): It does not vary with u. On the other hand, the scalogram of I does in general depend on u, as can be seen in Fig. 3b.

We introduce the *Texture Gradient Equation*, which links together the partial derivatives of $w(u, s)$:

$$\partial_u w(u, s) - v(u, s)\, \partial_{\log s} w(u, s) = 0. \tag{8}$$

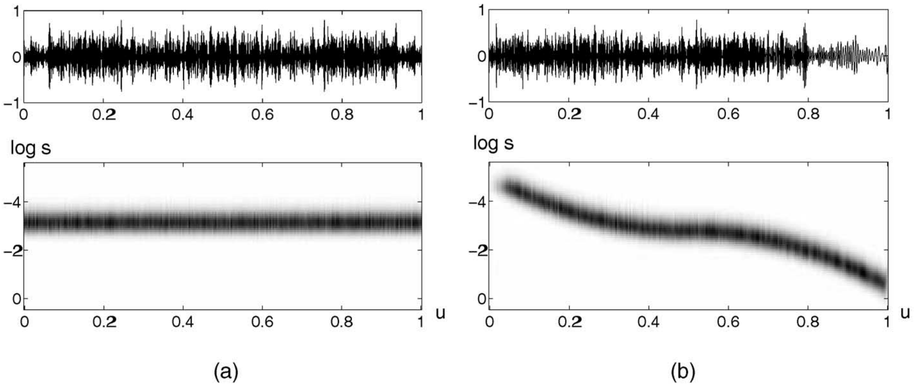


Fig. 3. (a) Top: A realization of stationary process $R(x)$. Bottom: scalogram $\mathbf{E}\{\left|\langle R, \psi_{u,s} \rangle\right|^2\}$. The horizontal and vertical axes represent u and $\log s$, respectively. Dark points indicate high amplitude. (b) Top: a realization of deformed process $I(x) = R(d(x))$. Bottom: scalogram $w(u, s) = \mathbf{E}\{\left|\langle I, \psi_{u,s} \rangle\right|^2\}$.

The velocity term $v(u,s)$ can be interpreted as a *texture gradient*: It measures how the image energy moves across scales according to the position in the image. The Texture Gradient Equation (8) is comparable to the Optical Flow Equation for motion estimation [10]. The conservation equation (7), expressing the stationarity of the original process R, is analogous to brightness constancy in Optical Flow. Appendix A.1 shows that, when the scale s is small relative to the scale of variation of the deformation, the scalograms of I and of R are related by a simple migration in the position-scale parameter space. The scalogram coefficient of I at position u and scale s corresponds to the scalogram coefficient of R at position $d(u)$ and scale $d'(u)s$:

$$w(u,s) = \mathbf{E}\{|\langle I, \psi_{u,s}\rangle|^2\} \approx \mathbf{E}\{|\langle R, \psi_{d(u),d'(u)s}\rangle|^2\}. \qquad (9)$$

As a consequence, Appendix A.1 shows that the texture gradient equation can be derived from the conservation equation (7). For s sufficiently small, the velocity $v(u,s)$ of the texture gradient equation does not depend upon s and is nearly equal to $d''(u)/d'(u)$, which is the one-dimensional equivalent of the deformation gradient (5):

$$v(u,s) \approx \frac{d''(u)}{d'(u)}. \qquad (10)$$

It is natural that $d(x)$ should appear under this form in (10), because with no additional assumption on the stationary process, the deformation is only specified up to an affine transform. Indeed, if $I(x) = R_1(d_1(x))$, where R_1 is stationary, and if $d_2(x) = \alpha d_1(x) + \beta$, then one can find another stationary process, $R_2(x) = R_1(\alpha x + \beta)$, such that $I(x) = R_2(d_2(x))$. The functions d_1 and d_2, which satisfy $d_1''/d_1' = d_2''/d_2'$, cannot be distinguished with the sole knowledge that I is obtained through the deformation of a stationary process.

Appendix A.1 derives (10) in a loose fashion, but a careful analysis of the higher-order terms gives the following result [12]: If the covariance of R, $C(\tau)$, defined in (2), satisfies

$$C(0) - C(\tau) = |\tau|^{\gamma}\, \eta(\tau), \qquad (11)$$

with $\gamma > 0$, $\eta(0) > 0$, and η continuously differentiable in a neighborhood of 0, then

$$(1 + O(s))\, \partial_u w(u,s) - \frac{d''(u)}{d'(u)} \partial_{\log s} w(u,s) = 0. \qquad (12)$$

The resolution error $O(s)$ tends to zero at least as fast as s, and can therefore be neglected at fine scales. Condition (11) on the covariance of R is quite weak and is satisfied by most correlation functions [18]. This proves that, at small scales $s \to 0$, the velocity of the *Texture Gradient Equation* converges to the deformation gradient.

A scalogram $w(u,s)$ is displayed on the bottom of Fig. 3b. The large amplitude coefficients of the scalogram are transported in the $(u; \log s)$ plane, with a velocity equal to the deformation gradient. Computing the deformation gradient from partial derivatives of the scalogram is thus in principle possible. There remains a difficulty: We only observe one realization of $I(x)$, from which we must estimate a scalogram, which involves estimating an ensemble average. Section $ focuses on this estimation problem.

3.2 In 2D: Warpogram Migration

A deformation can be locally approximated by an affine transformation, which is specified by a translation and a dilation. In 1D, the dilation parameter is a scalar, whereas, in 2D, it is a 2×2 matrix. Our 1D analysis of deformed stationary processes involved wavelets, constructed by translating and dilating a mother waveform. In 2D, wavelets are now replaced by *warplets*, constructed by applying a 2D affine transform to a mother waveform. Warplets are thus clearly designed to migrate under a 2D affine transform.

Let $\psi(x_1, x_2)$ be a function having a zero average. A warplet $\psi_{u,S}$ is indexed by its position $u = (u_1, u_2)$ and by a 2×2 invertible matrix

$$S = \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix},$$

which deforms the support of ψ:

$$\psi_{u,S}(x) = \det S^{-1} \psi(S^{-1}(x-u)).$$

For instance, modulated Gaussians, called Gabor functions,

$$\psi(x_1, x_2) = \exp\left(-\frac{x_1^2 + x_2^2}{2}\right) \exp(ikx_1),$$

which nearly have a zero average if $k \gg 1$, have widely been used for texture discrimination [19]. Fig. 4 displays three warplets $\psi_{u,S}$ constructed with a Gabor function. The four parameters of S control the size and shape of the support, and the spatial frequency of the oscillations.

A *warplet coefficient* of $I(x)$ at position u and warping matrix S is defined as the inner product $\langle I, \psi_{u,S}\rangle$. The variance of this warplet coefficient is called the *warpogram* of I:

$$w(u,S) = \mathbf{E}\left\{|\langle I, \psi_{u,S}\rangle|^2\right\}.$$

We introduce a *Texture Gradient Equation* which relates the partial derivatives of $w(u,S)$ and generalizes the 1D equation (8):

$$\partial_{u_k} w(u,S) - \sum_{i,j=1}^{2} v_{ij}^k(u,S)\, a_{ij}(u,S) = 0 \ \text{ for } k = 1,2, \qquad (13)$$

where the a_{ij} are the coefficients of the following matrix product

$$\begin{pmatrix} a_{11}(u,S) & a_{12}(u,S) \\ a_{21}(u,S) & a_{22}(u,S) \end{pmatrix} = \begin{pmatrix} \frac{\partial w(u,S)}{\partial s_{11}} & \frac{\partial w(u,S)}{\partial s_{12}} \\ \frac{\partial w(u,S)}{\partial s_{21}} & \frac{\partial w(u,S)}{\partial s_{22}} \end{pmatrix} \times \begin{pmatrix} s_{11} & s_{21} \\ s_{12} & s_{22} \end{pmatrix}.$$

The velocity terms $v_{ij}^k(u,S)$ define a texture gradient which measures how the image energy moves across the position-warping parameter space.

If $I(x) = R(d(x))$ and if $d(x)$ is regular, then for $\det S$ small enough, Appendix A.2 shows that the warpograms

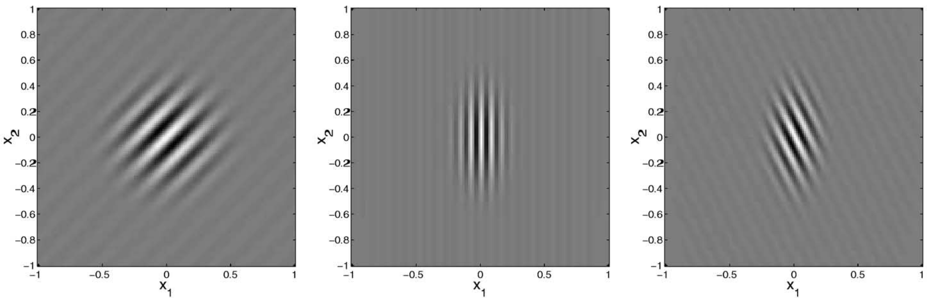


Fig. 4. Warplets $\psi_{0,S}(x_1,x_2)$ for three different warping matrices S. Light and dark pixels indicate positive and negative values, respectively.

of I and of R are related by a migration property in the six-dimensional position-warping parameter space $(u_1,u_2,s_{11},s_{12},s_{21},s_{22})$:

$$w(u,S)=\mathbf{E}\left\{|\langle I,\psi_{u,S}\rangle|^2\right\}\approx\mathbf{E}\left\{|\langle R,\psi_{d(u),J_d(u)S}\rangle|^2\right\}. \tag{15}$$

As a consequence, Appendix A.2 shows that if $\det S$ is small, then the texture gradient is nearly independent of S and nearly equal to the deformation gradient defined in (5). Let us denote for $k=1,2$:

$$J_d(u)^{-1}\partial_{u_k}J_d(u)=\begin{pmatrix} g^k_{11}(u) & g^k_{12}(u)\\ g^k_{21}(u) & g^k_{22}(u)\end{pmatrix}, \tag{16}$$

we then have

$$\begin{pmatrix} v^k_{11}(u,S) & v^k_{12}(u,S)\\ v^k_{21}(u,S) & v^k_{22}(u,S)\end{pmatrix}\approx\begin{pmatrix} g^k_{11}(u) & g^k_{12}(u)\\ g^k_{21}(u) & g^k_{22}(u)\end{pmatrix}. \tag{17}$$

In [12], we prove more precisely that when $\det S\to 0$, the velocity of the Texture Gradient Equation, if uniquely defined, converges to the deformation gradient. Section 4 shows how to use this property in order to estimate the deformation gradient from warplet coefficients of the image.

4 Consistency of Statistical Estimation

For the sake of simplicity, this section focuses on the 1D estimation problem. We want to estimate $d''(u)/d'(u)$, which, according to (10), is almost equal to the velocity of the Texture Gradient Equation (8) at small scales. For this, we need to estimate partial derivatives $\partial_u w(u,s)$ and $\partial_{\log s} w(u,s)$ from one realization of I. Since $w(u,s)=\mathbf{E}\left\{|\langle I,\psi_{u,s}\rangle|^2\right\}$, for a generic variable a representing u or $\log s$, one can verify that

$$\partial_a w(u,s)=2Re\left[\mathbf{E}\left\{\langle I,\psi_{u,s}\rangle\langle I,\partial_a\psi_{u,s}\rangle^*\right\}\right] \tag{18}$$

with

$$\partial_u\psi(x)=\psi'(x), \tag{19}$$

$$\partial_{\log s}\psi(x)=-\psi(x)-x\,\psi'(x). \tag{20}$$

Let ψ be a compactly supported wavelet with m vanishing moments, i.e., whose inner product with any polynomial of degree $k<m$ vanishes:

$$\int\psi(x)\,x^k\,dx=0.$$

Then, $\partial_u\psi$ and $\partial_{\log s}\psi$ are also compactly supported wavelets and an integration by parts shows that they respectively have $m+1$ and m vanishing moments. Expression (18) indicates that $\partial_a w(u,s)$ simply depends on the wavelet transform of I with mother wavelets ψ and $\partial_a\psi$. Appendix B.1 details the implementation of these wavelet transforms.

In view of (18), one could use the following unbiased estimator to estimate $\partial_a w(u,s)$ from a single realization of I:

$$\widehat{\partial_a w}(u,s)=2Re\left[\langle I,\psi_{u,s}\rangle\langle I,\partial_a\psi_{u,s}\rangle^*\right]. \tag{21}$$

Unfortunately, the variance of $\widehat{\partial_a w}(u,s)$ is typically larger than $|\partial_a w(u,s)|$, which leads to an unacceptably large mean-squared error. To reduce the variance, we compute a weighted average of (12), by convolution with a continuous, positive window function $h_\Delta(x)=\Delta^{-1}h(\Delta^{-1}x)$ supported in $[-\Delta,\Delta]$:

$$(1+O(s))\partial_u w(\cdot,s)*h_\Delta(u)=\left(\frac{d''(\cdot)}{d'(\cdot)}\partial_{\log s}w(\cdot,s)\right)*h_\Delta(u). \tag{22}$$

We assume that d is $\mathbf{C}^3$ and that $d'(u)\geq\eta>0$. For $x\in[u-\Delta,u+\Delta]$, one can verify that

$$d''(x)/d'(x)=d''(u)/d'(u)+O(\Delta). \tag{23}$$

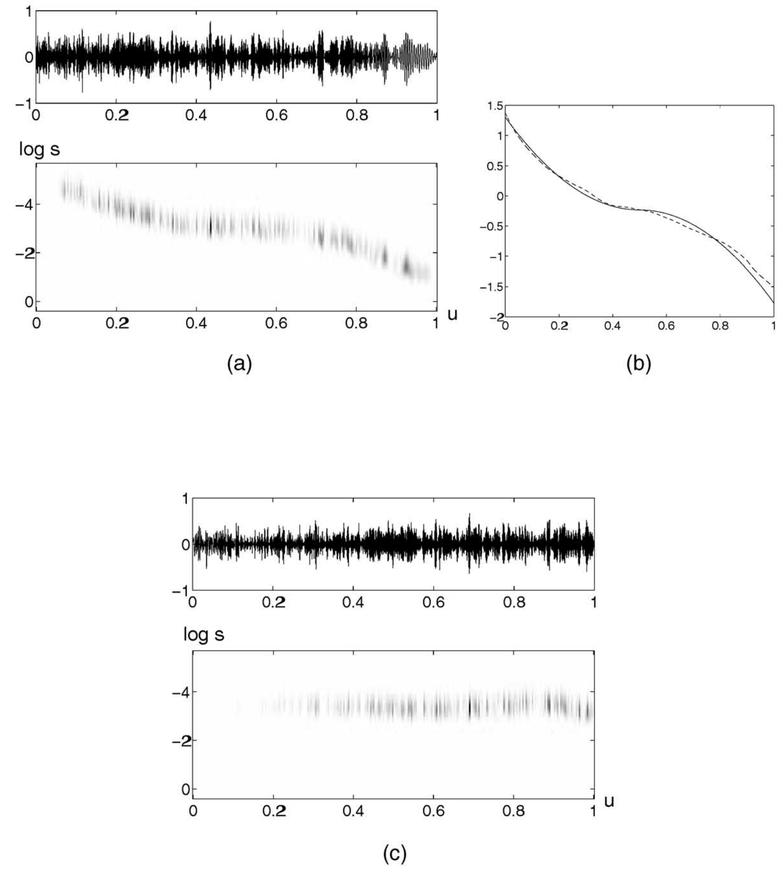
Replacing (23) inside (22), for $\Delta>s$, we obtain

$$\frac{d''(u)}{d'(u)}=\frac{\partial_u w(\cdot,s)*h_\Delta(u)}{\partial_{\log s}w(\cdot,s)*h_\Delta(u)}+O(\Delta). \tag{24}$$

The error $O(\Delta)$ can be interpreted as a bias due to the smoothing over a width Δ. Recalling the estimator $\widehat{\partial_a w}(u,s)$ defined in (21), (24) suggests the following estimator for $\frac{d''(u)}{d'(u)}$:

$$\widehat{\frac{d''(u)}{d'(u)}}=\frac{\widehat{\partial_u w}(\cdot,s)*h_\Delta(u)}{\widehat{\partial_{\log s}w}(\cdot,s)*h_\Delta(u)}. \tag{25}$$

If the signal is measured at a resolution N, the wavelet transform can be calculated up to the scale $s=N^{-1}$. To optimize the estimation, we must adjust Δ so that the bias term is of the same order as the variance of the estimator.



(a)

(b)

(c)

Fig. 5. (a) A realization of $I(x)$ and the squared amplitude of its wavelet coefficients $|\langle I, \psi_{u,s}\rangle|^2$. (b) Estimated deformation $\log(\hat{d}')$ in dashed and exact deformation $\log(d')$ in full line. (c) Stationarized signal $\hat{R}(x) = I(\hat{d}^{-1}(x))$ and $|\langle \hat{R}, \psi_{u,s}\rangle|^2$.

We have proved in [12] that, for $s = N^{-1}$ and for $\Delta = N^{-1/5}$, if R is a Gaussian process with a covariance which satisfies (11) for a certain $\gamma > 0$ and if the number of vanishing moments of ψ is larger than $(2\gamma + 1)/4$, then

$$\mathrm{Prob}\left\{\left|\widehat{\frac{d''(u)}{d'(u)}} - \frac{d''(u)}{d'(u)}\right| \leq 2\,(\log N)\, N^{-1/5}\right\}\xrightarrow[N\to\infty]{} 1.$$

In other words, $\widehat{\frac{d''(u)}{d'(u)}}$ tends to $\frac{d''(u)}{d'(u)}$ with a probability that tends to 1 when the resolution N goes to infinity.

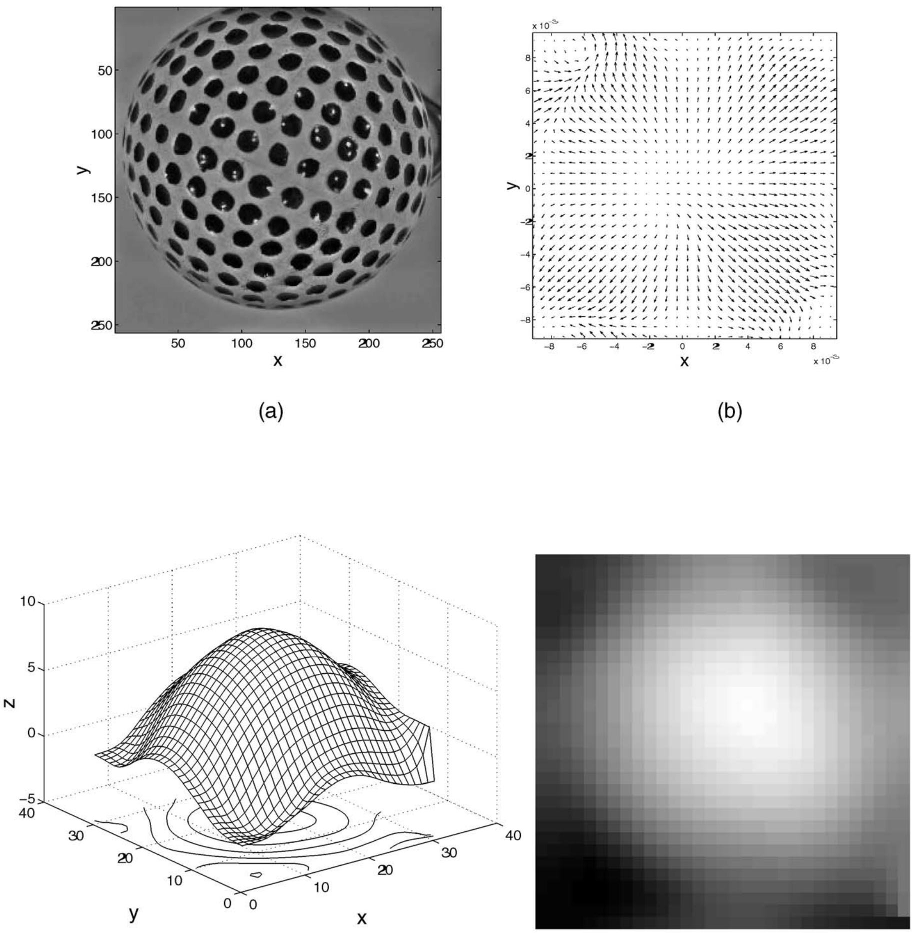
This consistency result guarantees the convergence of the estimator, but, in practice, for a fixed resolution, averaging the smoothed estimators across several scales improves the result. Therefore, we propose a modified estimator for $\frac{d''(u)}{d'(u)}$:

$$\widehat{\frac{d''(u)}{d'(u)}} = \frac{\sum_i \widehat{\partial_u w \cdot}, s_i) * h_\Delta(u)}{\sum_i \widehat{\partial_{\log s}} w(\cdot, s_i) * h_\Delta(u)}. \qquad (26)$$

In the example of Fig. 5, the signal $I(x)$, displayed in (a), is sampled over $[0,\,1]$ at a resolution $N = 4{,}096$ and we choose six scales in a range $\log s \in [-4.5,\,-3]$ for which the signal has nonnegligible energy. The partial derivatives of the scalogram, $\widehat{\partial_u w}(\cdot, s_i)$ and $\widehat{\partial_{\log s}} w(\cdot, s_i)$, are computed with (21) using wavelet coefficients $\langle I, \psi_{u,s_i}\rangle$, $\langle I, \partial_u \psi_{u,s_i}\rangle$, and $\langle I, \partial_{\log s} \psi_{u,s_i}\rangle$ that are calculated with a FFT-based procedure explained in Appendix B.1. The smoothing kernel is $h(x) = 1 - |x|$ for x inside $[-1,\,1]$ and zero outside this interval. The estimator $\widehat{\frac{d''(u)}{d'(u)}}$ is computed with (26), for $\Delta = N^{-1/5}$ and $i = 1, , 6$. The overall algorithm requires $O(N \log N)$ operations. It is important to note that only the wavelet coefficients corresponding to six different scales need to be computed. The whole wavelet transform plane is displayed in Figs. 5a and 5c for an explanatory purpose only. Fig. 5b shows $\log d'$ (full line) and its estimate $\widehat{\log d'}$ (dotted line), obtained by integrating $\widehat{d''/d'}$ and choosing the additive integration constant to satisfy $\int_0^1 \exp(\widehat{\log d'}) = \int_0^1 d'$. An estimate $\widehat{d}$ for the warping function can be obtained, up to an affine transformation, by integrating $\exp(\widehat{\log d'})$. It is then possible to stationarize I by computing $\hat{R}(x) = I(\widetilde{d}^{-1}(x))$. Fig. 5c displays $\hat{R}$: As expected, its wavelet transform remains nearly constant when u varies, modulo statistical fluctuations.

5 Application to Shape from Texture

We now turn to the estimation of shape from texture. Section 5.1 details the deformation gradient estimation from the warplet coefficients of the image when the deformation gradient is equal to the velocity of the 2D Texture Gradient



(a)

(b)

(c)

(d)

Fig. 6. (a) Original image with shading removed. (b) Normal vector computed. (c) Surface reconstructed from the normal vector. (d) Visualization of (c) as gray levels. Because (a) does not contain texture on its borders, there are errors in the border of the reconstructed surface (c).

Equation (13). Section 5.2 presents shape recovery from the deformation gradient and, last, Section 5.3 gives a condition on the texture, for general (i.e., not necessarily developable) surfaces, so that the deformation gradient is indeed equal to the velocity of the 2D Texture Gradient Equation at small scales.

5.1 Deformation Gradient Estimation

As explained in Section 2, the shading term is removed by estimating $\mathbf{E}\{I^2(x)\}$ from a single realization $I(x)$. The estimator computes $I^2 * G_\sigma(x)$, where G_σ is a 2D Gaussian whose variance σ is adjusted to the scale of shading variations. The shading term is nearly removed by dividing $I(x)$ by $(I^2 * G_\sigma)^{1/2}(x)$. The images in Figs. 6a and 7a are the result of this preprocessing step.

If the velocity terms $v^k_{ij}(u, S)$ of the Texture Gradient Equation (13) are nearly equal to the deformation gradient terms $g^k_{ij}(u)$ at small scales, as in (17), then the Texture Gradient Equation can be rewritten for $k = 1, 2$

$$[a_{11}(u,S), a_{12}(u,S), a_{21}(u,S), a_{22}(u,S)] \begin{bmatrix} g^k_{11}(u) \\ g^k_{12}(u) \\ g^k_{21}(u) \\ g^k_{22}(u) \end{bmatrix} \approx \partial_{u_k} w(u,S), \tag{27}$$

where the $a_{ij}(u, S)$ have been defined in (14). A collection of equations ((27)) corresponding to P different warping matrices $\{S_i\}_{i=1,,P}$ are concatenated in a linear system

$$\begin{pmatrix} a_{11}(u,S_1) & a_{12}(u,S_1) & a_{21}(u,S_1) & a_{22}(u,S_1) \\ a_{11}(u,S_2) & a_{12}(u,S_2) & a_{21}(u,S_2) & a_{22}(u,S_2) \\ \vdots & \vdots & \vdots & \vdots \\ a_{11}(u,S_P) & a_{12}(u,S_P) & a_{21}(u,S_P) & a_{22}(u,S_P) \end{pmatrix} \begin{pmatrix} g^k_{11}(u) \\ g^k_{12}(u) \\ g^k_{21}(u) \\ g^k_{22}(u) \end{pmatrix} \approx \begin{pmatrix} \partial_{u_k} w(u,S_1) \\ \partial_{u_k} w(u,S_2) \\ \vdots \\ \partial_{u_k} w(u,S_P) \end{pmatrix}. \tag{28}$$

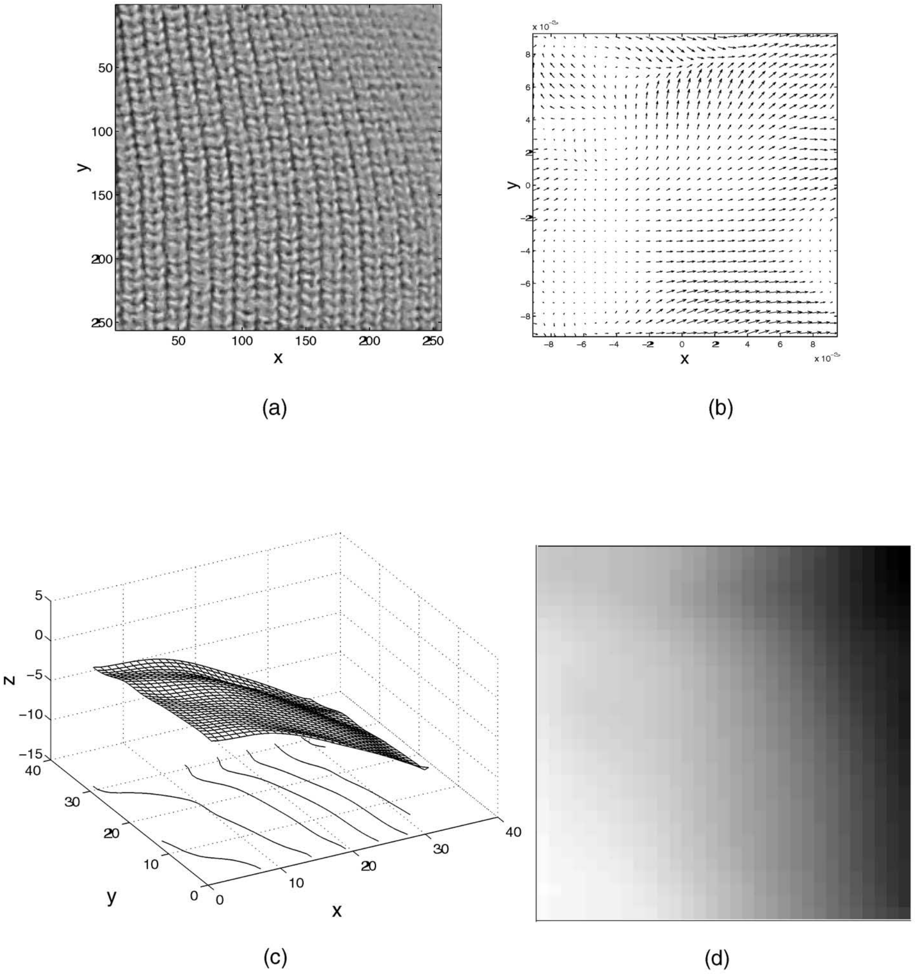


Fig. 7. (a) Original image with shading removed. (b) Normal vector computed. (c) Surface reconstructed from the normal vector. (d) Visualization of (c) as gray levels.

Because we only observe one realization of $I(x)$, as in 1D, expected values are estimated by averaging (28) with a 2D window $h_\Delta(x) = \Delta^{-2} h(\Delta^{-1}x)$ supported inside $[-\Delta, \Delta]^2$.

Since $d(x)$ is $\mathbf{C}^3$, one can verify that convolving (28) with h_Δ yields

$$\begin{pmatrix} \overline{a_{11}}(u,S_1) & \cdots & \overline{a_{22}}(u,S_1)\\ \vdots & \vdots & \vdots \\ \overline{a_{11}}(u,S_P) & \cdots & \overline{a_{22}}(u,S_P)\end{pmatrix}\begin{pmatrix} g^k_{11}(u)\\ g^k_{12}(u)\\ g^k_{21}(u)\\ g^k_{22}(u)\end{pmatrix} \qquad (29)$$
$$= \begin{pmatrix}\partial_{u_k} w(\cdot,S_1) * h_\Delta(u)\\ \vdots \\ \partial_{u_k} w(\cdot,S_P) * h_\Delta(u)\end{pmatrix} + O(\Delta),$$

where

$$\left(\overline{a_{lm}}(u,S)\right)_{1\le l,m\le 2} = \left(\partial_{s_{ij}} w(\cdot,S) * h_\Delta(u)\right)_{1\le i,j\le 2} \times S^T.$$

Using $\partial_a w(u,S) = 2\,Re\big[\mathbf{E}\{\langle I,\psi_{u,S}\rangle\langle I,\partial_a\psi_{u,S}\rangle^*\}\big]$, we estimate $\partial_a w(\cdot,S) * h_\Delta$ with $\widehat{\partial_a w}(\cdot,S) * h_\Delta$, where

$$\widehat{\partial_a w}(u,S) = 2\,Re\big[\langle I,\psi_{u,S}\rangle\langle I,\partial_a\psi_{u,S}\rangle^*\big]. \qquad (30)$$

Let us normalize the image support to $[0,1]^2$. If I has $N^2 = 256^2$ pixels, we can only compute the warpogram for warplets $\psi_{u,S}$ whose support in any direction is larger than N^{-1}. Therefore, we require all the eigenvalues of S to be greater than N^{-1}.

Denoting

$$\left(\widehat{a_{lm}}(u,S)\right)_{1\le l,m\le 2} = \left(\widehat{\partial_{s_{ij}} w}(\cdot,S) * h_\Delta(u)\right)_{1\le i,j\le 2} \times S^T,$$

(29) suggests estimating the deformation gradient by inverting the linear system

$$\begin{pmatrix} \widehat{a_{11}}(u,S_1) & \cdots & \widehat{a_{22}}(u,S_1)\\ \vdots & \vdots & \vdots \\ \widehat{a_{11}}(u,S_P) & \cdots & \widehat{a_{22}}(u,S_P)\end{pmatrix}\begin{pmatrix} \widehat{g^k_{11}}(u)\\ \widehat{g^k_{12}}(u)\\ \widehat{g^k_{21}}(u)\\ \widehat{g^k_{22}}(u)\end{pmatrix} \qquad (31)$$
$$= \begin{pmatrix}\widehat{\partial_{u_k} w}(\cdot,S_1) * h_\Delta(u)\\ \vdots \\ \widehat{\partial_{u_k} w}(\cdot,S_P) * h_\Delta(u)\end{pmatrix}.$$

To obtain at least as many equations as unknowns, one must choose warplets corresponding to at least four

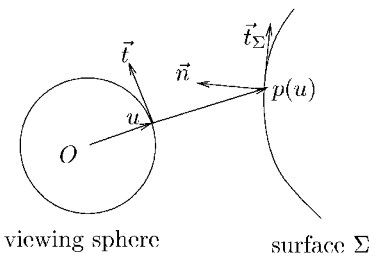


Fig. 8. The slant-tilt frame field of Σ is $(\vec{n}, \vec{t}_\Sigma, \vec{b}_\Sigma)$, where $\vec{b}_\Sigma = \vec{n} \times \vec{t}_\Sigma$ is tangent to the surface Σ, and perpendicular to the plane of the figure.

different warping matrices S and, in practice, we shall use even more. The estimators $\widehat{g^k_{ij}}$ are then calculated by singular value decomposition. The choice of matrices S must be adjusted so that the system (31) is not degenerated. However, for some shape and texture configurations, even with an arbitrarily large number of different warping matrices, the system may remain underdetermined, as in the aperture problem for Optical Flow. Let us take the example of the cylindrical shape in Fig. 1, which is curved in the horizontal direction. If it is covered with a texture with horizontal stripes, then the matrix on the left-hand side of (31) will have a rank strictly smaller than four, which does not allow recovery of the deformation gradient. With such a texture, one cannot characterize the cylindrical shape and, for example, distinguish it from a plane.

We parameterize S under the form:

$$S = R_{\theta_1} \begin{pmatrix} s_1 & 0 \\ 0 & s_2 \end{pmatrix} R_{\theta_2}.$$

In Figs. 6 and 7, calculations are performed with six angles $\theta_1 \in \{l\pi/6, l = 0, \dots, 5\}$ and with $\theta_2 = 0$. Four scale couples (s_1, s_2) are selected, ensuring that $|\langle I, \psi_{u,S}\rangle|$ is large to avoid numerical instabilities. Ideally, the set of warping matrices $\{S_i, i = 1 \dots, P\}$ should be selected adaptively according to u, but we use the same set throughout the image. This explains why we use a total of $P = 6 \times 4 = 24$ warping matrices, instead of the minimal value of 4. As in 1D, it is important to note that warplet coefficients need only be calculated for relatively few scaling matrices, compared to a full warplet transform. For each scaling matrix, we calculate $\langle I, \psi_{u,S}\rangle$, $\langle I, \partial_{u_k}\psi_{u,S}\rangle$, and $\langle I, \partial_{s_{ij}}\psi_{u,S}\rangle$ with the FFT, as explained in Appendix B.2. There are thus a total of $168 = 24 \times (1 + 2 + 4)$ sets of warplet coefficients to compute. We calculate $\widehat{\partial_a w}(u, S)$ with (30) and convolve it with h_Δ. In our examples, we use $\Delta = N^{-1/5} = 0.33$, which must be compared to the image support which is $[0, 1]^2$. Finally, the linear least-squares solution of (31) is computed with a singular value decomposition [20].

Proving the statistical consistency of the deformation gradient estimator in 2D is far more complicated than in 1D and has not been done in the most general setting. However, for a separable deformation $d(x_1, x_2) = (d_1(x_1), d_2(x_2))$, the 1D consistency results of Section 4 extend automatically to 2D. Figs. 6 and 7 show that good numerical results are also obtained for nonseparable deformations.

5.2 Recovering the 3D Surface Coordinates

Our goal is now to calculate the normal vector $\vec{n}$ to the surface, from the deformation gradient. We first recall the geometrical setting presented in [2]. The basic *Shape from Texture* geometry assumes the image to be projected onto a viewing sphere, as shown in Fig. 8. The perspective backprojection p maps the viewing sphere to the surface Σ. The *tilt* direction $\vec{t}$ is defined as the direction of maximum change of the distance $\|\overrightarrow{Op(u)}\|$. Defining $\vec{b} = \overrightarrow{Ou} \times \vec{t}$, we obtain an orthonormal frame field $(\overrightarrow{Ou}, \vec{t}, \vec{b})$ of the viewing sphere. The differential of the backprojection transforms $\vec{t}$ and $\vec{b}$ into two orthogonal vectors, which are denoted $\vec{t}_\Sigma$ and $\vec{b}_\Sigma$ after being unit-normalized. The resulting orthonormal frame field $(\vec{n}, \vec{t}_\Sigma, \vec{b}_\Sigma)$ of Σ is called the slant-tilt frame field. The *slant* is the angle σ between $\vec{n}$ and $\overrightarrow{Op(u)}$. The variations of the surface normal $\vec{n}$ depend upon the surface curvature, and are specified by

$$\begin{pmatrix} \nabla_{\vec{t}_\Sigma}\vec{n} \\ \nabla_{\vec{b}_\Sigma}\vec{n} \end{pmatrix} = \begin{pmatrix} \kappa_t & \tau \\ \tau & \kappa_b \end{pmatrix} \begin{pmatrix} \vec{t}_\Sigma \\ \vec{b}_\Sigma \end{pmatrix},$$

where κ_t and κ_b are the normal curvatures of the surface in the tilt and perpendicular directions, respectively, and τ is the geodesic torsion in the tilt direction [2]. In the rest of the paper, we consider the deformation gradient to be measured on the image plane and not on the viewing sphere. The *gaze* transformation, which maps one to another, can actually be approximated by the identity as long as the surface Σ remains close to the optical axis of the camera. If this is not the case, a correction term must be taken into account ([5], Appendix A.2).

If $(\vec{x}_1, \vec{x}_2)$ is an orthonormal basis of the image plane, the *tilt* angle θ is such that the projection of $\vec{t}$ on the image plane is given by $\cos\theta\,\vec{x}_1 + \sin\theta\,\vec{x}_2$. We define

$$R_\theta = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

According to [5], [21], the deformation gradient (5) is related to local surface parameters by

$$J_d(u)^{-1}\partial_{x_1} J_d(u) = R_\theta \left(M_t(u)\cos\theta - M_b(u)\sin\theta\right) R_{-\theta}, \qquad (32)$$

$$J_d(u)^{-1}\partial_{x_2} J_d(u) = R_\theta \left(M_t(u)\sin\theta + M_b(u)\cos\theta\right) R_{-\theta}, \qquad (33)$$

where $M_t(u)$ and $M_b(u)$ are given by

$$M_t(u) = \tan\sigma \begin{pmatrix} 2 + \|\overrightarrow{Op(u)}\|\kappa_t/\cos\sigma & \|\overrightarrow{Op(u)}\|\tau \\ 0 & 1 \end{pmatrix}, \qquad (34)$$

$$M_b(u) = \tan\sigma \begin{pmatrix} \|\overrightarrow{Op(u)}\|\tau & \|\overrightarrow{Op(u)}\|\kappa_b\cos\sigma \\ 1 & 0 \end{pmatrix}. \qquad (35)$$

In order to recover local surface shape from the deformation gradient, the tilt and slant must be estimated. Algebraic manipulations detailed in Appendix C show that this can be achieved via a simple one-dimensional minimization. From the tilt and slant, we then compute the normal

$$\vec{n} = \cos\sigma\,\overrightarrow{Ou} - \sin\sigma\,\vec{t},$$

on a grid whose resolution is 16 times smaller than the image resolution. This is due to the fact that each vector is derived from the estimated deformation gradient which depend on averaged warplet coefficients. Let the 3D coordinates of $\vec{n}$ be (n_1, n_2, n_3). A needle map, displayed in Figs. 6b and 7b, is given by the 2D vector

$$\vec{n}' = (n'_1, n'_2) = (n_1/n_3, n_2/n_3).$$

In the golf-ball example of Fig. 6b, since the image border does not contain any texture, we imposed that $n'_1 = n'_2 = 0$ at the image corners.

The needle map can be integrated to obtain the depth $f(x_1, x_2)$ of a point at position (x_1, x_2), up to a multiplicative scaling factor. Noticing that $\partial_{x_1} f = n'_1$ and $\partial_{x_2} f = n'_2$, it is clear that f is the solution of $\Delta f = \partial_{x_1} n'_1 + \partial_{x_2} n'_2$ [13]. This equation is solved with a standard finite difference scheme. The reconstructed surface depth $f(x_1, x_2)$ is plotted in Figs. 6c, 6d, 7c, and 7d. In the overall algorithm, the most significant amount of computation is devoted to calculating the warplet coefficients for different warping matrices S, each requiring $O(N^2 \log N)$ operations.

5.3 From Developable to Nondevelopable Surfaces

In Section 2, we modeled the image of a textured surface under perspective projection as

$$I(x) = a(x)\,\tilde{R}(p(x)).$$

If Σ is a developable surface, then the reflectance $\tilde{R}$, defined on Σ, can be "flattened" into a 2D process:

$$I(x) = a(x)\,\tilde{R}(p(x)) = a(x)\,R(d(x)). \tag{36}$$

When R is wide-sense stationary, Section 3 shows that the deformation gradient corresponding to $d(x)$ is almost equal to the velocity of the Texture Gradient Equation, at small scales (17). We now propose a similar approach for a general nondevelopable differentiable surface. Noting that (17) need only hold in the *small scale* limit ($\det S \to 0$), we introduce a local version of model (36). In order to transform $\tilde{R}$ into a 2D process, we project it locally onto the tangent plane to Σ at $p(u)$, denoted $T_{p(u)}(\Sigma)$, through the *exponential map* [15]. This map, $\exp_{p(u)}$, projects a neighborhood of 0 in $T_{p(u)}(\Sigma)$ to a neighborhood of $p(u)$ on Σ, as depicted in Fig. 9. It transforms radial lines stemming from 0 in $T_{p(u)}(\Sigma)$ into geodesics on Σ stemming from $p(u)$, while preserving lengths along these geodesics. We can define a 2D process $R_{p(u)}$ in the neighborhood of 0 on $T_{p(u)}(\Sigma)$ by

$$R_{p(u)}(v) = \tilde{R}(\exp_{p(u)}(v)).$$

Let $d_u(x)$ be the function such that $\exp_{p(u)}(d_u(x)) = p(x)$. By definition,

$$\begin{aligned} I(x) &= a(x)\,\tilde{R}(p(x)) = a(x)\,\tilde{R}(\exp_{p(u)}(d_u(x))) \\ &= a(x)\,R_{p(u)}(d_u(x)), \end{aligned} \tag{37}$$

which is a local version of model (36).

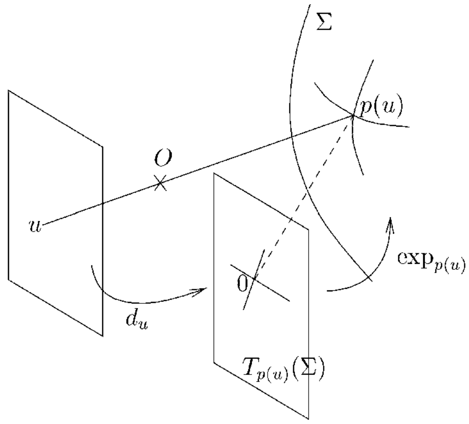


Fig. 9. The exponential map $\exp_{p(u)}$ maps a neighborhood of 0 on the tangent plane to a neighborhood of $p(u)$ on Σ. We define a local mapping d_u from the image plane to the tangent plane by $\exp_{p(u)}(d_u(x)) = p(x)$.

Let us now impose a homogeneity condition on $\tilde{R}$. First of all, it is natural to ask that $\mathbf{E}\{|\tilde{R}(p)|^2\}$ be independent of position $p \in \Sigma$. As a result, a local contrast normalization can be performed, leading to an image $I(x)$ such that

$$I(x) = R_{p(u)}(d_u(x)).$$

Let D_u be the deformation operator such that $D_u f(x) = f(d_u(x))$. Just like the function d_u, the operator D_u depends both on the local surface shape *and* on the perspective projection. Its adjoint is written $\overline{D}_u$ and

$$\langle \overline{D}_u f, g\rangle = \langle f, D_u g\rangle.$$

Because of distortions due to surface curvature, it does not make sense to require $R_{p(u)}$ to be wide-sense stationary. Moreover, as proved in [12] even in the developable case, when R is wide-sense stationary, the deformation gradient is only approximately equal to the texture gradient (17), with a resolution error of order $O(\det S)^{1/2}$. Introducing in the 2D Texture Gradient Equation (13) an additional error term of the same order is of no consequence. We can therefore tolerate the nonstationarity of $R_{p(u)}$ to induce an error of order $O(\det S)^{1/2}$. We impose that, for a position v close to 0 on the tangent plane, such that $|v| \leq (\det S)^{1/2}$,

$$\begin{aligned} &\|\mathbf{E}\{\langle R_{p(u)}, \overline{D}_u \psi_{v,S}\rangle\,\langle R_{p(u)}, \vec{\nabla}_x \overline{D}_u \psi_{v,S}\rangle^*\}\| \\ &\quad = O(\det S)^{1/2} \|\vec{\nabla}_u w(u,S)\|, \end{aligned} \tag{38}$$

where the gradients $\vec{\nabla}_x$ and $\vec{\nabla}_u$ are 2D vectors and $\|\cdot\|$ is the Euclidean norm. This condition imposes a nontrivial relationship between the surface geometry and the type of texture homogeneity. If Σ is developable and $R_{p(u)} = R$ is stationary, then the left-hand side of (38) vanishes, so the condition is trivially satisfied. However, the condition is much more general. For instance, it applies if $\tilde{R}$ is the restriction of a 3D stationary isotropic process to a sphere Σ, or if $\tilde{R}$ is a nonisotropic texture oriented along parallels or meridians as in Fig. 10, and considered at the equator [22].

Under condition (38), one can prove that the deformation gradient

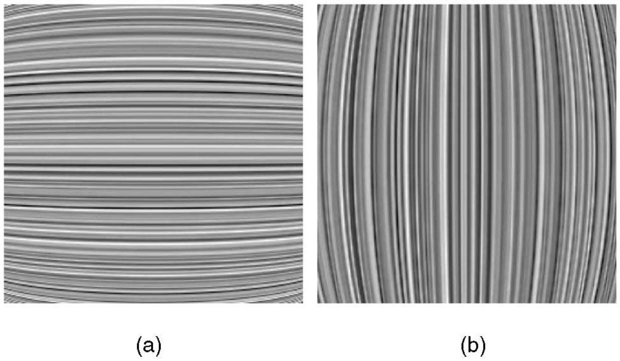


Fig. 10. A nonisotropic texture on a sphere, oriented along parallels (a) or meridians (b), obeys the weak stationarity condition (38) at the equator.

$$J_{d_u}(u)^{-1}\partial_{u_k} J_{d_u}(u)$$

is equal to the velocity of the Texture Gradient Equation, with an error term that tends to zero when $\det S \to 0$. Moreover, the geometrical relationships (32), (33) between deformation gradient and local surface shape, which derive from a differential analysis, are also valid for $J_{d_u}(u)^{-1}\partial_{u_k} J_{d_u}(u)$. The surface normal $\vec{n}$ can therefore be recovered from the image of the textured surface with the procedure described in Section 5.2.

The homogeneity assumption made in [5] is very restrictive [21], [23] because it supposes the local texture coordinates to follow either a frame field which is locally parallel [15], or a frame field whose differential rotation is known. The textures displayed in Fig. 10 are not oriented along a parallel frame field, even at the equator, but are nevertheless valid for Shape from Texture.

The condition (38) which we suggest is appropriate from a perceptual point of view. It is a condition between the surface and the texture, that allows the calculation of surface shape from the texture gradient, with an error term that tends to zero when the image resolution increases to infinity. Owing to the error term it tolerates, this condition applies to a broad class of textures and surfaces, for which the Shape from Texture problem can be solved visually. The remaining difficult issue is to specify precisely the class of shape-texture combinations satisfying (38).

6 Conclusion

The warplet transform is a natural tool for analyzing the image of a textured surface under perspective projection. Indeed, the warpogram of the image satisfies a transport equation, the *Texture Gradient Equation*. Under an appropriate homogeneity assumption on the original texture, the velocity of the Texture Gradient Equation is equal to the geometric deformation gradient, which measures relative metric changes between the surface and the image plane. We have introduced an estimator for the deformation gradient and demonstrated the Shape from Texture algorithm on photographs.

More work is necessary to fully understand what are the necessary and sufficient relationships between the texture and the surface shape, which guarantee that the deformation gradient is nearly equal to the velocity of the Texture Gradient Equation and, hence, that the texture gradient can be used for shape recovery.

Another area for further research concerns texture modeling. The Lambertian assumption is somewhat restrictive, and a much wider class of natural textures could be considered with a 3D modeling [24]. This could open promising directions for Shape from Texture.

Appendix A
Texture Gradient Equations

A.1 In One Dimension

We show that the velocity term $v(u, s)$ of the Texture Gradient Equation (8) tends to the deformation gradient $d''(u)/d'(u)$ when $s \to 0$. For this purpose, we first verify that when the scale s is small enough, the scalograms of I and of R are related by a simple migration property in the position-scale parameter space.

Since $I(x) = R(d(x))$, we have

$$\langle I, \psi_{u,s}\rangle = \int R(d(x))\, \frac{1}{s}\, \psi\Big(\frac{x-u}{s}\Big)\, dx,$$

and after a change of variable $x' = d(x)$, we get

$$\langle I, \psi_{u,s}\rangle = \int R(x')\, \frac{1}{s}\, \psi\Big(\frac{d^{-1}(x')-u}{s}\Big) \frac{dx'}{d'(d^{-1}(x'))}.$$

Because ψ is supported in $[-1, 1]$, $\psi\Big(\frac{d^{-1}(x')-u}{s}\Big)$ is nonzero only if x' belongs to $[d(u-s),\, d(u+s)]$. If $s \ll d'(u)/d''(u)$, a Taylor expansion for d around position u implies that $|x' - d(u)| \leq 2d'(u)\, s$. Hence, when $s \to 0$, $d'(d^{-1}(x')) \approx d'(u)$, and if ψ is in $\mathbf{C}^1$, a Taylor expansion for ψ shows that

$$\psi\Big(\frac{d^{-1}(x')-u}{s}\Big) \approx \psi\Big(\frac{x'-d(u)}{d'(u)s}\Big).$$

As a consequence,

$$\langle I, \psi_{u,s}\rangle \approx \int R(x)\, \frac{1}{d'(u)s}\, \psi\Big(\frac{x-d(u)}{d'(u)s}\Big)\, dx. \tag{39}$$

But according to (6),

$$\frac{1}{d'(u)s}\, \psi\Big(\frac{x-d(u)}{d'(u)s}\Big) = \psi_{d(u),d'(u)s}(x),$$

so $\langle I\, ,\, \psi_{u,s}\rangle \approx \langle R\, ,\, \psi_{d(u),d'(u)s}\rangle$, which implies the scalogram migration property (9):

$$\mathbf{E}\{|\langle I, \psi_{u,s}\rangle|^2\} \approx \mathbf{E}\{|\langle R, \psi_{d(u),d'(u)s}\rangle|^2\}.$$

Let s_0 be a fixed constant, with $s_0 \ll d'(u)^2/d''(u)$, and let $s(u) = s_0/d'(u)$. For $s = s(u)$, (9) gives

$$w(u, s(u)) = \mathbf{E}\{|\langle I, \psi_{u,s(u)}\rangle|^2\} \approx \mathbf{E}\{|\langle R, \psi_{d(u),s_0}\rangle|^2\}.$$

Since R is stationary, $\mathbf{E}\{|\langle R, \psi_{d(u),s_0}\rangle|^2\}$ does not depend on u; therefore,

$$\frac{d}{du} w(u, s(u)) \approx 0.$$

One can expand the total derivative $\frac{d}{du}$ as a linear combination of partial derivatives:

$$\frac{d}{du}w(u,s(u)) = \partial_u w(u,s(u)) + s'(u)\partial_s w(u,s(u)).$$

Noticing that $s'(u) = -\frac{d''(u)}{d'(u)}s(u)$, for s sufficiently small, we get

$$\partial_u w(u,s) - \frac{d''(u)}{d'(u)}\,\partial_{\log s} w(u,s) \approx 0, \qquad (40)$$

which, when combined with (8), verifies (10): $v(u,s) \approx \frac{d''(u)}{d'(u)}$.

A.2 In Two Dimensions

Let us first verify the warpogram migration property (15). If $\det S$ is small enough

$$\begin{aligned}\langle I, \psi_{u,S}\rangle &= \iint R(d(x))\,\frac{1}{\det S}\psi\big(S^{-1}(x-u)\big)\,dx\\ &= \iint R(x')\frac{1}{\det S}\psi\big(S^{-1}(d^{-1}(x')-u)\big)\frac{dx'}{\det J_d(d^{-1}(x'))}\\ &\approx \iint R(x')\frac{1}{\det S}\psi\big(S^{-1}J_d^{-1}(u)(x'-d(u))\big)\frac{dx'}{\det J_d(u)},\end{aligned}$$

where $J_d(x)$ is the Jacobian matrix defined in (4). Hence,

$$\langle I, \psi_{u,S}\rangle \approx \langle R, \psi_{d(u),J_d(u)S}\rangle,$$

so $\mathbf{E}\{|\langle I,\psi_{u,S}\rangle|^2\} \approx \mathbf{E}\{|\langle R,\psi_{d(u),J_d(u)S}\rangle|^2\}$ which verifies the migration property (15).

Let $S(u) = J_d(u)^{-1}S_0$. If $\det S_0$ is small enough, setting $S = S(u)$ in the warpogram migration property yields

$$w(u,S(u)) = \mathbf{E}\Big\{|\langle I,\psi_{u,S(u)}\rangle|^2\Big\} \approx \mathbf{E}\Big\{|\langle R,\psi_{d(u),S_0}\rangle|^2\Big\}.$$

Since R is stationary, the right-hand side of the above relation is independent of u. Therefore, for $k = 1,2$,

$$\frac{d}{du_k}w(u,S(u)) \approx 0. \qquad (41)$$

Let

$$C^k(u) = (c^k_{ij}(u))_{\{1\le i,j\le 2\}} = \partial_{u_k}\Big(J_d(u)^{-1}\Big)S_0,$$

and let $S(u) = J_d(u)^{-1}S_0$. Because of (41),

$$\frac{d}{du_k}w(u,S(u)) = \partial_{u_k}w(u,S(u)) + \sum_{i,j} c^k_{ij}(u)\,\partial_{s_{ij}}w(u,S(u)) \approx 0. \qquad (42)$$

But,

$$\partial_{u_k}\Big(J_d(u)^{-1}\Big) = -J_d(u)^{-1}\partial_{u_k}(J_d(u))J_d(u)^{-1},$$

therefore,

$$C^k(u) = -J_d(u)^{-1}\partial_{u_k}(J_d(u))S(u).$$

By inserting the definition (14) of $a_{ij}(u,S)$ and the definition (16) of the texture gradient coefficients $g^k_{ij}(u)$, a direct calculation shows that

$$\sum_{i,j} c^k_{ij}(u)\,\partial_{s_{ij}}w(u,S(u)) = -\sum_{i,j} g^k_{ij}(u)\,a_{ij}(u,S(u)). \qquad (43)$$

Inserting (43) in (42) shows that, for for $S(u) = S$,

$$\partial_{u_k}w(u,S) - \sum_{i,j=1}^{2} g^k_{ij}(u)\,a_{ij}(u,S) \approx 0. \qquad (44)$$

This proves that the deformation gradient terms $g^k_{ij}(u)$ are admissible as velocity terms $v^k_{ij}(u,S)$ of the Texture Gradient Equation.

Appendix B
Wavelet and Warplet Expressions

B.1 Wavelet: Modulated Spline

The wavelet coefficients are calculated with the FFT [17]: A wavelet transform can be obtained as a convolution product

$$\langle I,\psi_{u,s}\rangle = \int I(x)s^{-1}\psi(s^{-1}(x-u))\,dx = I * \tilde{\psi}_s(u), \qquad (45)$$

with $\tilde{\psi}_s(x) = s^{-1}\psi(-s^{-1}x)$. The Fourier transform of $\tilde{\psi}_s(x)$ is $\widehat{\tilde{\psi}_s}(\omega) = \widehat{\psi}^*(s\omega)$. We choose ψ to be a modulated box-spline whose Fourier transform is

$$\widehat{\psi}(\omega) = \left(\frac{\sin(\omega/2-\pi)}{\omega/2-\pi}\right)^5 \exp(-i(\omega/2-\pi)).$$

For a discrete signal of size N, the wavelet $\tilde{\psi}_s$ and the variable u are discretized over the sampling grid and (45) is computed with the FFT, requiring $O(N\,\log N)$ operations.

The wavelet coefficients $\langle I,\partial_a\psi_{u,s}\rangle = I * \partial_a\tilde{\psi}_s(u)$ are also calculated with the FFT using Fourier expressions derived from (19), (20),

$$\widehat{\partial_u\tilde{\psi}_s}(\omega) = -i\omega\,\widehat{\psi}(s\omega)$$
$$\widehat{\partial_{\log s}\tilde{\psi}_s}(\omega) = s\omega\,\widehat{\psi}'(s\omega).$$

B.2 Warplet: Modulated Gaussian

Like the wavelet transform, the warplet transform can be written as the result of a 2D convolution product:

$$\langle I,\psi_{u,S}\rangle = \int I(x)\det S^{-1}\psi(S^{-1}(x-u))\,dx = I * \tilde{\psi}_S(u), \qquad (46)$$

with $\tilde{\psi}_S(x) = \det S^{-1}\psi(-S^{-1}x)$. Note that the Fourier transform of $\tilde{\psi}_S$ is $\widehat{\tilde{\psi}_S}(\omega) = \widehat{\psi}^*(S^T\omega)$.

We choose ψ to be a Gabor function whose Fourier transform is

$$\widehat{\psi}(\omega_1,\omega_2) = \exp\left(-\frac{(\omega_1-2\pi)^2+\omega_2^2}{4}\right).$$

The warplet $\tilde{\psi}_S$ and the variable u are discretized over the image sampling grid. Computing (46) with the FFT requires $O(N^2\,\log N)$ operations. Similarly to (46), $\langle I,\partial_a\psi_{u,S}\rangle = I * \partial_a\tilde{\psi}_S(u)$ can be computed with the FFT, using the Fourier transform expressions

$$\begin{aligned}
\widehat{\partial_{s_{11}}\tilde{\psi}_S}(\omega) &= -\frac{\omega_1}{2}\,(s_{11}\omega_1 + s_{21}\omega_2 - 2\pi)\,\widehat{\psi}(S^T\omega),\\
\widehat{\partial_{s_{12}}\tilde{\psi}_S}(\omega) &= -\frac{\omega_1}{2}\,(s_{12}\omega_1 + s_{22}\omega_2)\,\widehat{\psi}(S^T\omega),\\
\widehat{\partial_{s_{21}}\tilde{\psi}_S}(\omega) &= -\frac{\omega_2}{2}\,(s_{11}\omega_1 + s_{21}\omega_2 - 2\pi)\,\widehat{\psi}(S^T\omega),\\
\widehat{\partial_{s_{22}}\tilde{\psi}_S}(\omega) &= -\frac{\omega_2}{2}\,(s_{12}\omega_1 + s_{22}\omega_2)\,\widehat{\psi}(S^T\omega),\\
\widehat{\partial_{u_1}\tilde{\psi}_S}(\omega) &= -i\omega_1\,\widehat{\psi}(S^T\omega),\\
\widehat{\partial_{u_2}\tilde{\psi}_S}(\omega) &= -i\omega_2\,\widehat{\psi}(S^T\omega).
\end{aligned}$$

Appendix C

Tilt and Slant Estimation

For the sake of completeness, we reproduce results of [21] on least-squares tilt and slant estimation. Let us denote the elements of matrices $M_t(u)$ and $M_b(u)$ defined in (34) and (35) as

$$M_t(u) = \begin{pmatrix} \alpha & \beta \\ 0 & \gamma \end{pmatrix}, M_b(u) = \begin{pmatrix} \beta & \delta \\ \gamma & 0 \end{pmatrix}. \quad (47)$$

We concatenate the elements of the two deformation gradient matrices (16) into a single vector:

$$\begin{aligned} g = &(g^1_{11}(u), g^1_{12}(u), g^1_{21}(u), g^1_{22}(u),\\ &g^2_{11}(u), g^2_{12}(u), g^2_{21}(u), g^2_{22}(u))^T. \end{aligned}$$

Equations (32) and (33) show that there is a linear relationship between g and $(\alpha, \beta, \gamma, \delta)$:

$$g = H(\theta)\,(\alpha, \beta, \gamma, \delta)^T,$$

where $H(\theta)$ is the matrix

$$\begin{bmatrix}
c^3 & -2c^2s & 2cs^2 & cs^2\\
c^2s & -2cs^2 & -2c^2s & s^3\\
c^2s & c(c^2-s^2) & -s(c^2-s^2) & -c^2s\\
cs^2 & s(c^2-s^2) & c(c^2-s^2) & -cs^2\\
c^2s & c(c^2-s^2) & -s(c^2-s^2) & -c^2s\\
cs^2 & s(c^2-s^2) & c(c^2-s^2) & -cs^2\\
cs^2 & 2c^2s & -2cs^2 & c^3\\
s^3 & 2cs^2 & 2c^2s & c^2s
\end{bmatrix}$$

with $c = \cos\theta$ and $s = \sin\theta$. Given the estimation of the deformation gradient g obtained in Section 5.1, we seek the value of $(\alpha, \beta, \gamma, \delta, \theta)$ which minimizes

$$\|g - H(\theta)\,(\alpha, \beta, \gamma, \delta)^T\|_2^2.$$

For a specific θ, the $(\alpha, \beta, \gamma, \delta)_\theta$ which minimizes $\|g - H(\theta)(\alpha, \beta, \gamma, \delta)^T\|_2^2$ is given in closed form by

$$(\alpha, \beta, \gamma, \delta)_\theta = \left(\left(H(\theta)^T H(\theta)\right)^{-1} H(\theta)^T g\right)^T. \quad (48)$$

The only step left is to find the θ which minimizes

$$\left\| g - H(\theta)\left(H(\theta)^T H(\theta)\right)^{-1} H(\theta)^T g \right\|_2^2,$$

and this can be achieved by a one-dimensional minimization method. Once the tilt direction θ has been estimated, $(\alpha, \beta, \gamma, \delta)_\theta$ are calculated with (48). The slant direction is then given by $\sigma = \arctan\gamma$. Although they are not actually used in our surface shape estimation algorithm, the curvature parameters κ_t, κ_b, and τ can also be computed from $(\alpha, \beta, \gamma, \delta, \theta)$, up to a scaling factor $\|\overrightarrow{Op(u)}\|$.

Acknowledgments

The authors would like to thank the referees for theirhelpful comments. Funding for this work was provided in part by AFOSR-MURI grant 25-74100-F1837.

References

[1] J. Gibson, *The Perception of the Visual World.* Boston, Mass.: Houghton Mifflin, 1950.

[2] J. Gårding, "Shape from Texture for Smooth Curved Surfaces in Perspective Projection," *J. Math. Imaging Vision,* vol. 2, pp. 327–350, 1992.

[3] W. Hwang, C.-S. Lu, and P.-C. Chung, "Shape from Texture: Estimation of Planar Surface Orientation through the Ridge Surfaces of Continuous Wavelet Transform," *IEEE Trans. Image Processing,* vol. 7, pp. 773–780, 1998.

[4] K. Kanatani and T. Chou, "Shape from Texture: General Principle," *Artificial Intelligence,* vol. 38, pp. 1–48, 1989.

[5] J. Malik and R. Rosenholtz, "Computing Local Surface Orientation and Shape from Texture for Curved Surfaces," *Int'l J. Computer Vision,* vol. 23, no. 2, pp. 149–168, 1997.

[6] H. Permuter and J. Francos, "Estimating the Orientation of Planar Surfaces: Algorithms and Bounds," *IEEE Trans. Information Theory,* vol. 46, no. 5, 2000.

[7] B. Super and A. Bovik, "Shape from Texture Using Local Spectral Moments," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 4, pp. 333–343, Apr. 1995.

[8] R. Bajcsy and L. Lieberman, "Texture Gradient as a Depth Cue," *Computer Graphics and Image Processing,* vol. 5, pp. 52–67, 1976.

[9] M. Clerc and S. Mallat, "Shape from Texture through Deformations," *Proc. Seventh Int'l Conf. Computer Vision,* 1999.

[10] B. Horn and B. Schunck, "Determining Optical Flow," *Artificial Intelligence,* vol. 17, nos. 1-3, pp. 185–203, 1981.

[11] R. Rosenholtz and J. Malik, "Surface Orientation from Texture: Isotropy or Homogeneity (or Both)?," *Vision Research,* vol. 37, no. 16, pp. 2283–2293, 1997.

[12] M. Clerc and S. Mallat, "Estimating Deformations of Stationary Processes," Technical Report 2000-192, CERMICS, L'Ecole Nationale des Ponts et Chaussões, 2000.

[13] B. Horn, *Robot Vision.* McGraw-Hill, 1986.

[14] M. Clerc and S. Mallat, "Shape from Texture and Shading with Wavelets," *Dynamical Systems, Control, Coding, Computer Vision, Progress in Systems and Control Theory,* vol. 25, pp. 393–417, 1999.

[15] M.D. Carmo, *Differential Geometry of Curves and Surfaces.* Prentice-Hall, 1976.

[16] P. Lions, E. Rouy, and A. Tourin, "Shape-from-Shading, Viscosity Solutions and Edges," *Numerische Mathematik,* vol. 64, pp. 323–353, 1993.

[17] S. Mallat, *A Wavelet Tour of Signal Processing.* Academic Press, 1999.

[18] A. Yaglom, *Correlation Theory of Stationary and Related Random Functions,* vol. 1. Springer-Verlag, 1987.

[19] T. Randen and J. Husoy, "Filtering for Texture Classification: A Comparative Study," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 4, pp. 291–301, Apr. 1999.

[20] G.H. Golub and C.F. Van Loan, *Matrix Computations.* Johns Hopkins Univ. Press, 1989.

[21] J. Gårding, "Surface Orientation and Curvature from Differential Texture Distorsion," *Proc. Fifth Int'l Conf. Computer Vision,* 1995.

[22] M. Clerc, "Estimation de Processus Localement Dilatés et Application au Gradient de Texture," PhD thesis, Ecole Polytechnique, Paris, 1999.

[23] D. Forsyth, "Shape from Texture and Integrability," *Proc. Eighth Int'l Conf. Computer Vision,* 2001.

[24] T. Leung and J. Malik, "Recognizing Surfaces Using Three-Dimensional Textons," *Proc. Seventh Int'l Conf. Computer Vision,* 1999.

Maureen Clerc graduated from Ecole Polytechnique, Paris, in 1993 and from Ecole Nationale des Ponts et Chaussées (ENPC) in 1996. She received the PhD degree in applied mathematics from Ecole Polytechnique in 1999. She was a visiting scholar in the Statistics Department at Stanford University in 2000. Since 2001, she has held a research position at CERMICS, joint laboratory between INRIA and ENPC. Her research interests include wavelet-based image analysis, shape recovery, texture processing and functional brain imaging.

Stéphane Mallat graduated from Ecole Polytechnique in 1984 and from Ecole Nationale Supérieure des Télécommunications, Paris, in 1985. He received the PhD degree in electrical engineering from the University of Pennsylvania, Philadelphia, in 1988. In 1988, he joined the Computer Science Department of the Courant Institute of Mathematical Sciences at New York University, where he became an associate professor in 1993, and is now a research professor. In the fall 1994, he was a visiting professor in the Electrical Engineering Department at Massachusettes Institute of Technology and in the spring 1994 in the Applied Mathematics Department at the University of Tel Aviv. Since 1995, he has been a professor in the Applied Mathematics Department at Ecole Polytechnique, Paris, France, and is now chairman of the department. His research interest include computer vision, signal processing, and diverse applications of wavelet transforms. He is the author of a book *A Wavelet Tour of Signal Processing*, Academic Press, 1999. Dr. Mallat received the 1990 IEEE Signal Processing Society's paper award, the 1993 Alfred Sloan fellowship in mathematics, the 1997 Outstanding Achievement Award from the SPIE Optical Engineering Society, and the 1997 Blaise Pascal Prize in applied mathematics, from the French Academy of Sciences. He is a member of the IEEE.