# The Texture Gradient Equation for recovering Shape from Texture

Maureen Clerc
CERMICS
Ecole Nationale des Ponts et Chaussées, France

Stéphane Mallat
CMAP, Ecole Polytechnique, France
and Courant Institute, New York University

November 22, 2000

**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 the solution 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, wavelets, warplets.

## 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 fifty years ago by Gibson [10], 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, 8, 14, 15, 18]. The *Shape from Texture* problem is generally broken down into two independent steps. The first step is to measure the texture distorsion in the image, and the second is to recover the surface coordinates from this texture distorsion.

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 [15, 18, 24]. Measuring the distorsion 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 [1, 18], and wavelets have also recently been introduced for *Shape from Texture* [14].

Traditionally, one measures the texture distorsion 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 in measuring the relative distorsion of the texture within the observed image, without reference to the original texture. In [18], the relative texture distorsion 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 $2 \times 2$ warping matrix. The observed textured image is modeled as the realization of a stochastic process. The texture distorsion 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-scale parameter space. This fundamental transport equation obeyed by the warpogram is called *Texture Gradient Equation*. It can be seen as the analog of the Optical Flow Equation for motion estimation [13]. Whereas the velocity term in the Optical Flow Equation is related to the projection of the 3D velocity in the image, here, the velocity measures relative texture distorsion in the image. The texture distorsion is thus calculated by estimating the different terms of this equation.

The next step is to recover the 3D surface coordinates from the texture distorsion. For this, a key assumption on the underlying texture has to be made: that it displays some form of spatial homogeneity on the surface. 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 [22]. Here, we address *Shape from Texture* without supposing isotropy for the original texture. As natural though it may appear from a perceptual point of view, texture homogeneity on a general surface is very difficult to state mathematically. We can distinguish two independent subproblems nested in the recovery of 3D surface coordinates from the texture gradient. One of the subproblems is purely geometrical, and concerns the change of metric between the 3D surface and the image plane, due to the projection (either orthographic or perspective) and to the surface curvature. 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 [8] and further analyzed by Malik and Rosenholtz [18], 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 other subproblem concerns texture modelization, and imposes a homogeneity condition on the texture, under which the deformation gradient is the solution of the *Texture Gradient Equation*, and thus can be calculated.

To recapitulate, we decompose *Shape from Texture* into three separate problems:

- Impose a homogeneity condition on the texture, under which the deformation gradient is the solution of the *Texture Gradient Equation*.
- Estimate the deformation gradient from the *Texture Gradient Equation*.
- Measure the 3D surface coordinates from the deformation gradient.

The paper is organized as follows. In Section 1, 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 2.1, 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 2.2 establishes the *Texture Gradient Equation* in 2D. Wavelets are now replaced by warplets, which are especially designed to migrate in the position-scale parameter space under a 2D affine transform. In Section 3, 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 4 presents our Shape from Texture algorithm, with examples on photographs. Lastly, we propose a new homogeneity condition, based on the *Texture Gradient Equation*, which generalizes the homogeneity condition of Section 1 to general surfaces. This paper is oriented towards modelization and algorithms: although we state some mathematical results, we refer to [7] for their detailed proofs.

# 1 Shape from Texture Model

We assume the surface to have a Lambertian reflectance distribution. This supposes the texture to be "painted" on the surface, and to have neither rugosity, nor self-occlusions. With our 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 (Figure 1). For instance, if the light comes from a point source in direction $\vec{s}$, and if $\vec{n}$ is the surface normal, then $a(x) = \vec{n}(p(x)) \cdot \vec{s}(p(x))$ [12, 6].

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 decompose the *Shape from Texture* problem in three steps: making a homogeneity assumption on the texture under which the deformation gradient is the solution of the Texture Gradient Equation; estimating the terms of the Texture Gradient Equation, and obtaining the 3D surface coordinates from the deformation gradient. We now constrain model (1) by imposing a homogeneity condition on the original texture $\tilde{R}$.

A developable surface $\Sigma$ (i.e. with zero Gaussian curvature) can be unfolded isometrically into a portion of a plane, thus defining a mapping from each position $p(x) \in \Sigma$ onto $d(x) \in \mathbb{R}^2$ [4]. 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

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

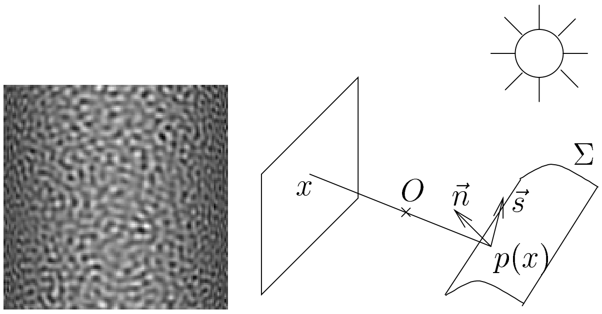



Figure 1: Perspective image of a textured cylinder (left). Image formation (right): each position $x$ in the image corresponds to a point $p(x)$ at the intersection between the surface $\Sigma$ and the light ray connecting $x$ and the optical center $O$. Vectors $\vec{n}$ and $\vec{s}$ respectively represent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

We define the original texture $\tilde{R}$ to be homogeneous if $R$ is a wide-sense stationary process:

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

In this case,

$$\mathsf{E}\left\{|I(x)|^2\right\} = |a(x)|^2 \mathsf{E}\left\{|R(d(x))|^2\right\} = |a(x)|^2\, C(0)\ .$$

The shading term $a(x)$ can thus be estimated up to a multiplicative constant from the second moment of the image $\mathsf{E}\left\{|I(x)|^2\right\}$. *Shape from Shading* studies shape recovery from the shading term only [17, 20]. Here, we concentrate on the texture distorsion, and hence compensate for illumination changes. We estimate the second moment of the image, and then calculate $\mathsf{E}\left\{|I(x)|^2\right\}^{-1/2} I(x)$. The image resulting from this local contrast renormalization is still denoted $I(x)$ for convenience. The model therefore simplifies to

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

We assume that the surface $\Sigma$ 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 $\mathrm{I\!R}^2$ 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 $\Sigma$ is developable, and it can be isometrically unfolded into a portion of a plane, $J_d(x)$ represents the change of metric between the surface $\Sigma$ 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 $k = 1, 2$:

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

We want to solve the following inverse problem: estimate the deformation gradient (5), given one realization of $I(x) = R(d(x))$.

# 2 Texture Gradient Equation

## 2.1 In 1D: Scalogram Migration

For the sake of simplicity, let us start with a 1D *Shape from Texture* problem, in which the shape $\Sigma$ to be recovered is a curve. Let $R$ denote the "reflectance" of $\Sigma$, parameterized by arc-length $\ell$: $R(\ell)$ is assumed stationary, and is depicted, in Figure 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 $\Sigma$, 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 $\psi$ 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

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

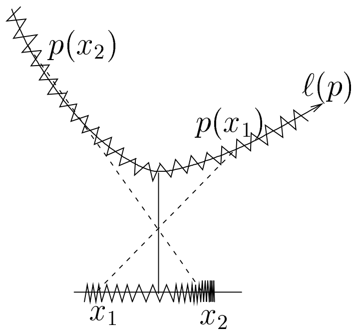



Figure 2: A point $x$ in the image backprojects to a point $p(x)$ on $\Sigma$, whose arc-length is $d(x) = \ell(p(x))$. The zig-zag line represents a stationary texture covering $\Sigma$. When projected onto the image, it gives rise to a non-stationary process $I(x) = R(d(x))$.

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 $s$ [19], and we call *scalogram* of $I$ the variance of this wavelet coefficient:

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

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

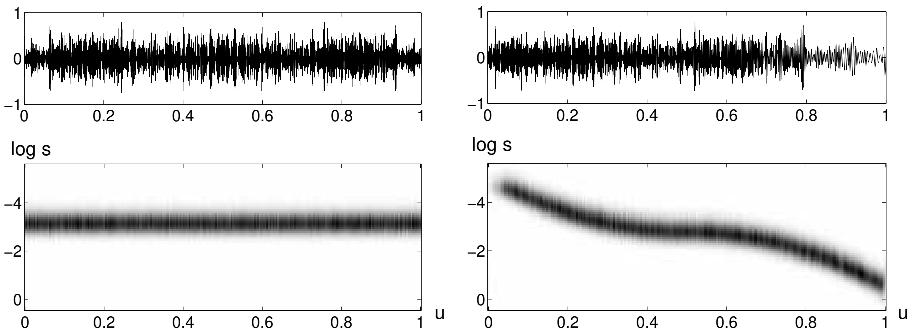
$$\frac{d}{du} \mathsf{E}\{|\langle R, \psi_{u,s} \rangle|^2\} = 0 \ . \tag{7}$$

In Figure 3(a), 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 Figure 3(b).

We introduce the *Texture Gradient Equation*, which relates the partial derivatives of $w(u, s)$ with respect to $u$ and to $\log s$:

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

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 [13]. The conservation equation




(a) (b)

Figure 3: (a) Top: a realization of stationary process $R(x)$. Bottom: scalogram $\mathsf{E}\{|\langle R, \psi_{u,s}\rangle|^2\}$. 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 $w(u,s) = \mathsf{E}\{|\langle I, \psi_{u,s}\rangle|^2\}$.

(7), expressing the stationarity of the original process $R$, is analogous to brightness constancy in Optical Flow.

We are now going to show that the velocity term $v(u,s)$ tends to the deformation gradient $d''(u)/d'(u)$ when $s \to 0$. For this, we prove 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.

Because $\psi_{u,s}$ is supported in $[u-s, u+s]$, a wavelet coefficient $\langle I, \psi_{u,s}\rangle$ only depends on image intensities in a neighborhood of $u$:

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

and a change of variable $x' = d(x)$ yields

$$\int R(x')\, \frac{1}{s}\, \psi\left(\frac{d^{-1}(x')-u}{s}\right) \frac{dx'}{d'(d^{-1}(x'))}\ .$$

If $s \ll d'(u)/d''(u)$, then $\psi\left(\frac{d^{-1}(x')-u}{s}\right)$ is non-zero only if $x'$ belongs to a neighborhood of $d(u)$ proportional to $d'(u)s$. Hence $d'(d^{-1}(x')) \approx d'(u)$, and if $\psi$ is in $\mathbf{C}^1$,

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

As a consequence,

$$\langle I, \psi_{u,s}\rangle \approx \int R(x)\frac{1}{d'(u)s}\psi\left(\frac{x-d(u)}{d'(u)s}\right) dx . \tag{9}$$

But according to (6),

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

so on the right-hand side of (9), we recognize $\langle R, \psi_{d(u),d'(u)s}\rangle$ which is a wavelet coefficient of $R$ at position $d(u)$ and scale $d'(u)s$. Let $s_0$ be a fixed constant, much smaller than $d''(u)$, and let $s(u) = s_0/d'(u)$. Then (9) implies that

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

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

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

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

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

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

$$\partial_u w(u,s) - \frac{d''(u)}{d'(u)}\partial_{\log s} w(u,s) \approx 0 . \tag{10}$$

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.

Equation (10) was derived in a loose fashion, but a careful analysis of the higher-order terms gives the following result [7]: if the covariance of $R$, $C(\tau)$, defined in (2), satisfies

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

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

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

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 [25]. This proves that at small scales $s \to 0$, the deformation gradient is the solution of the *Texture Gradient Equation*.

A scalogram $w(u,s)$ is displayed on the bottom of Figure 3(b): the positions of the scalogram maxima 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. Section 3 focuses on this estimation problem.

### 2.2 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 positive scalar, whereas in 2D, it is a $2 \times 2$ 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 compactly supported function in $[-1,1] \times [-1,1]$, with zero average. A warplet $\psi_{u,S}$ is indexed by its position $u = (u_1, u_2)$, and by a $2 \times 2$ invertible matrix

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

which deforms the support of $\psi$:

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

For instance, modulated Gaussians, called Gabor functions,

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

have widely been used for texture discrimination [3, 21].

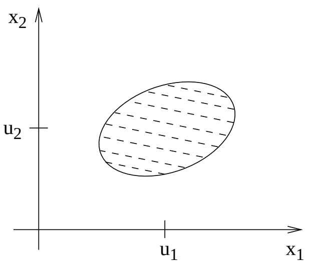



Figure 4: Approximate spatial localization of a warplet $\psi_{u,S}$ constructed with a Gabor function $\psi$. The four parameters of $S$ control the shape and orientation of the ellipse, as well as the frequency of its oscillations (represented by the stripes).

A *warplet coefficient* of $I(x)$ at position $u$ and for a 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 $I$:

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

As in 1D, we are going to prove that the deformation gradient is the solution of a Texture Gradient Equation in 2D, for small scales, when $\det S \to 0$. For this, we observe that if $I(x) = R(d(x))$, then the warpograms of $I$ and of $R$ are related by a migration in the six-dimensional $(u_1, u_2, s_{11}, s_{12}, s_{21}, s_{22})$ parameter space. A warplet coefficient $\langle I, \psi_{u,S} \rangle$ only depends on the values of $I$ in a neighborhood of $u$ whose size is controlled by $S$. If $\det S$ is small enough,

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

where $J_d(x)$ is the Jacobian matrix defined in (4). Warplet coefficients of $I$ and of $R$ are thus related by a migration in position-scale:

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

Let $S(u) = J_d(u)^{-1} S_0$. If $\det S_0$ is small enough, the warpograms of $I$ and $R$ are related by

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

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

$$\frac{d}{du_k} w(u, S(u)) \approx 0 \ . \tag{13}$$

We show in Appendix A that, after expanding the total derivative $\frac{d}{du_k}$, the above equation can be rewritten, for $k = 1, 2$,

$$\partial_{u_k} w(u, S) - \sum_{i,j=1}^{2} g_{ij}^k(u) \, a_{ij}(u, S) \approx 0 \ , \tag{14}$$

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

$$\begin{pmatrix} a_{11}(u,S) & a_{12}(u,S) \\ a_{21}(u,S) & a_{22}(u,S) \end{pmatrix} = \begin{pmatrix} \frac{\partial w(u,S)}{\partial s_{11}} & \frac{\partial w(u,S)}{\partial s_{12}} \\ \frac{\partial w(u,S)}{\partial s_{21}} & \frac{\partial w(u,S)}{\partial s_{22}} \end{pmatrix} \times \begin{pmatrix} s_{11} & s_{21} \\ s_{12} & s_{22} \end{pmatrix} \ , \tag{15}$$

and the $g_{ij}^k$ represent the deformation gradient (5):

$$\begin{pmatrix} g_{11}^k(u) & g_{12}^k(u) \\ g_{21}^k(u) & g_{22}^k(u) \end{pmatrix} = J_d(u)^{-1} \partial_{x_k} J_d(u) \ . \tag{16}$$

In [7], we prove that the resolution error in equation (14) is of the order of $O(\det S)^{1/2}$. This shows that, when $\det S \to 0$, the deformation gradient is the solution of a 2D *Texture Gradient Equation*, which is a vector-valued equation with two components, for $k = 1, 2$:

$$\partial_{u_k} w(u, S) - \sum_{i,j=1}^{2} v_{ij}^k(u, S)\, a_{ij}(u, S) = 0 \ . \tag{17}$$

In Section 4, we will show how to use (14) to estimate the deformation gradient from warplet coefficients of the image.

# 3 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 we know from (12) to be the solution 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 $w(u, s) = \mathsf{E}\left\{|\langle I, \psi_{u,s}\rangle|^2\right\}$, for a generic variable $a$ representing $u$ or $\log s$, one can see that

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

with

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

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

Let $\psi$ 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 $\psi$ and $\partial_a \psi$. Appendix B.1 details the implementation of the wavelet transform.

In view of (18), we 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 $k_\Delta(x) = \Delta^{-1}k(\Delta^{-1}x)$ supported in $[-\Delta, \Delta]$:

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

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

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

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

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

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

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

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 $\Delta$ so that the bias term is of the same order as the variance of the estimator. We have proved in [7] that for $s = \lambda N^{-1}$ and for $\Delta = \mu N^{-1/5}$, if $R$ is a Gaussian process, with a covariance which satisfies (11) for a certain $h > 0$, and if the number of vanishing moments of $\psi$ is larger than $(2h+1)/4$, then

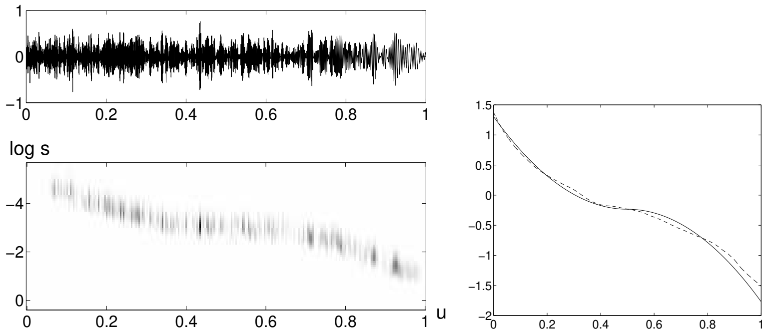
$$\mathrm{Prob}\left\{\left|\widehat{\tfrac{d''(u)}{d'(u)}} - \tfrac{d''(u)}{d'(u)}\right| \leq 2\,(\log N)\, N^{-1/5}\right\} \xrightarrow[N\to\infty]{} 1 \ .$$

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.

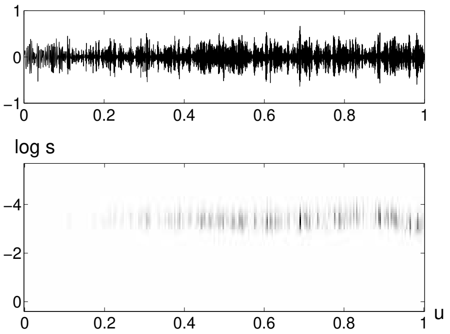
This consistency result guarantees the convergence of the estimator, but in practice, for a fixed resolution, averaging the smoothed estimators across several scales improves the result. We therefore propose a modified estimator for $\frac{d''(u)}{d'(u)}$:

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

In the example of Figure 5, the signal $I(x)$, displayed in (a), is sam-




(a) (b)




(c)

Figure 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 $|\langle \hat{R}, \psi_{u,s}\rangle|^2$ .

pled over [0, 1] at a resolution $N = 4096$, and we choose 6 scales in a

range $\log s \in [-4.5, -3]$ for which the signal has energy (see Figure 5(a)). 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 an FFT procedure explained in Appendix B.1. The smoothing kernel is $k(x) = 1 - |x|$ for $x$ inside $[-1, 1]$, and zero outside this interval. The estimator $\widehat{\frac{d''(u)}{d'(u)}}$ is computed with (26), for $\Delta = 0.77 * N^{-1/5}$. The overall algorithm requires $O(N \log N)$ operations. It is important to note that only the wavelet coefficients corresponding to 6 different scales need to be computed. The whole wavelet transform plane is displayed in Figures 5(a) and (c) for an explanatory purpose only. Figure 5(b) 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 $\hat{R}(x) = I(\widehat{d}^{-1}(x))$. Figure 5(c) displays $\hat{R}$: as expected, its wavelet transform remains nearly constant when $u$ varies, modulo statistical fluctuations. MATLAB scripts reproducing Figure 5 are available at
`http://cermics.enpc.fr/~maureen/ShapeFromTexture.html`.

# 4 Application to Shape from Texture

We now turn to the estimation of shape from texture. Section 4.1 details the deformation gradient estimation from the warplet coefficients of the image, when the deformation gradient is the solution of the 2D Texture Gradient Equation. Section 4.2 presents shape recovery from the deformation gradient, and lastly, Section 4.3 gives a condition on the texture, for general (not necessarily developable) surfaces, so that the deformation gradient is indeed the solution of the 2D Texture Gradient Equation at small scales.

## 4.1 Deformation Gradient estimation

As explained in Section 1, we preprocess the image to remove the shading term. We suppose $I$ only to have positive values, and we convolve $I^2$ with a 2D Gaussian $G_\sigma$. The variance $\sigma$ of the Gaussian must be adjusted according to the scale of shading variations. The original image $I(x)$ is then divided by $(I^2 * G_\sigma)(x)^{1/2}$. The images in Figures 6(a) and 7(a) are the result of this preprocessing step.

When the deformation gradient is the solution of the Texture Gradient Equation, and when $\det S \to 0$, (14) can be rewritten as

$$[a_{11}(u,S), a_{12}(u,S), a_{21}(u,S), a_{22}(u,S)] \begin{bmatrix} g_{11}^k(u) \\ g_{12}^k(u) \\ g_{21}^k(u) \\ g_{22}^k(u) \end{bmatrix} = \partial_{u_k} w(u,S) \ . \qquad (27)$$

Recall that $g_{ij}^k$ are the coefficients of the deformation gradient (16), and the $a_{ij}$ have been defined in (15). A collection of equations (27) corresponding to $P$ different warping matrices $\{S_i\}_{i=1,\ldots,P}$ are concatenated in a linear system

$$\begin{pmatrix} a_{11}(u,S_1) & a_{12}(u,S_1) & a_{21}(u,S_1) & a_{22}(u,S_1) \\ a_{11}(u,S_2) & a_{12}(u,S_2) & a_{21}(u,S_2) & a_{22}(u,S_2) \\ \vdots & \vdots & \vdots & \vdots \\ a_{11}(u,S_P) & a_{12}(u,S_P) & a_{21}(u,S_P) & a_{22}(u,S_P) \end{pmatrix} \begin{pmatrix} g_{11}^k(u) \\ g_{12}^k(u) \\ g_{21}^k(u) \\ g_{22}^k(u) \end{pmatrix} = \begin{pmatrix} \partial_{u_k} w(u,S_1) \\ \partial_{u_k} w(u,S_2) \\ \vdots \\ \partial_{u_k} w(u,S_P) \end{pmatrix} . \qquad (28)$$

Because we only observe one realization of $I(x)$, as in 1D, (28) is smoothed with a 2D window $k_\Delta(x) = \Delta^{-2} k(\Delta^{-1} x)$ supported inside $[\Delta,\, \Delta]^2$. Since $d$ is $\mathbf{C}^3$, one can verify that convolving (28) with $k_\Delta$ yields

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

where

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

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

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

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 $N^{-1}$. We therefore require all the eigenvalues of $S$ to be greater than $N^{-1}$.

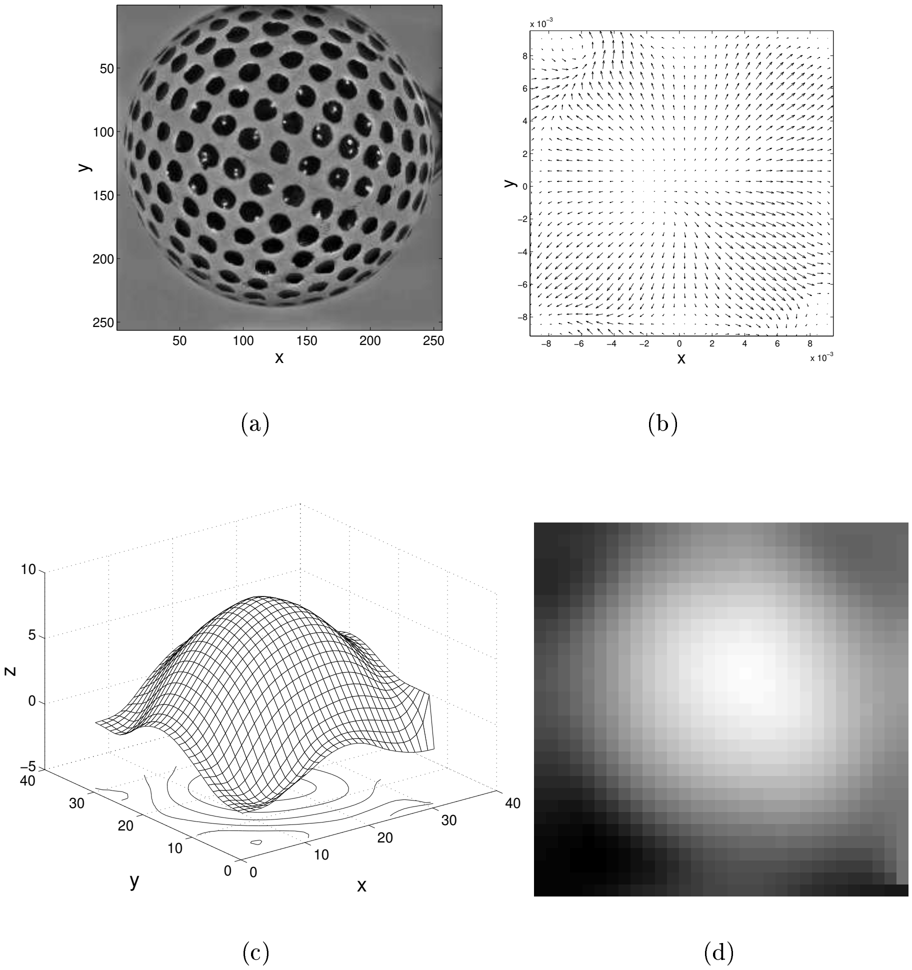
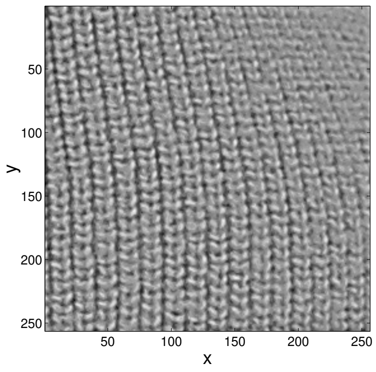



Figure 6: (a) Original image, with shading removed. (b) Normal vector computed. (c) Surface reconstructed from the normal vector. (d) Visualization of (c) as gray levels. Because (a) does not contain texture on its borders, there are errors in the border of the reconstructed surface (c).

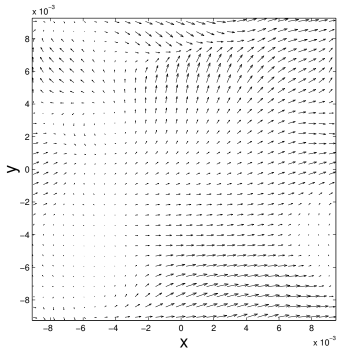
Denoting

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

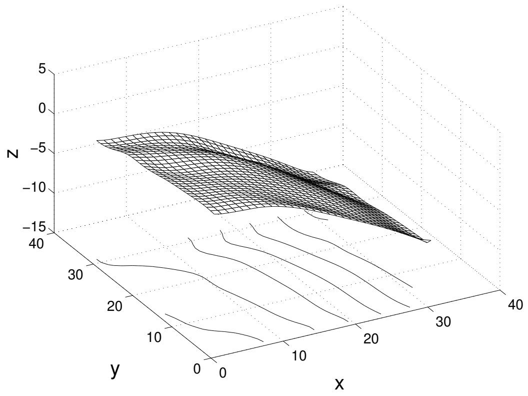
equation (29) suggests estimating the deformation gradient by inverting the



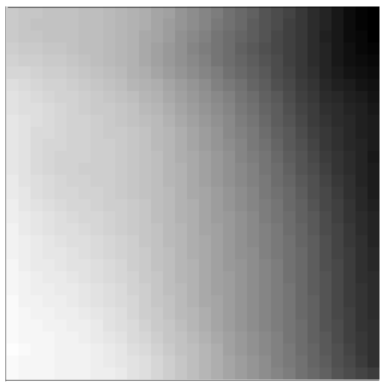

(a)




(b)




(c)



(d)

Figure 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.

linear system

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

To obtain at least as many equations as unknowns, one must choose warplets corresponding to at least four different warping matrices $S$, and in practice, we shall use even more. The estimators $\widehat{g_{ij}^k}$ are then calculated by singular value decomposition. The choice of matrices $S$ must be adjusted so that the resulting system is not degenerated. However, for some shape and texture configurations, even with a great number of different warping matrices, the system may remain underdetermined, mimicking the aperture problem in Optical Flow. Let us take the example of the cylindrical shape in Figure 1, which is curved in the horizontal direction. If it is covered with a texture that is constant in the horizontal direction, 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.

We parameterize $S$ under the form: $S = R_{\theta_1}\begin{pmatrix} s_1 & 0 \\ 0 & s_2 \end{pmatrix} R_{\theta_2}$. In Figures 6 and 7, calculations were performed with six directions $\theta_1 \in \{l\pi/6, l = 0, \ldots, 5\}$ 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 $\{S_i, i = 1\ldots, P\}$ should be selected adaptively according to $u$, but we used the same set throughout the image. This explains why we used 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 a FFT procedure detailed in Appendix B.2. There are thus a total of $168 = 24 \times 7$ sets of warplet coefficients to compute. We calculate $\widehat{\partial_a w}(u, S)$ with (30), and convolve it with $k_\Delta$. In our examples, we used $\Delta = N^{-1/5} = 0.33$, 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 [11].

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 3 extend automatically to 2D, and Figures 6 and 7 show that good numerical results are also obtained for non-separable deformations.

## 4.2 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 [8].

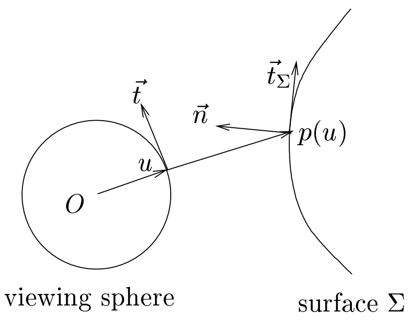



Figure 8: The slant-tilt frame field of $\Sigma$ is $(\vec{n}, \vec{t}_\Sigma, \vec{b}_\Sigma)$, where $\vec{b}_\Sigma = \vec{n} \times \vec{t}_\Sigma$ is tangent to the surface $\Sigma$, and perpendicular to the plane of the figure.

The basic *Shape from Texture* geometry assumes the image to be projected onto a viewing sphere, as shown in Figure 8. The perspective backprojection $p$ maps the viewing sphere to the surface $\Sigma$. 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 $\Sigma$ is called the slant-tilt frame field. The *slant* is the angle $\sigma$ between $\vec{n}$ and $\overrightarrow{Op(u)}$. The variations of the surface normal $\vec{n}$ depend upon the surface curvature, and are specified by

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

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 $\Sigma$ remains close to the optical axis of the camera. If this is not the case, a correction term must be taken into account ([18], App. A.2).

If $(\vec{x}_1, \vec{x}_2)$ is an orthonormal basis of the image plane, the *tilt* angle $\theta$ 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 [18, 9], the deformation gradient (5) is related to local surface parameters by

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

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

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

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

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

In order to recover local surface shape from the deformation gradient, the five parameters $(\theta, \sigma, \kappa_t, \kappa_b, \tau)$ must be estimated. After algebraic manipulations detailed in [9], this reduces to computing the tilt direction by linear 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 Figures 6(b) and 7(b), 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 Figure 6(b), since 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 $\Delta f = \partial_{x_1} n_1' + \partial_{x_2} n_2'$ [12]. This equation is solved with a standard finite difference scheme. The reconstructed surface depth $f(x_1, x_2)$ is plotted in Figures 6(c,d) and 7(c,d). 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. MATLAB routines reproducing the results of this section can be downloaded from `http://cermics.enpc.fr/~maureen/ShapeFromTexture.html`.

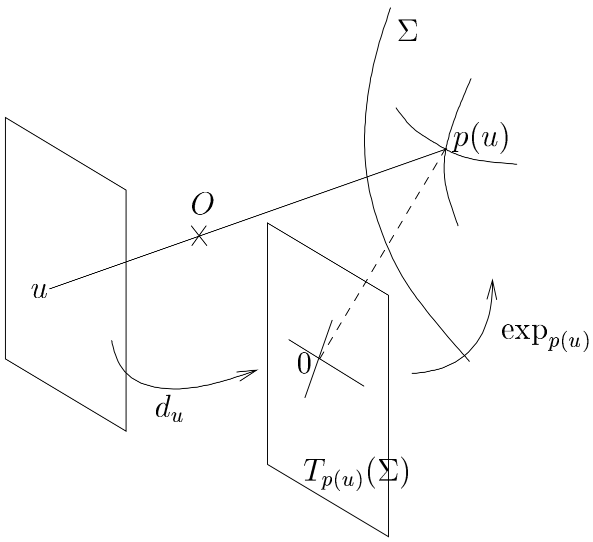



Figure 9: The exponential map $\exp_{p(u)}$ maps a neighborhood of 0 on the tangent plane to a neighborhood of $p(u)$ on $\Sigma$. We define a local mapping $d_u$ from the image plane to the tangent plane by $\exp_{p(u)}(d_u(x)) = p(x)$.

### 4.3 From Developable to General Surfaces

In Section 1, we modeled the image of a textured surface under perspective projection as

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

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

$$I(x) = a(x)\,\tilde{R}(p(x)) = a(x)\,R(d(x)) \ . \tag{34}$$

When $R$ is wide-sense stationary, Section 2 shows that the deformation gradient corresponding to $d(x)$ is solution of the Texture Gradient Equation, at small scales. We now propose a similar approach for a general, non-developable surface. Noting that the deformation gradient need only be the solution of the Texture Gradient Equation in the *small scale* limit ($\det S \to 0$), we introduce a local version of model (34). In order to transform $\tilde{R}$ into a 2D process, we project it locally onto the tangent plane to $\Sigma$ at $p(u)$, denoted $T_{p(u)}(\Sigma)$, through the *exponential map* [4]. This map, $\exp_{p(u)}$, projects a neighborhood of 0 in $T_{p(u)}(\Sigma)$ to a neighborhood of $p(u)$ on $\Sigma$, as depicted in Figure 9. It transforms radial lines stemming from 0 in $T_{p(u)}(\Sigma)$ into geodesics on $\Sigma$ 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,

$$I(x) = a(x)\,\tilde{R}(p(x)) = a(x)\,\tilde{R}(\exp_{p(u)}(d_u(x))) = a(x)\,R_{p(u)}(d_u(x)) \ , \quad (35)$$

which is a local version of model (34).

Let us now impose a homogeneity condition on $\tilde{R}$. First of all, it is natural to ask that $\mathsf{E}\{|\tilde{R}(p)|^2\}$ be independent of position $p \in \Sigma$. As a result, a local contrast renormalization 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 $D_u f(x) = f(d_u(x))$. 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 distorsions due to surface curvature, it does not make sense to require $R_{p(u)}$ to be wide-sense stationary. Moreover, even in the developable case, when $R$ is stationary, the deformation gradient is the solution of the Texture Gradient Equation with a resolution error of order $O(\det S)^{1/2}$ in equation (14). Introducing an additional error term of the same order to equation (14) is of no consequence. We can therefore tolerate the non-stationarity 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 $|v| \leq (\det S)^{1/2}$,

$$\|\mathsf{E}\{\langle R_{p(u)}, \overline{D}_u \psi_{v,S}\rangle \, \langle R_{p(u)}, \vec{\nabla}_x \overline{D}_u \psi_{v,S}\rangle^*\}\| = O(\det S)^{1/2} \|\vec{\nabla}_u w(u,S)\| \ , \quad (36)$$

where the gradients $\vec{\nabla}_x$ and $\vec{\nabla}_u$ are 2D vectors, and $\|\cdot\|$ is the Euclidean norm. This condition imposes a non-trivial relationship between the surface geometry and the type of texture homogeneity. If $\Sigma$ is developable, and $R_{p(u)} = R$ is stationary, then the left-hand side of (36) 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

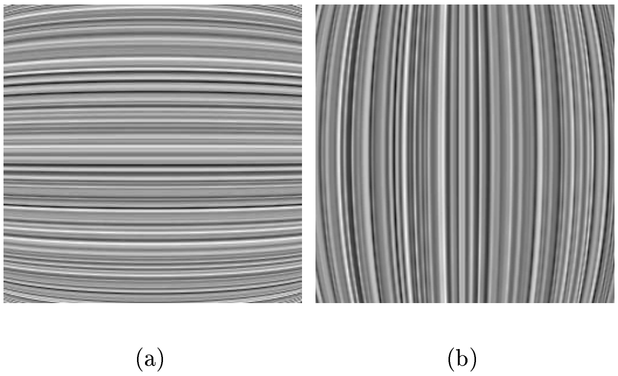


Figure 10: A non-isotropic texture on a sphere, oriented along parallels (a) or meridians (b), obeys the weak stationarity condition (36) at the equator.

process to a sphere $\Sigma$, or if $\tilde{R}$ is a non-isotropic texture oriented along parallels or meridians, and considered at the equator [5].

Under condition (36), one can prove that the deformation gradient

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

is the solution of the Texture Gradient Equation, with an error term that tends to zero when $\det S \to 0$. Moreover, the geometric relationships (32-33) between deformation gradient and local surface shape, which derive from a differential analysis, are also valid for $J_{d_u}(u)^{-1}\partial_{x_k}J_{d_u}(u)$. The surface normal $\vec{n}$ can therefore be recovered from the image of the textured surface, with the procedure described in Section 4.2.

Condition (36) 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. This condition, owing to the error term it tolerates, applies to a broad class of textures and surfaces, for which the Shape from Texture problem can be solved visually. The remaining issue is to specify precisely the class of shape-texture combinations satisfying (36).

## 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 fundamental transport equation, the *Texture Gradient Equation*. Under an appropriate homogeneity assumption on the original texture, the deformation gradient, which measures relative metric changes between the surface and the image plane, is the solution of the Texture Gradient Equation. We have introduced an estimator for the deformation gradient, and demonstrated the Shape from Texture algorithm on photographs.

Concerning texture homogeneity, more work is necessary to fully understand the relationship between the texture and the surface shape, so that the deformation gradient is solution of the Texture Gradient Equation, and thus the texture gradient a shape cue.

Another area for further research concerns texture modelization. The Lambertian assumption is somewhat restrictive, and a much wider class of natural textures could be considered with a 3D modelization [16, 23]. This could open promising directions for Shape from Texture.

## A Transport equation in two dimensions

We detail the calculations leading to (14). Let

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

and let $S(u) = J_d(u)^{-1} S_0$. Because of (13),

$$\frac{d}{du_k} w(u, S(u)) = \partial_{u_k} w(u, S(u)) + \sum_{i,j} c^k_{ij}(u)\, \partial_{s_{ij}} w(u, S(u)) \approx 0 \ . \qquad (37)$$

But

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

therefore

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

One can hence verify that

$$\sum_{i,j} c^k_{ij}(u)\, \frac{\partial}{\partial s_{i,j}} w(u, S(u)) = -\sum_{i,j} g^k_{ij}(u)\, a_{ij}(u, S(u)) \ , \qquad (38)$$

and replacing (38) in (37) proves (14), for $S(u) = S$.

# B Wavelet and warplet expressions

## B.1 Wavelet: modulated spline

The wavelet coefficients are calculated with a standard FFT procedure [19]: a wavelet transform can be obtained as a convolution product

$$\langle I, \psi_{u,s} \rangle = \int I(x) s^{-1} \psi(s^{-1}(x-u))\, dx = I * \tilde{\psi}_s(u) \ , \tag{39}$$

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 $\psi$ to be a modulated box-spline, whose Fourier transform is

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

For a discrete signal of size $N$, the wavelet $\tilde{\psi}_s$ and the variable $u$ are discretized over the sampling grid, and (39) is computed with an 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 an FFT procedure, using Fourier expressions derived from (19-20),

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

## B.2 Warplet: modulated Gaussian

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

$$\langle I, \psi_{u,S} \rangle = \int I(x) \det S^{-1} \psi(S^{-1}(x-u))\, dx = I * \tilde{\psi}_S(u) \ , \tag{40}$$

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 $\psi$ 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, and computing (40) with an FFT requires $O(N^2 \log N)$ operations.

Similarly to (40), $\langle I, \partial_a \psi_{u,S} \rangle = I * \partial_a \tilde{\psi}_S(u)$ can be computed with the FFT procedure, using the Fourier transform expressions

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

## References


[1] R. Bajcsy and L. Lieberman. Texture gradient as a depth cue. *Computer Graphics and Image Processing*, 5:52–67, 1976.

[2] D. Blostein and N. Ahuja. Shape from texture: integrating texture-element extraction and surface estimation. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 11(12):1233–1251, 1990.

[3] A. C. Bovik, N. Gopal, T. Emmoth, and A. Restrepo (Palacios). Localized measurement of emergent image frequencies by Gabor wavelets. *IEEE Trans. Inform. Theory*, 38(2):691–712, 1992.

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

[5] M. Clerc. *Estimation de processus localement dilatés et application au gradient de texture.* PhD thesis, Ecole Polytechnique, Paris, 1999. (in French).

[6] M. Clerc and S. Mallat. Shape from texture and shading with wavelets. In *Dynamical Systems, Control, Coding, Computer Vision*, volume 25 of *Progress in Systems and Control Theory*, pages 393–417. Birkhäuser, 1999.

[7] M. Clerc and S. Mallat. Estimating deformations of stationary processes. Technical report, CERMICS, 2000. (downloadable at `http://cermics.enpc.fr`).

[8] J. Gårding. Shape from texture for smooth curved surfaces in perspective projection. *J. Math. Imaging Vision*, 2:327–350, 1992.

[9] J. Gårding. Surface orientation and curvature from differential texture distorsion. In *Proc. 5th Int. Conf. on Computer Vision*, Cambridge, Massachussetts, 1995.

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

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

[12] B.K.P. Horn. *Robot Vision.* McGraw-Hill, 1986.

[13] B.K.P Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.

[14] 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 Proc.*, 7:773–780, 1998.

[15] K. Kanatani and T.C. Chou. Shape from texture: General principle. *Artificial Intelligence*, 38:1–48, 1989.

[16] T. Leung and J. Malik. Recognizing surfaces using three-dimensional textons. In *Proc. 7th Int. Conf. on Computer Vision*, Kerkyra, Greece, 1999.

[17] P.L. Lions, E. Rouy, and A. Tourin. Shape-from-shading, viscosity solutions and edges. *Numer. Math.*, 64:323–353, 1993.

[18] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *Int. J. Comp. Vision*, 23(2):149–168, 1997.

[19] S. Mallat. *A wavelet tour of signal processing.* Academic Press, 1999.

[20] J. Oliensis. Uniqueness in shape from shading. *Int. J. Comput. Vision*, 6:75–104, 1991.

[21] T. Randen and J.H. Husoy. Filtering for texture classification: A comparative study. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 21(4):291–301, 1999.

[22] R. Rosenholtz and J. Malik. Surface orientation from texture: Isotropy or homogeneity (or both)? *Vision Res.*, 37(16):2283–2293, 1997.

[23] P.H. Suen and G. Healy. The analysis and recognition of real-world textures in three dimensions. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 22(5):491–503, May 2000.

[24] B. Super and A. Bovik. Shape from texture using local spectral moments. *IEEE Trans. Patt. Anal. and Mach. Intell.*, 17(4):333–343, 1995.

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