

Shape from texture and integrability

D.A. Forsyth

Computer Science Division
U.C. Berkeley
Berkeley, CA 94720
daf@cs.berkeley.edu

Abstract

*We describe a shape from texture method that constructs a maximum a posteriori estimate of surface coefficients using both the deformation of individual texture elements — as in local methods — and the overall distribution of elements — as in global methods. The method described applies to a much larger family of textures than any previous method, local or global. We demonstrate an analogy with shape from shading, and use this to produce a numerical method. Examples of reconstructions for synthetic images of surfaces are provided, and compared with ground truth. The method is defined for orthographic views, but can be generalised to perspective views simply. **Keywords:** Shape from texture, texture, computer vision, surface fitting*

There are surprisingly few methods for recovering a surface model from a projection of a texture field that is assumed to lie on that surface. **Global methods** attempt to recover an entire surface model, using assumptions about the distribution of texture elements. Appropriate assumptions are **isotropy** [14] (the disadvantage of this method is that there are relatively few natural isotropic textures) or **homogeneity** [1, 2]. Methods based around homogeneity assume that texels are the result of a homogenous Poisson point process on a plane; the gradient of the density of the texel centers then yields the plane's parameters. However, deformation of individual texture elements is not accounted for.

Local methods recover some differential geometric parameters at a point on a surface (typically, normal and curvatures). This class of methods, which is due to Garding [6], has been successfully demonstrated for a variety of surfaces by Malik and Rosenholtz [10, 13]; a reformulation in terms of wavelets is due to Clerc [3]. The method has a crucial flaw; it is necessary either to know that texture element coordinate frames form a frame field that is locally parallel around the point in question, or to know the differential rotation of the frame field (see [7] for this

point, which is emphasized by the choice of textures displayed in [13]; the assumption is known as **texture stationarity**). For example, if one were to use these methods to recover the curvature of a doughnut dipped in chocolate sprinkles, it would be necessary to ensure that the sprinkles were all parallel on the surface (or that the field of angles from sprinkle to sprinkle was known). As a result, the method can be demonstrated to work only on quite a small class of textured surfaces. A second, important, difficulty lies in the data recovered; these methods all make local estimates of normal *and* curvature. But curvature is a derivative of the normal; as a result, while one local estimate may be helpful, there is no reason to believe that a collection of local estimates will be consistent. This is a problem of **integrability**. Surface interpolation methods have largely fallen out of fashion in computer vision, due to the uncertainty regarding the semantic status of surface patches in regions where data is absent. Shape from texture is a problem where an interpolate has an unquestionably useful role — it expresses the fact that, because one has a prior belief that surfaces are relatively slowly changing, incomplete local measurements of the surface normal can constrain one another and lead to good global estimates of the normal at some points.

We describe a shape-from-texture method that exploits the deformation of individual texels, *and* spatial properties of the distribution of texels, to produce a global approximating surface. To do so, we introduce a new texture model, which is a generalisation of that of Blake and Marinos [2]; we show how this model leads to a problem rather like classical shape from shading; and then demonstrate a variety of reconstructions resulting from this formulation.

1 A Texture Model

We model a texture on a surface as an inhomogenous, marked Poisson point process. A **point process** is some random procedure that results in points lying on a surface (exact definitions involve tedious measure theory [5]).

An **inhomogenous Poisson process** has the property that there is some measure M such that the expected number of points in a set \mathcal{U} is given by $E(\#\mathcal{U}) = M(\mathcal{U})$. We confine our attention to cases where this measure can be given by a density with respect to area, so that there is some function $\lambda(\mathbf{x})$ of position on the surface \mathbf{x} such that $E(\#\mathcal{U}) = \int_{\mathcal{U}} \lambda(\mathbf{x}) dA$ (where dA is the area form on the surface). The most familiar case is a **homogenous Poisson process**, where λ is a constant. We assume that we are working with a parametric family of intensities, $\lambda(\mathbf{x}; \mathbf{w})$, with parameter \mathbf{w} . A **marked point process** is one where each point carries a **mark**, drawn randomly according to some mark density from an available collection (for example, points might be red or blue).

In our model, the marks are texture elements (*texels* or *textons*, as one prefers; e.g. [9, 8] for automatic methods of determining appropriate marks) and the orientation of those texture elements with respect to some surface coordinate system. We assume that the marks are drawn from some known, finite set of classes of Euclidean equivalent texels. Each mark is defined in its own coordinate system; the surface is textured by taking a mark, placing it on the tangent plane of the surface at the point being marked, translating the mark's origin to lie on the surface point being marked, and rotating randomly about the mark's origin (according to the mark distribution). We assume that these texture elements are sufficiently small that they will, in general, not overlap, and that they can be isolated. Furthermore, we assume that they are sufficiently small that they can be modelled as lying on a surface's tangent plane at a point.

We assume that we have an orthographic view of a compact smooth surface, and that the boundary in this view is known. In section 4, we indicate what is required to relax the assumption of an orthographic view. We assume the viewing direction is the z -axis. Now consider one class of texture element; each instance in the image of this class was obtained by a Euclidean transformation of the model texture element, followed by a foreshortening that *in an appropriate coordinate system on the surface and in the image* can be written as

$$\mathcal{F}_i = \begin{pmatrix} \cos \sigma_i & 0 \\ 0 & 1 \end{pmatrix}$$

where σ is the angle between the surface normal at mark i and the z axis. The relevant coordinate system is the orthonormal coordinate system constructed around the slant and tilt vectors for the tangent plane at the mark. Between the i 'th mark and the j 'th mark, this coordinate system rotates. In particular, consider the transformation $\mathcal{T}_{i \rightarrow j}$, taking the i 'th mark *in the image* to the j 'th mark *in the image*. Ignoring the translation component, this transformation will have the form $\mathcal{T}_{i \rightarrow j} = \mathcal{R}_1 \mathcal{F}_j \mathcal{R}_2 \mathcal{F}_i^{-1} \mathcal{R}_3$ where

\mathcal{R} represents a rotation and \mathcal{F} is as above. In particular, \mathcal{R}_3 rotates the image coordinate system to the slant-tilt coordinate system at i , \mathcal{R}_2 accounts for the in-plane rotation of the texture element on the tangent plane and \mathcal{R}_1 rotates the new slant-tilt coordinate system to the image coordinate system. It is straightforward that $\det(\mathcal{T}_{i \rightarrow j}) = \cos \sigma_j / \cos \sigma_i$. Notice that the advantage of this formulation is *that no assumption regarding the relative rotation of the texture elements on the surface is required*. To recover a surface, we must: find the different texture elements; compute estimates of these determinants; and fit an appropriate surface model to the resulting data set. We defer discussion of finding the texture elements and estimating determinants to section 3.1 and show next how a surface can be recovered from this data.

2 Fitting a maximum a posteriori surface to the texture model

We have the following data: a family of surfaces \mathcal{S} ; a set of N textons, which lie on a surface S which may or may not belong to \mathcal{S} ; the image domain R to which this surface projects; the image of each of these textons in R ; the outline of S (we assume this is the boundary of R , ∂R); for each pair of points, $\det(\mathcal{T}_{i \rightarrow j})$; a parametric family of intensities $\lambda(\mathbf{x}; \mathbf{w})$, describing the different possibilities for the point process intensity; and the fact that S is smooth and compact. We wish to recover a representation of the visible component of S .

2.1 The analogy with shape from shading

To understand this problem further, assume that N tends to infinity, so that we have $m(\mathbf{x}, \mathbf{y}) = \cos \sigma(\mathbf{x}) / \cos \sigma(\mathbf{y})$, for any two distinct points $\mathbf{x}, \mathbf{y} \in R$. Since S is compact, there is at least one point \mathbf{y}^0 in R such that $\cos \sigma(\mathbf{y}^0) = 1$. We can identify this point (or these points), because $m(\mathbf{x}, \mathbf{y}^0)$ is always smaller than $m(\mathbf{x}, \mathbf{y})$, for any \mathbf{y}, \mathbf{x} . Now $m(\mathbf{x}, \mathbf{y}^0) = \cos \sigma(\mathbf{x})$; this means that our problem is exactly analogous with classical shape from shading with the illumination direction in the viewing direction.

Typically, the shape from shading literature approaches the reconstruction problem by representing S as a Monge patch (i.e. $(x, y, f(x, y))$), identifying the z component of the unit normal with $1/m(\mathbf{x}^0, \mathbf{y})$, and transforming the resulting partial differential equation to obtain $f_x^2 + f_y^2 = 1/m_{\frac{\partial f}{\partial x}}^2 + \frac{\partial f^2}{\partial y} = (1/m(\mathbf{x}, \mathbf{y}^0))^2 - 1 = g(\mathbf{x})$. There is no room for a detailed review of this old and well studied problem.

Uniqueness: Oliensis shows that *if* a solution exists, it is unique [11]. Note that points where $g = 0$ are particularly important, because the gradient field has a zero at these points. This uniqueness result is proven by noting that (1) the characteristic strips grow from points where $g = 0$

(but in a fashion that depends on the index of the underlying zero of the gradient field, the sign of which cannot be determined from g); (2) all characteristic strips approach the boundary transversally; (3) this constraint uniquely determines the sign of the indices of the gradient field; (4) once the indices are known, the reconstruction is unique. This means that the outline has a curious role in the uniqueness proof; one does not need to supply an outline precisely to prove uniqueness, but one does need to supply a closed curve which is (a) guaranteed to enclose all points where $g = 0$ and (b) guaranteed to have the characteristics (or, equivalently, the gradient) cross the curve transversally at every point.

Existence: Existence is significantly more difficult; it appears that for generic m such that (a) $0 \leq m \leq 1$ for all points in R and (b) $m = 0$ for points on the boundary of R , no solution exists.

Failure of the analogy: Apart from the fact that we wish to deal with small collections of texture elements, there are two important reasons that it is unwise to simply use a shape from shading method:

Estimation: shape from shading methods transform the variational criterion from $\int (\hat{n}_z - m)^2 dA$ to the form given above, to obtain a simpler PDE (the square-root in the unit normal is cleared by this transformation). However, the two criteria are not equivalent: they correspond to different noise models, and they lead to different minima, unless the value at the minimum is zero. Since we shall assume that m is subject to additive Gaussian noise, we cannot perform this transformation and still find a statistically meaningful solution.

Boundary conditions: the use of a Monge patch is not innocuous. In particular, Oliensis' uniqueness result shows the importance of the boundary (uniqueness can be shown only given the position of the boundary). But a Monge patch is badly behaved at the boundary; the gradient of the function must be infinite there. Any variational method should consider only functions with infinite gradient at the boundary — which, by the smoothness term, will sharply restrict the behaviour of the function within the domain. However, it is technically difficult to construct a family of functions that is guaranteed to turn away from the view at the boundary points. Furthermore, the approximation of curvature by second derivatives must fail near the boundary, meaning that the smoothness term is difficult to work with precisely at the points where it is most needed.

2.2 A likelihood function

We assume that our measurements of $d_{ij} = \det(\mathcal{T}_{i \rightarrow j})$ are subject to additive Gaussian noise, with constant standard deviation σ_d . While this is a somewhat dubious noise model, it is not outrageous and it leads to a tractable problem. Now we have a series of measurements

$\mathbf{d} = (d_{00}, \dots, d_{ij}, \dots)$ corrupted with zero mean, additive Gaussian noise. They are conditionally independent given the surface, and so yield a log-likelihood

$$-\log P(d_{ij}|z) = \frac{1}{2\sigma_d^2} \left(\sum_{ij} \left(d_{ij} - \frac{\hat{n}_z(\mathbf{x}_i; z)}{\hat{n}_z(\mathbf{x}_j; z)} \right)^2 \right) + K$$

where K is some normalising constant. This likelihood incorporates all information obtainable from the point-to-point deformation of texture elements in viewing, in the absence of rigid constraints on element-to-element rotation.

There is a second source of information about the surface; we expect that texture elements are distributed around the surface in a particular manner, consistent with an inhomogenous Poisson point process. This can be expressed by looking at the **quadrat counts**. In particular, we subdivide the parameter domain into a fixed collection of non-intersecting sets (called **quadrats** by analogy with the procedure for point processes on the plane [4]). Notice that the natural alternative — the K-statistic, which is essentially, a histogram of the number of points within circles of increasing radius [12] — is very difficult to use for a curved surface, as it requires solving geodesic equations. The count of points in each quadrat is multinomial, given the number of points N , any particular choice of surface z and of intensity function $\lambda(\mathbf{x}; \mathbf{w})$. We write the i 'th quadrat as q_i , and write

$$p_i = \frac{\int_{q_i} \lambda(\mathbf{x}; \mathbf{w}) dA}{\sum_j \int_{q_j} \lambda(\mathbf{x}; \mathbf{w}) dA}$$

In particular, we have

$$P(\#(q_1) = n_1, \#(q_2) = n_2, \dots | z, \mathbf{w}, N) = \left(\frac{n_1! n_2! \dots}{N!} \right) p_1^{n_1} p_2^{n_2} \dots$$

These quadrat counts are a discrete and generalised version of the process intensity cue used by Blake and Marinos [2]. Now the quadrat counts — which measure the extent to which the process intensity implied by the reconstruction corresponds to the intensity chosen — are independent of the normal measurements given the surface and the intensity. This means that we have a log-likelihood

$$\log P(\text{measurements} | \text{surface, intensity, no. of points}) = \log P(d_{ij}|z) + \log P(\#(q_1), \#(q_2), \dots | z, \mathbf{w}) + K$$

2.3 Priors

Our prior will be non-zero except for a family of surfaces (which we describe in section 3.2). We require a prior to enforce smoothness on z . However, it is unwise to use the second derivative approximation to curvature, because

this is very badly behaved at the boundary (where the surface is nearly vertical). Instead, we compute the norm of the shape operator and sum over the surface. We must now choose a surface to maximise the negative log posterior, given by

$$\frac{1}{2\sigma_d^2} \left(\sum_{ij} (d_{ij} - \frac{\hat{n}_z(\mathbf{x}_i; z)}{\hat{n}_z(\mathbf{x}_j; z)})^2 \right) + K + \left(\frac{1}{2\sigma_k^2} \right) \int_R (\kappa_1^2 + \kappa_2^2) dA + \log \left(\left(\frac{n_1! n_2! \dots}{N!} \right) p_1^{n_1} p_2^{n_2} \dots \right) + \log \pi(\mathbf{w})$$

This posterior may, at first glance, look ambiguous; in particular, we have only ratios of cosines, not the cosines themselves. This ambiguity does not manifest itself in practice, because of the quadrat count term. Assume that we obtain cosines from the ratios by choosing the value of the maximum cosine (i.e. the angle between the most frontal texture element and the view direction). If the value of this choice is too low, the surface will (generally) bulge out toward the view; in turn, this means that the area represented by some quadrats will increase significantly, and the value of the quadrat term will decline sharply.

3 Implementation

3.1 Estimating Texture Foreshortening

Finding texture elements is an established technology (e.g. [8, 9]). For our demonstration of this reconstruction method, we assume that the locations and approximate scales of texture elements from each class are known. Now in this case, we need to estimate the relative foreshortening, $\det(\mathcal{T}_{i \rightarrow j})$. One possibility is to construct an affine transformation from element i to element j that minimises (say) the sum of least-squares errors — perhaps using the iterative method of Malik and Rosenholtz [10] — and take the determinant of that transformation. In fact, because we care only about the relative foreshortening, it is unnecessary to use this relatively elaborate (and expensive) method.

In particular, we note that for a function f , a region R and an affine transformation \mathcal{T} , $\int_{\mathcal{T}(R)} f(\mathcal{T}(\mathbf{x})) d\mathbf{x} = (1/|\mathcal{T}|) \int_{\mathcal{T}(R)} f(\mathcal{T}(\mathbf{x})) d(\mathcal{T}\mathbf{x})$. Now if f falls to zero rather quickly, so that it is approximately zero close to the boundaries of both R and $\mathcal{T}(R)$, we have that $\int_{\mathcal{T}(R)} f(\mathcal{T}(\mathbf{x})) d\mathbf{x} \approx (1/|\mathcal{T}|) \int_{(R)} f(\mathcal{T}(\mathbf{x})) d(\mathcal{T}\mathbf{x})$. This suggests integrating with respect to a weight function, which declines to zero quickly outside a small domain. We use this approximation, which can be shown to be successful over a satisfactory range of values. Note that this

method requires that texture elements be separated, something that cannot be guaranteed with a Poisson model; because the texture elements are small, the bias resulting from approximating the hard-core model required with a Poisson model is tiny.

Note that it is difficult to estimate $\det(\mathcal{T}_{i \rightarrow j})$ for either i or j at a substantial inclination to the frontal direction. The value will be either very large, or very small; and, since one of the texture elements involved will cover relatively few pixels, it will be particularly difficult to estimate. It is not always possible to tell when this estimation error is likely to occur as there may be regions in the interior where the inclination is large, but this error is pretty much guaranteed to occur close to the rim. This suggests that texture elements close to the rim should be ignored. We describe the method for doing so below.

3.2 An appropriate family of surfaces

The success of this approach depends sharply on the use of an appropriate family of surfaces. In particular, it is important to ensure that any surface considered *turns away from the eye at the boundary*. This requirement substantially reduces the available ambiguities (e.g. see Oliensis' uniqueness proof [11]). It is, in practice, relatively straightforward to ensure that this criterion is met. We describe our method for a circular boundary; the extension to any Jordan curve is straightforward; an extension to any boundary is technically more tricky, but appears tractable. We will work with parametric surfaces. Given a circular boundary, it is natural to use polar coordinates. We write our surface as $(p(\rho) \cos \theta, p(\rho) \sin \theta, z(\rho, \theta))$ and the surface's unit normal as $\hat{n} = (\hat{n}_x, \hat{n}_y, \hat{n}_z)$. The boundary occurs at $\rho = 1$. Now if $p(0) = 0$, $p(1) = 1$, $dp/d\rho(1) = 0$, and $p'(\rho) > 0$ for $0 \leq \rho < 1$, we have that $\hat{n}_z(1, \theta) = 0$, independent of our choice of z or θ . This means that this surface has the curve $(\cos \theta, \sin \theta)$ as a component of its outline.

By this construction we have eliminated a substantial number of ambiguities, because we are confining our attention to surfaces whose outline lies on ∂R , something one cannot do with a Monge patch representation. We now represent z as an element of a linear family, whose basis is given by a tensor product. In particular, we model $z = \sum_{kl} a_{kl} \phi_k(\rho) \psi_l(\theta)$. By requiring that $\partial \phi_k / \partial \rho(0) = 0$ for all i , we ensure a properly defined surface normal at $\rho = 0$. We must also require that $\partial z / \partial \rho$ does not change sign along the curve $\rho = 1$ (or else the surface will be singular at the points where this expression changes sign).

3.3 Numerical Methods

Efficiency: the original form of the posterior leads to an extremely slow method, because there are N^2 ratio terms for N texture elements in the surface orientation term. Most of these terms are not independent. Instead, we will

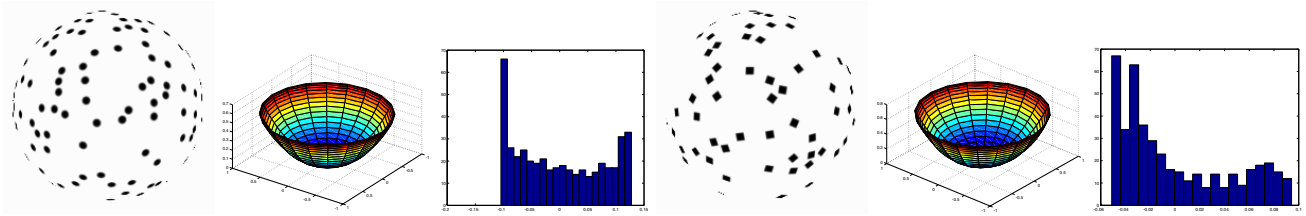


Figure 1: *Reconstruction examples for two surfaces with known structure. The figures on the top are rendered images of textured surfaces. On the left, the texture elements are circles, so that rotation of a texture element is moot. On the right, the texture elements are squares, and are rotated randomly. No known local shape from texture method can deal with such an example, without being informed in advance of the rotation component. The figures in the center show views of the reconstructed surfaces. The original surfaces are unit radius hemispheres. The reconstructed surfaces are guaranteed to pass through the point $(0, 0, 0)$ at the center of the domain. We can compare the reconstruction with ground truth by computing the distances between points on the reconstructed surface and the point $(0, 0, 1)$ (which is the center of the original sphere). The figures on the bottom are histograms of the distances for the reconstructed surfaces offset by the mean, suggesting that they are strongly similar to the original surface (the distance error lies within a range of rather less than 10% of the radius).*

choose s_i to minimise

$$\sum_{i,j} \left(d_{ij} - \frac{s_i}{s_j} \right)^2$$

and then maximise

$$\begin{aligned} & -\frac{1}{2\sigma_d^2} \left(\sum_{ij} \left(\frac{s_i}{s_j} - \frac{\hat{n}_z(\mathbf{x}_i; z)}{\hat{n}_z(\mathbf{x}_j; z)} \right)^2 \right) + K \\ & - \left(\frac{1}{2\sigma_k} \right) \int_R (\kappa_1^2 + \kappa_2^2) dA - \\ & \log \left(\left(\frac{n_1! n_2! \dots}{N!} \right) p_1^{n_1} p_2^{n_2} \dots \right) + \log \pi(\mathbf{w}) \end{aligned}$$

Excising the rim: we ignore all texture elements for which $r > 0.95$, so as to avoid very poor normal estimates. This means that we have no measurement of the geometry of the surface for a strip around the boundary. Now we can benefit from the nature of the involvement of the boundary in Oliensis' proof here (see section 2.1) if we can guarantee that (a) there are no frontal points of the surface in this strip and (b) that the characteristics cross the internal boundary of this strip transversally. The first is a restriction on the surfaces that can be reconstructed, but a reasonable one. The second is achieved by insisting that $\partial z / \partial \rho > 0$ inside this strip.

Constraints: requiring that $\partial z / \partial \rho$ is positive for $\rho > \rho_0$ is a set of linear constraints on a_{kl} ; we can evaluate $\psi_l(\theta)$ at a discrete set of points (ρ_p, θ_p) and use the linear constraints

$$\begin{aligned} \sum_{kl} a_{kl} \frac{\partial \phi_k}{\partial \rho}(\rho_p) \psi_l(\theta_p) & > 0 \\ \mathbf{Ca} & > 0 \end{aligned}$$

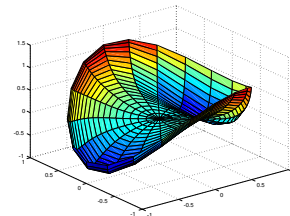


Figure 3: *Local minima exist for this method, and it must be started at random to avoid them. This figure shows a local minimum for the case of the sphere with square texture elements in figure 1; the value of the negative log posterior is about 1.5 times the value for the correct reconstruction.*

and we need to maximise the negative log-posterior subject to these constraints.

4 Discussion

Maximisation: the resulting problem is difficult. For a reasonable basis (four ϕ_k and five ψ_l , leading to a total of 20 elements), it can be handled by Matlab's `constr` function, if this is started at a feasible point. There are clearly a variety of local minima, some of which are illustrated below. These can be dealt with by running the method at a selection of randomly chosen feasible start points; in our experience, of the order of five starts usually leads to a satisfactory reconstruction. It is wise to include $\psi_l(\theta) = 1$ in the basis to allow for surfaces with rotational symmetry, and to constrain ϕ_k and ψ_l to ensure that the point $\rho = 0$ has a properly defined surface normal.

Homogeneity: we have implemented this method for the restricted case of an homogenous Poisson distribution.

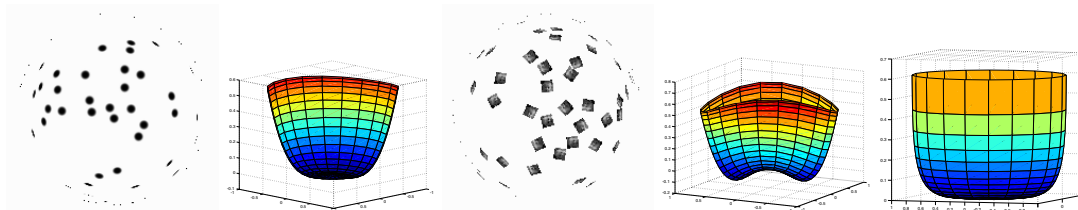


Figure 2: Reconstruction examples for a surface with more complex structure. The figures on the top show a rendered images of a textured surface. The texture elements for the surface on the top right are rotated randomly. No local shape from texture method can deal with such an example. The figures in the center show the best local minima obtained using our method of random restarting for each case; the figure on the bottom shows a view of the original surface, which is not a basis element. Notice that the reconstruction is, again, satisfactory, though the method cannot suppress small oscillations in the surface orientation in between texture elements — of the order of 10% of the depth of the surface — because the basis is relatively small.

This is because there are often too few texture elements to make a reasonable estimate of $\lambda(\mathbf{x}; \mathbf{w})$.

4.1 Results

The method yields satisfactory reconstructions, as the figures show. We used $\sigma_k \approx 100 * \sigma_d$, to deemphasize the significance of the smoothness constraint. Figure 1 shows reconstructions for a sphere — a surface element contained in the basis, to allow us to detect local minima. Figure 2 shows reconstructions for a flattened sphere. Notice that we have used some texture distributions that allow local curvature estimates, and some that do not. The same start point generator was used for each reconstruction. Figure 3 shows a typical local minimum encountered in reconstructing a sphere.

There are two basic sources of information about shape in surface texture. One is the density of the elements on a surface; the other is the deformation of the shape of the elements. We have shown how to integrate these sources for an orthographic view of a curved surface. Extending our assumption of orthographic viewing to perspective viewing involves a straightforward adjustment to our posterior. One substitutes $\cos \sigma_i / z_i$ for $\cos \sigma_i$ and proceeds. We make no claim as to the practical stability of the resulting method, which we have not investigated. The formalism extends to a very wide family of textures; we have demonstrated reconstructions for homogenous textures. There is a small bias introduced by the fact that we must assume that texture elements lie in the tangent plane of a surface.

Shape from texture is intriguing, because of the curious relationship between local and global information in the problem. Our method is not local; but it is not purely global, because it takes advantage of the fact that uniqueness requires that a curve be known which is crossed transversally by the gradient, rather than the actual outline.

References

- [1] Y. Aloimonos. Detection of surface orientation from texture. i. the case of planes. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 584–593, 1986.
- [2] A. Blake and C. Marinos. Shape from texture: estimation, isotropy and moments. *Artificial Intelligence*, 45(3):323–80, 1990.
- [3] M. Clerc and S. Mallat. Shape from texture through deformations. In *Int. Conf. on Computer Vision*, pages 405–410, 1999.
- [4] N.A. Cressie. *Statistics for Spatial Data*. Wiley, 1993.
- [5] D.J. Daley and D. Vere-Jones. *An Introduction to the theory of point processes*. Springer-Verlag, 1988.
- [6] J. Garding. Shape from texture for smooth curved surfaces. In *European Conference on Computer Vision*, pages 630–8, 1992.
- [7] J. Garding. Surface orientation and curvature from differential texture distortion. In *Int. Conf. on Computer Vision*, pages 733–9, 1995.
- [8] T. Leung and J. Malik. Detecting, localizing and grouping repeated scene elements from an image. In *European Conference on Computer Vision*, pages 546–555, 1996.
- [9] J. Malik, S. Belongie, J. Shi, and T. Leung. Textons, contours and regions: cue integration in image segmentation. In *Int. Conf. on Computer Vision*, pages 918–925, 1999.
- [10] J. Malik and R. Rosenholtz. Computing local surface orientation and shape from texture for curved surfaces. *Int. J. Computer Vision*, pages 149–168, 1997.
- [11] J. Oliensis. Uniqueness in shape from shading. *Int. J. Computer Vision*, pages 75–104, 1991.
- [12] B.D. Ripley. *Pattern recognition and neural networks*. Cambridge University Press, 1996.
- [13] R. Rosenholtz and J. Malik. Surface orientation from texture: isotropy or homogeneity (or both)? *Vision Research*, 37(16):2283–2293, 1997.
- [14] A.P. Witkin. Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17:17–45, 1981.