A General Algorithm For Shape From Shading

J. Oliensis
P. Dupuis

COINS TR92-73

A General Algorithm For Shape From Shading[1]

John Oliensis†

Department of Computer Science

University of Massachusetts at Amherst

Amherst, Massachusetts 01003

and

Paul Dupuis‡

Box F, Division of Applied Mathematics

Brown University

Providence, Rhode Island 02912

Abstract

A general algorithm for reconstructing shape from shading is described. This algorithm incorporates an earlier local algorithm that has previously been shown to be capable of fast, robust surface reconstruction for general surfaces if a small amount of information on the surface is provided. The required information is 1) a list of those singular points (the brightest image points) corresponding to local minima of the surface height (as opposed to the other possibilites of a local maximum or a saddle point), and 2) the heights of these singular points. The new proposed algorithm is capable of determining this information automatically, and thus can reconstruct a general surface from shading with no a priori information on the surface. In experimental tests on complex synthetic images, this algorithm has produced good surface reconstructions over most of the image. For 128 ×128 images, the reconstruction takes less than 30 seconds on a DECstation 5000. Moreover, the algorithm appears noise resistant, giving good reconstructions even in the extreme case of an added pixel noise of ±10%.

Keywords: Shape from shading, shape from X, optimal control, stochastic optimal control, partial differential equations.

[1]This work was supported by the National Science Foundation under grants IRI-9113690, CDA-8922572†and NSF-DMS-9115762‡, and by a grant from DARPA, via TACOM, contract number DAAE07-91-C-R035†.

A General Algorithm For Shape From Shading

Abstract

A general algorithm for reconstructing shape from shading is described. This algorithm incorporates an earlier local algorithm that has previously been shown to be capable of fast, robust surface reconstruction for general surfaces if a small amount of information on the surface is provided. The required information is 1) a list of those singular points (the brightest image points) corresponding to local minima of the surface height (as opposed to the other possibilites of a local maximum or a saddle point), and 2) the heights of these singular points. The new proposed algorithm is capable of determining this information automatically, and thus can reconstruct a general surface from shading with no a priori information on the surface. In experimental tests on complex synthetic images, this algorithm has produced good surface reconstructions over most of the image. For 128 $\times$128 images, the reconstruction takes less than 30 seconds on a DECstation 5000. Moreover, the algorithm appears noise resistant, giving good reconstructions even in the extreme case of an added pixel noise of $\pm$10%.

Keywords: Shape from shading, shape from X, optimal control, stochastic optimal control, partial differential equations.

1 Introduction

Shape from shading has proven to be a difficult problem, even under the standard idealizing assumptions of a Lambertian surface, known light source, and no shadowing or occlusion. Recently, a new approach has been developed, based on relating shape from shading to an "equivalent" optimal control problem [15,516,21,22,1]. The advantages of this approach are both theoretical and practical. First, it makes possible an easy uniqueness proof for the surface corresponding to a shaded image (under suitable conditions); thus shape from shading, contrary to previous belief, is often a well–posed problem. Second, the approach leads naturally to an algorithm for surface reconstruction that is simple, fast, provably convergent, and (again, under suitable conditions) provably convergent to the correct surface. In experiments on 200 $\times$ 200 and 128 $\times$ 128 real and synthetic images, convergence took fewer than 15 iterations, and less than 10 seconds on a DECstation 5000 [15,5,1]. Typically, the time for convergence is expected to be approximately constant independent of the image size. Finally, it has been proven that the algorithm converges to the correct continuous surface in the limit where the pixel spacing is infinitely small [4,13]. These results are to be contrasted with traditional reconstruction algorithms which typically require thousands of iterations, and for which no convergence results are known [11].

The algorithm above is in a sense local, and is referred to below as the local algorithm, to distinguish it from the global extension of the algorithm studied in the present paper. The local algorithm uses the fact that a singular point—defined as a maximally bright image point—uniquely determines the surface within some image neighborhood [3,23,24,18]. More precisely, this is true for singular points at which the surface height has a local minimum or local maximum, as is generally true for about half the singular points. The algorithm is guaranteed to give the correct surface up to an overall translation in depth near each local minimum singular point in the image[2]. The image region around such a singular point where the surface is correctly

[2]Local maxima points could also be used. The local algorithm has two versions, one using local minimum singular points and the other using local maxima.

reconstructed will be referred to as the domain of attraction of the singular point.

The surface over the whole image will also be determined, and correctly reconstructed by the local algorithm, if it is known which of the singular points in the image correspond to local minima of the surface height, and if the surface heights at these points are available. Under these conditions, we have the uniqueness and provable correct convergence mentioned above. In fact, only the first item is essential. Since the domains of attraction of different singular points generally "touch" in the image, the relative heights of these points can be determined easily [17, 18]. Therefore, for correct reconstruction, it is mainly required to determine which of the finite—and generally small—number of singular points correspond to local minima of the surface. If this information is not known, then there is a potential convex–concave–saddle ambiguity for the surface shape at each singular point[3]. In the experiments mentioned above, this information was provided to the local algorithm.

In this paper, we describe a general shape from shading algorithm which can determine this information automatically. This algorithm, which we refer to as the global algorithm, reconstructs shape from shading with no a priori information about singular points or the surface. The global algorithm incorporates the earlier local algorithm, but also makes use of a global consistency analysis of shape from shading similar to that of [18]. Typical reconstruction times are less than 30 seconds on a DECstation 5000 for 128×128 images. However, although the global algorithm can probably be proven to converge to the correct continuous surface in the limit of small pixel spacing (under appropriate conditions), it is more sensitive to discretization and other errors than the local algorithm.

The uniqueness and convergence results mentioned earlier, for the "local" case when the heights at local minimum singular points are known, require very minimal assumptions on the surface smoothness. Unlike most previous work, the surface is not required to be twice differentiable, which is quite natural since the image irradiance equation itself involves only

[3]Inflection points are ruled out in the generic situation that the image intensity has a nonsingular second derivative matrix, with unequal eigenvalues.

first derivatives. Actually, generalized nondifferentiable solutions can also be handled—the experimental correlate to this fact is that surfaces reconstructed by the local algorithm frequently display orientation discontinuites. (This occurs if some of the standard assumptions are violated, such as the Lambertian nature of the imaged surface, or if the singular points are wrongly identified.) The robustness of the theoretical analysis and local algorithm is partly due to the fact that these results do not depend on the surface being smooth.

However, for the global algorithm and global uniquenes (e.g. [18]), it is necessary to assume that the surface to be reconstructed is twice differentiable. Note that this is a weaker assumption than the usual smoothness condition [11]. Otherwise, there is no way to resolve the ambiguity in assigning the nature of the surface at singular points. Experimentally this is seen in the fact that the local algorithm reconstructs a surface that corresponds exactly to the original shaded image for any assignment of local minimum singular points (although the reconstructions generally display orientation discontinuities). There are also a few additional requirements necessary for the global algorithm. Because the assumptions on the surface are more stringent for the global algorithm, and because no information on the surface is provided, it is expected to be less robust than the local algorithm. Nevertheless, even though some of its requirements are violated in our experiments, the global algorithm appears to correctly reconstruct the surface over most of our test images, and to be suprisingly immune to image noise. Moreover, the global algorithm in its present form exploits only some of the consistency requirements between different singular points in the image. A later version of the algorithm, which exploits these redundant constraints more fully, should be capable of robust identification of all local minimum singular points and correct surface reconstruction, at the cost of speed.

Our experiments with the global algorithm have so far been carried out only for synthetic surfaces and images, though these are rather complex. Nevertheless, this algorithm is the first capable of reconstructing shape from general images with some degree of robustness, and is orders of magnitude faster than traditional variational algorithms [7,6,10,12].

2 Review of the Local Algorithm

We consider the standard idealized version of the shape from shading problem, in which the intensity at an image point (x, y) is given by the image irradiance equation

$$I(x, y) = \hat{L} \cdot \hat{n},$$

where $\hat{L}$ is the light source direction, the optical axis is along the $-z$ direction, and $\hat{n}$ is the surface normal at the corresponding surface point. With the surface represented by the height function $z(x, y)$,

$$\hat{n} \equiv \frac{(-z_x, -z_y, 1)}{(1 + z_x^2 + z_y^2)^{1/2}}.$$

For simplicity, our theoretical exposition will be for the case of illumination along the viewing direction, i.e. $\hat{L} = \hat{z}$, even though our experiments are done for general illumination direction, and the theory is also valid for that case. In this situation, which we refer to as that of "vertical light", the image irradiance equation is

$$I(x, y) = \frac{1}{(1 + |\nabla z(x, y)|^2)^{1/2}}.$$

Singular points occur when I takes on the maximal value $I = 1$. At such points $\nabla z = 0$, i.e. the surface has an extremal point. Thus we may take the surface height at a singular point to have either a local minimum, a local maximum, or a saddle point. (For the continuous problem, the fourth possibility of an inflection point generally can be ruled out from the intensity data alone. Its occurrence is nongeneric, as discussed in in Section 3. However, our experimental surface reconstructions succeed even though the test surfaces do have inflection singular points.) Note that the importance of singular points in determing the surface stems from the fact that only at these points is the surface orientation uniquely determined. For "vertical light", the surface is locally horizontal at all singular points.

In [15,16], it is shown that the surface height z has the following explicit representation (see

also [21,22]). Define the terminal cost function

$$g(x,y) = \begin{cases} z(x,y) & \text{if } (x,y) \text{ is a local minimum singular point,} \\ +\infty & \text{otherwise.} \end{cases}$$

Then

$$z(x,y) = \inf \left[\int_0^\tau ds \frac{1}{2} \left(\dot{x}(s)^2 + \dot{y}(s)^2 + \left(\frac{1}{I^2(x(s),y(s))} - 1 \right) \right) + g(x(\tau), y(\tau)) \right]$$

where the infimization is over all $\tau < \infty$, and all paths $(x(s), y(s))$ in the image plane beginning at (x,y), i.e. with $(x(0), y(0)) = (x,y)$. Thus $z(x,y)$ is given as the least value of the cost for an optimal control problem. The paths which give the infimum of the cost can be identified with characteristic strips, and also turn out to be paths of steepest descent on the surface [15, 18, 17,9]. Thus, all optimal paths converge to image points corresponding to local minima of the surface height, which are exactly the local minimum singular points. This must be the case, since the terminal cost $g(x(\tau), y(\tau))$ imposes an infinite cost if the path terminates anywhere other than at a singular point.

An algorithm for computing this minimal cost value is obtained from the principle of dynamic programming. A brief heuristic description follows; for more detail and rigorous proofs, consult [15,4]. This principle essentially states that the minimal cost $U(x,y,T)$ over all paths lasting for a time T and starting at a given point (x,y) is related to the minimal cost $U(x+\delta x, y+\delta y, T-\delta T)$ over all paths starting at nearby points $(x+\delta x, y+\delta y)$ and lasting a shorter time $T-\delta T$. If for simplicity we neglect the terminal cost term $g(\cdot)$, this can be converted to a differential relation

$$\begin{aligned} \frac{\partial U}{\partial T}(x,y,T) = \inf_{(\dot{x}(0),\dot{y}(0))} \{ & \\ & \frac{1}{2} \left(\dot{x}^2(0) + \dot{y}^2(0) + \left(\frac{1}{I(x,y)^2} - 1 \right) \right) \\ & + \frac{\partial U}{\partial x}(x,y,T)\dot{x}(0) + \frac{\partial U}{\partial y}(x,y,T)\dot{y}(0) \} \end{aligned}$$

where the infimization is now just over the initial velocity of motion at the start point (x,y). This equation is a recursion relation, which allows $U(x,y,T)$ to be computed forward in time,

eventually leading to the computation of $U(x,y,T) \rightarrow U(x,y,\infty) \equiv z(x,y)$, the minimal cost over infinite time paths. An initial condition for this recursion relation is provided by the fact that for zero time paths, the minimal cost $U(x,y,0) = 0$ (again neglecting the terminal cost).

The actual, rather simple algorithm, and experimental results obtained with it, can be found in [15,16]. It can be shown that this algorithm converges monotonically, stably, and quickly [15]. The height function z as given by the explicit representation in (2) is easily proven to be continuous, but need not necessarily be differentiable everywhere.

Strictly, the representation in (2), and the proof that the local algorithm converges to the correct surface, are valid only for image points that are connected to a singular point by a steepest descent curve whose projection is contained entirely in the image. The surface reconstruction will be wrong for those points, typically near the image boundary [17], for which this is false. To take care of this subtlety, we impose the additional assumption

A2.1 *The image region $\mathcal{G}$ is such that the image projection of the steepest descent direction on the surface is always inward at the boundary of $\mathcal{G}$.*

When this is the case, no steepest descent curve can exit the image, and the reconstruction will be correct at all image points.

We have described the local algorithm in the version using the local minimum singular points. An alternative version of the algorithm uses data from the local maximum singular points. For this algorithm, we use the representation of the height $z(\cdot)$ as the supremum of the cost over all image paths. Not surprisingly, the cost for a path in this version of the algorithm is the negative of that in (2).

3 The Global Algorithm

The most important additional assumption for the global theoretical analysis is that the height function $z(x,y)$ is twice differentiable. In addition, we assume that the singular points in the

image, and, corespondingly, the extremal points on the surface, are isolated, and that the surface has nonzero curvature at these points. The second condition explicitly rules out inflection singular points. (Again, these additional assumptions are not always respected in our experiments.)

As noted in [18, 17], the two additional conditions are generically true for twice differentiable surfaces $z(x, y)$—that is, they are valid for almost all such functions, and if invalid for some particular function $z_{NG}(\cdot)$, then there are functions arbitrarily close to $z_{NG}(\cdot)$ such that the conditions do hold. However, though this is the case for continous surfaces, inflection singular points as well as non–isolated singular points occur frequently for discretely sampled images and surfaces, and are a serious concern for the global algorithm. Nevertheless, the algorithm succeeds in giving good reconstructions despite the presence of such points.

The above two conditions are implied by the following assumption on the intensity data $I(\cdot)$ [18], which is again generically valid:

A3.1 *All singular points in the image are isolated. At these points, the matrix of second derivatives of $I(\cdot)$ with respect to x, y has nonzero, unequal eigenvalues.*

In addition, we assume below that $I(\cdot)$ is defined over a region $\mathcal{G}$ satisfying A2.1. The uniqueness of $z(\cdot)$ given $I(\cdot)$ under essentially these assumptions was established in [18, 17]. The algorithm described here exploits the constraints producing this uniqueness.

Suppose that a singular point $s \equiv (x, y)$ is known to be an isolated local minimum of the height $z(\cdot)$. The results of [15] imply that the local algorithm, if provided $z(s)$ as an initial datum, will reconstruct the correct surface in the domain of attraction of s. Below we use "image curve of steepest descent" as a shorthand for the image projection of a curve of steepest descent on the surface. Then the domain of attraction for s, denoted A_s, consists of all image points q such that the image curve of steepest descent passing through q converges to s. (This curve is uniquely determined since $z(\cdot)$ is twice differentiable.) In general, from the arguments of [18, 17], there will be other singular points s' on the boundary of the domain A_s, denoted ∂A_s. By continuity of the surface [15], the heights of these points will be correctly determined

by the local algorithm.

From A3.1 above, the surface curvature at these points is nonzero, so they are either local maxima, minima, or saddle points of $z(\cdot)$. Actually, it is easy to see that when s is a local minimum, the only possibilites for singular points on the boundary of A_s are saddles and local maxima. If it is possible to identify one of these points s' as a local maxima, then we are in the same situation as before. The local algorithm can again be applied with the height provided as initial datum at the new singular point s', which extends the surface reconstruction over the region $A_{s'} \cup A_s$. The arguments of [18, 17] show that every singular point in the image is connected to every other singular point by a sequence of image curves of steepest descent. Thus, iterating this process will eventually yield $z(\cdot)$ at all local minima and maxima singular points, and a correct surface reconstruction over the entire image domain $\mathcal{G}$.

For the above strategy to work, the crucial requirement is a method for identifying as such the local maxima (or minima) on the boundary of the domain of attraction of the original singular point s. This is nontrivial, since the surface reconstructed by the local algorithm assuming just the one singular point s will in general reconstruct all other singular points as inflection points. This is shown in Figures 1 and 2. The local maxima in Figure 1 have disappeared in Figure 2, which is a reconstruction using the local algorithm based on one singular point. The reason for this is that the algorithm reconstructs a height $z(\cdot)$ given by the representation as the infimum of the cost. Crudely, the infimum of the cost for points further from s will be higher, because there is a larger cost involved in reaching the singular point s. (Recall that a minimal cost path must terminate at s in order to avoid the large cost from $g(\cdot)$ of terminating elsewhere.) Thus the reconstructed height in general will increase further from s, and local minima or maxima will not be correctly reconstructed.

Nevertheless, it is possible to use the local reconstruction based on s to identify the local maximum singular points on the boundary of A_s. Suppose for simplicity that A_s is entirely contained in the interior of the image region $\mathcal{G}$. Then if one follows steepest ascent paths on

the surface upward from s, they must terminate at extremal surface points, corresponding to singular points in ∂A_s. Since the surface is twice differentiable, the curves of steepest ascent or descent on this surface are uniquely determined through each point. Therefore the image curves of steepest descent (or ascent), which are the image projections of the surface curves, are likewise uniquely determined through each image point, and fill out A_s. Since from A3.1 there are only a finite number of singular points, it is easy to show [18] that all but a finite number of these image curves of steepest ascent converge to local maximum points in ∂A_s. The finite number of exceptions converge to saddle points in ∂A_s, exactly one to each saddle point.

The following discussion is purely formal, as our purpose here is simply to suggest a numerical scheme. Each direction of initial motion in the image away from the local minimum s corresponds to a different image curve of steepest ascent. From the above, if we start a particle off in a randomly chosen direction from s and let it proceed along an image curve of steepest ascent, it has essentially a unit probability of ending up at a local maximum point in ∂A_s. Suppose we could compute the probability density $Pr(q)$, $p \in \mathcal{G}$, for this particle to pass through the point q. Then this density is zero outside $\bar{A}_s$, the closure of A_s, and develops a singularity at any local maxima points in ∂A_s. In contrast, there will be no such singularity at saddle points, because only one image curve of steepest ascent converges to each saddle point. Thus, singularities in $Pr(q)$ would give a signature for local maximum points in ∂A_s.

A scheme for computing an approximation to $Pr(q)$ on $\mathcal{G}^h$ can be developed that is similar in spirit to our local reconstruction algorithms. However it is much simpler. Let $V(x,y)$ be the reconstruction obtained by our algorithm using $z(s)$ as the initial boundary datum. Effectively, what we want to do is a Monte Carlo simulation, in which a particle starting from s in a randomly chosen direction, climbs the reconstructed surface $V(x,y)$ along a steepest ascent path. By keeping a tally over multiple trials of where on ∂A_s the particle ends up, we can identify the points at which the particle ends up frequently as maximum points. (Note that our algorithm as implemented is much simpler and more efficient than this!) However, a major

problem in doing this is that the surface is given only on a discrete grid of pixels, making the paths of steepest descent or ascent hard to compute accurately.

We solve this problem by introducing an additional element of randomness, as in our original derivation of the local algorithm. We let the particle move probabilistically at every step, not just on the first step from s, in such a way that on average it will move in the steepest ascent direction. The advantage of doing this is that while the particle itself is always confined to the discrete pixel grid, the average motion of the particle is not, and is a better approximation to the continous steepest ascent path. Morover, a simple and deterministic algorithm for computing $Pr(x, y)$ can then be derived by dynamic programming in the same way as for the local algorithm.

Specifically, we assign the probabilities for the particle motion as follows. At s, let $p(s, s \pm (1,0)) = p(s, s \pm (0,1)) = 1/4$, giving an equal probability of emerging from s in any direction. Here $p(q, q')$ denotes the probability of moving from the point q to the point q'. For a better approximation to motion in a random direction from s, it is also possible to have the particle move to diagonal points $s + (\pm 1, \pm 1)$ with the appropriate probabilities. Next consider points $q \neq s$. For these points, it is first necessary to identify the quadrant in the image within which the direction of steepest ascent lies. This quadrant is characterized by the fact that the heights for the nearest neighbor points of q are largest in this quadrant. Thus let v_x and v_y be the largest values from the sets

$$\{V(q+(1,0)), V(q-(1,0))\} \text{ and } \{V(q+(0,1)), V(q-(0,1))\},$$

respectively. Also, let

$$s_x = \text{ sign}\,(V(q+(1,0)) - V(q-(1,0)))\,,$$

$$s_y = \text{ sign}\,(V(q+(0,1)) - V(q-(0,1)))\,,$$

and

$$d_i \equiv \max\,(0, s_i(v_i - V(q)))\,,$$

for $i = x, y$. The d_i are essentially forward or backward derivatives, depending on which direction of motion gives the steepest ascent. For $0 = d_x = d_y$ define $p(q, q) = 1$. Otherwise, define

$$p(q, q + s_x(1,0)) = \frac{d_x}{d_x + d_y},$$

$$p(q, q + s_y(0,1)) = \frac{d_y}{d_x + d_y},$$

with all other probabilities zero. For example, if $s_x = s_y = 1$, then the average motion of the particle is proportional to

$$(V(q + (1,0)) - V(q), V(q + (0,1)) - V(q)),$$

which is an approximation of the steepest ascent direction as desired.

Let ξ_i denote the position of the particle at the time step i. We are interested in the probability

$$Pr_n(q) \equiv p(q = \xi_i,\ i \in [0, n]),$$

which approximates the probability density mentioned above. Take $Pr_0(q) = \delta_{sq}$, with δ the usual discrete delta function. As before, from the principle of dynamic programming, the accumulated probability $Pr_n(q)$ that a particle has passed through the pixel q at some time step $i \leq n$ can be related to the probability that it has passed through neighboring pixels at the previous time step. A more detailed argument shows that $Pr_n(q)$ can be computed via the recursion

$$\begin{aligned} Pr_{n+1}(q) \;=\; & Pr_n(q + (1,0))\, p(q + (1,0), q) + Pr_n(q - (1,0))\, p(q - (1,0), q) \\ & + Pr_n(q + (0,1))\, p(q + (0,1), q) + Pr_n(q - (0,1))\, p(q - (0,1), q). \end{aligned}$$

Finally, we define $Pr(q) \equiv \lim_{n \to \infty} Pr_n(q)$. The existence of this limit follows from the fact that the particle can not double back on itself, since it is always climbing to sites of larger V. Also, the convergence is stable since we have focused on the accumulated probability Pr_n, which is monotonically nondecreasing with the discrete time n. The apparently more natural alternative would have been simply to consider $P_n(q)$, defined as the probability for the particle

to be exactly at the site q at step n. However, this quantity does not vary monotonically with time.

The structure of the recursion for Pr_n is similar to that of the local reconstruction algorithm itself. Therefore, the number of iterations required for convergence here is expected to be similar to the number of iterations used in the convergence to V. This recursion, together with the initial condition $Pr_o(q) = \delta_{sq}$, is our algorithm for computing $Pr_n(q)$. Note that it is deterministic as promised.

For our experiments, the procedure sketched above was modified in a variety of ways. First, due to the randomness in the path, the particle may possibly escape the domain of attraction A_s, although this is impossible for a particle following a steepest ascent path on the continuous surface from s. This may result in spurious concentrations of probability at singular points outside of $A_s \cup \partial A_s$ and the false identification of these points as local maxima. To avoid this problem, we modify the transition probabilities. Let $m > 0$ be a distance less than half of the distance between any two singular points (or singular points clusters) in the image. Then the probabilities are modified so that the particle is prevented from exiting the set $\{q : \|q - s'\| < m\}$ once it has entered it, where s' is any singular point different from s. Since the steepest ascent paths converge on the local maxima points in ∂A_s, the particle has a high probability of approaching these points closely. This modification therefore reduces sharply the strength of $Pr(q)$ at singular points outside ∂A_s.

In the algorithm as implemented, there are 3 parameters whose value can be adjusted in the computation of $Pr(q)$. The first is the number of iterations, which we always set equal to the number required for convergence of the local algorithm. For all our test images, 6 iterations were used. The second parameter sets the threshold such that image points with intensity above this threshold are identified as singular points. This is necessary since, due to image noise, not all singular points have intensity exactly 1. Finally, the third parameter is m as defined above, which sets the size of the "trapping" region for the particle around singular points. This again

was taken to be 5 pixels for all three test images.

Once $Pr(q)$ has been computed, it is necessary to determine the local maximum (or local minimum) singular points using this quantity. This is done by first finding the local maxima of the probability $Pr(q)$ throughout the image. These are points where many image curves of steepest ascent converge, and thus should be close to a local maximum point of the surface. Finally, near each of these points a search is carried out for the highest intensity nearby singular point, which is then returned as a candidate local maximum of the surface.

There are two additional parameters in this segment of the algorithm. One is the threshold below which local maxima of probability are not accepted. Because (due to m) particles are mostly prevented from leaving the domain of attraction, $Pr(q)$ is typically small or 0 outside this domain. This threshold therefore excludes most singular or other points that are much outside the domain of attraction, as desired. This parameter was set mainly to .02 in our experiments. The second parameter is the search radius near the probability maxima, which in our experiments was set to 6 pixels. The settings of these parameters were largely determined in experiments on one surface. The fact that other surfaces could also be reconstructed with the same parameter settings argues for the robustness of the algorithm, and the lack of sensitivity to these parameters.

The discussion up to now has described the procedure for determining new local maxima singular points assuming a single local minimum singular point, namely s was known. However, the procedure works equally well if the reconstruction utilizes several local minimum singular points, or alternatively local maximum singular points. However, when more than one singular point is used, it is usually necessary to know the relative heights of these points. This iwill have been determined in a previous stage of the algorithm.

In operation, the algorithm works as follows. A single local minimum or local maximum singular point near the image center is first identified. This point is chosen near the image center so that it is more likely to have several other "nearest neighbor" singular points on

the boundary of its domain of attraction. The surface is then reconstructed using the local algorithm for the given singular point, the probability Pr_n computed, and new singular point candidates identified (either as local maxima or minima, depending on the nature of the original singular point). Usually there is more than one of these. The procedure is then repeated: the surface is reconstructed, and Pr_n computed, using all newly identified singular points as seeds. This normally leads to the classification of a still larger number of singular points. If the original singular point was correctly classified to begin with, it should be reidentified as a local minimum (or maximum) candidate at this stage. This whole procedure is repeated a number of times, alternately reconstructing using local maxima and local minima as seeds to the local algorithm. Finally, the procedure is halted after a small number of iterations—3–6 iterations in our experiments. The most computationally intensive part of this algorithm is the original local reconstruction algorithm.

It has been assumed so far that initially at least one singular point s is known to be a local minimum (or maximum). Since the height $z(\cdot)$ is ambiguous up to the addition of an overall constant, the height of this point is not needed to start the procedure above. However, for any singular point s, it is often possible to determine a priori whether it is a local maximum, minimum, or saddle point.

Let $I'(\cdot)$ be an arbitrary "intensity" function in some region around s such that $I'(s) = 1$ and $I'(q) \in (0,1)$, $q \neq s$. The corresponding height function $z'(q)$, given by the control representation of (2), will in general not be continuously differentiable. In the typical case, $\nabla z'(\cdot)$ will have line discontinuities, and the approximate $V(\cdot)$ reconstructed by the local algorithm will approximately reproduce these discontinuities. For $I(\cdot)$ also, our experiments show that in general if $V(\cdot)$ is computed using a singular point s under the incorrect assumption that it is a local minimum (or maximum), then the discrete "derivatives" $V(q+(0,1)) - V(q)$, will display orientation discontinuities. These can be easily detected. We have used the occurrence of these orientation discontinuities both to determine an initial local minimum point, and to check the

assignments of singular points as local minima, maxima, or saddle points, as determined by our iterative procedure above.

This is done as follows. The surface is first reconstructed using one singular point as a seed for the local algorithm. Orientation discontinuities are detected in the image by simply comparing the original image intensity to that computed for the surface reconstructed by the local algorithm. If these differ by greater than some threshold amount, an orientation discontinuity is assigned. However, this is subject to the requirement that the assigned discontinuity not be too near a singular point. This requirement is justified partly by its usefulness. Also these discontinuities, which are essentially due to the crossing of optimal paths in the image, are particularly likely to develop near local maxima or local minima singular points where these optimal paths converge. Then the probability $Pr(q)$ is computed as before, except that now the particle is halted on reaching an orientation discontinuity. The total stopping probability for a particle to terminate on an orientation discontinuity is calculated, and compared to the probability that the particle stops within the trapping region of some singular point. If the original singular point was correctly classified, the ratio of the first of these quantities to the second will be small, otherwise it will be big. To start the algorithm, we search for a singular point in the center of the image which has a relatively unambiguously low value for this ratio.

4 Experiments

We have carried out experiments on three 128×128 synthetic surfaces. These were generated by creating random images in frequency space, masking these with a filter so as to reduce the high frequencies, and then fourier transforming to obtain the surface. In all cases, the light was assumed to be coming from the direction $(0, .3., .95)$. The images were generated from these surfaces using the standard image irradiance equation and discrete derivatives.

The first image is shown in Figure 3, which was generated from the surface shown in Figure 4. The height of this surface varied from 0 to 25 pixel units. Figure 5 displays the singular

points in this image, defined as those points with intensity greater than 0.995. They form a relatively small number of clusters, which can each be treated essentially as a single singular point.

Figure 6 displays the reconstruction obtained for this surface. The local minimum singular point used to initiate this reconstruction is one of the central image points. Four cycles of reconstruction were used; that is, the surface was first reconstructed using a single local minima, then again using the recovered local maxima, again using local minima, and the result of reconstructing a final time using the final recovered local maxima is displayed in the figure.

The reconstruction is qualitatively quite good. Figure 7 illustrates the reconstruction error, i.e. the magnitude of the difference between the original surface height and that of the reconstruction. The reconstruction is very good except near the edges of the image. Discrepancies of this kind near the image edges were also observed in reconstructions using the purely local algorithm [15], and do not necessarily represent a failure of the algorithm. As noted in Section 2, for the version of the local algorithm utilizing local maximum singular points, the surface will only be correctly reconstructed at those points which are connected by a curve of steepest asscent to a local maximum in the image. It is therefore likely that the reconstructed surface will be in error at some points near the image boundary—namely, at those points where the steepest ascent direction is out of the image. On the other hand, these points may well be computed correctly by the local algorithm utilizing local minima. Note that since the lighting is oblique rather than vertical, the steepest ascent direction is defined with respect to the direction of the light source, rather than the z axis. However, it is also true that the surface reconstruction is likely to be completely ambiguous for small regions near the image boundary; this cannot be corrected for by any algorithm [17].

Not all of the singular points were correctly identified in the reconstruction cycle. Apparently, this is not necessary for reasonably reconstructions. The average height error in the large interior region of the image is just 0.5, which is remarkable compared to the total height range

of 25. Figures 8 and 9 illustrate a section of the surface and its reconstruction. This section consists of y pixel values bounded between 20 and 105, and the full range of 128 pixels in the x direction. The qualitative accuracy of reconstruction is extremely good, as expected from the low error observed for this region in Figure 7.

A second image and its singular points are displayed in Figures 10 and 11. The reconstruction, shown in Figure 13, is again qualitatively good except near the image boundaries. This is also clear in the error surface shown in Figure 14, although the effect is less striking than for the previous surface. Only three cycles, again starting from a local minimum singular point, were enough to give this reconstruction. The surface was also reconstructed starting from a different local minimum singular point, with comparably good results. Thus the starting singular point does not seem to be crucial in producing the correct surface. The average height error in the interior of the image is 1 unit, in comparison to the overall height range for this surface of 44 units. As before, the average accuracy of reconstruction is on the order of 2%. Again, this accuracy is illustrated in Figures 15 and 16 for an interior section of the surface.

We have also studied the effect of adding noise to the image in Figure 10. The noise added had a uniform distribution, with a maximum amplitude of 0.1. Since the maximum image intensity is only $I = 1$, this is a large noise—$\pm 10\%$!. The noisy image is shown in Figure 18, and the reconstruction in Figure 17. Although there are large errors in some parts of the image, the reconstuction is still remarkably good over most of the image. The error in the height is displayed in Figure 19, where saturated white represents a height error of 3. The error is less than 3 units over most of the image—specifically, that area which is not saturated white. In the region of the image with $127 > x, y > 40$, the mean height error is just 1.6. This represents a surprising immunity to the very large image noise.

Finally, a third surface is shown in Figure 20, and its image in Figure 21. This surface is more complicated than the others, both at middle and high frequencies, and the image contains more singular points. Thus more cycles of reconstruction were required—namely, 6. In the error

image in Figure 23, saturated white represents a height error of 5 units, compared to the total height range for this surface of 44. In the central region of the image, the height error averaged 2.6 units. The performance of the algorithm in this case was less good than for the previous surfaces, but still reasonable. This is probably due to the greater complexity of the surfaces, and the correspondingly larger number of singular points.

In the experiments so far, the algorithm did not in fact automatically select the starting singular point. However, it could have. For the case of the second surface, we carried out experiments using the technique described at the end of Section 3. The results are tabulated in Table 1. It is clear that the correct assignment of a local minimum or maximum singular point can be verified or falsified by the size of the ratio between the probability accumulated on singular points, versus that accumulated on orientation discontinuities. Note how much the ratio changes for points that are quite close to each other in the image, depending only on whether their assignment is correct.

X	76	76	72	38	38	40	19	19
Y	78	78	70	46	36	102	92	92
Treated as	MAX	MIN	MIN	MIN	MIN	MIN	MAX	MIN
Actually	MIN	MIN	SAD	MIN	SAD	SAD	MAX	MAX
Prob. on Discontinuities	.77	.28	.63	.4	.6	.5	.2	.45
Prob. on Singular pts.	1.48	13.3	3.5	5	1.8	1.9	5.4	.49

Table 1. Reconstruction using singular points correctly and incorrectly identified as local minima or local maxima. Correct identification corresponds to a high probability on singular points, while incorrect identification gives a relatively higher probability for terminating on an orientation discontinuity.

1. M. Bichsel, A. P. Pentland, , "A Simple Algorithm for Shape from Shading," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Champaign, Illinois, pp. 459-465, June 1992.

2. M. J. Brooks and B. K. P. Horn, "Shape and Source from Shading," *Proceedings of the International Joint Conference on Artificial Intelligence*, Los Angeles, CA, pp. 932–936, August 18–23, 1985.

3. A. R. Bruss, "The Eikonal Equation: Some Results Applicable to Computer Vision," *Journal of Mathematical Physics*, Vol. 23, No. 5, pp. 890-896, May 1982.

4. P. Dupuis and J. Oliensis, *in preparation.*

5. P. Dupuis and J. Oliensis, "Direct Method for Reconstructing Shape from Shading," in *IEEE Computer Vision and Pattern Recognition*, Champaign, Illinois, June 1992, pp. 453-458.

6. R. T. Frankot and R. Chellappa, "A Method for Enforcing Integrability in Shape from Shading Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 10, No. 4, pp. 439–451, July 1988.

7. B. K. P. Horn, "Height and Gradient From Shading," *International Journal of Computer Vision* Vol. 5 No. 1, pp. 37–75, 1990.

8. B. K. P. Horn, "Image Intensity Understanding," *Artificial Intelligence*, Vol. 8, No. 2, pp. 201–231, April 1977.

9. B. K. P. Horn, "Obtaining Shape from Shading Information," Chapter 4 in *The Psychology of Computer Vision*, P. H. Winston (ed.), McGraw–Hill: New York, 1975, pp. 115–155.

10. B. K. P. Horn and M. J. Brooks, "The Variational Approach to Shape from Shading," *Computer Vision, Graphics, and Image Processing*, Vol. 33, pp. 174–208, 1986.

11. B. K. P. Horn and M.J. Brooks (eds.) *Shape from Shading.* MIT Press: Cambridge, MA, 1989.

12. K. Ikeuchi and B. K. P. Horn, "Numerical Shape from Shading and Occluding Boundaries," *Artificial Intelligence*, Vol. 17, Nos. 1–3, pp. 141–184, August 1981.

13. H. J. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, Springer–Verlag: New York, 1992.

14. Y. G. Leclerc and A. F. Bobick, "The Direct Computation of Height from Shading," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Lahaina, Maui, Hawaii, 1991, pp. 552-558.

15. J. Oliensis and P. Dupuis, "Direct Method for Reconstructing Shape from Shading," in *Physics–Based Vision: Principles and Practice, Shape Inference Volume*, L. Wolff, S. Shafer, G. Healey, editors, Jones and Bartlett, Boston, June 1992, pp. 17–28.

16. J. Oliensis and Paul Dupuis, "Direct method for reconstructing shape from shading," *Proc. SPIE Conf. 1570 on Geometric Methods in Computer Vision*, San Diego, California, July 1991, pp. 116-128.

17. J. Oliensis, "Shape from Shading as a Partially Well–Constrained Problem," *Computer Vision, Graphics, and Image Processing: Image Understanding*, Vol. 54, No. 2, September 1991, pp. 163-183.

18. J. Oliensis, "Uniqueness in Shape From Shading," *The International Journal of Computer Vision*, Vol. 6 no. 2, pp. 75-104, 1991.

19. A. P. Pentland, "Linear Shape from Shading," *International Journal of Computer Vision* Vol. 4, pp. 153–162, 1990.

20. A. P. Pentland, "Local Shading Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, No. 2, pp. 170–187, March 1984.

21. E. Rouy, A. Tourin, "A Viscosity Solutions Approach To Shape–From–Shading," June 1991, to appear in *SIAM J. on Numerical Analysis.*

22. E. Rouy, A. Tourin, "A Viscosity Solutions Approach To Shape–From–Shading," unpublished report.

23. B. V. H. Saxberg, "An Application of Dynamical Systems Theory to Shape From Shading," in *Proc. DARPA Image Understanding Workshop*, Palo Alto, CA, pp. 1089–1104, May 1989.

24. B. V. H. Saxberg, "A Modern Differential Geometric Approach to Shape from Shading," MIT Artificial Intelligence Laboratory, TR 1117, 1989.

25. R. Szeliski, "Fast Shape from Shading," *Computer Vision, Graphics, and Image Processing: Image Understanding*, Vol. 53, pp. 129-153, 1991.

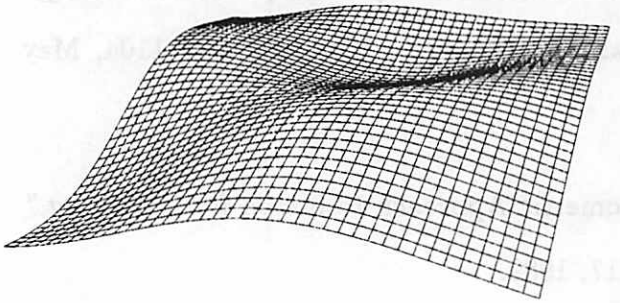

Fig. 1 Original Surface.

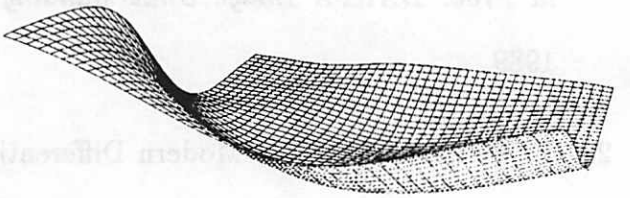

Fig. 2 Reconstruction with one singular point.

Fig. 3 First image.

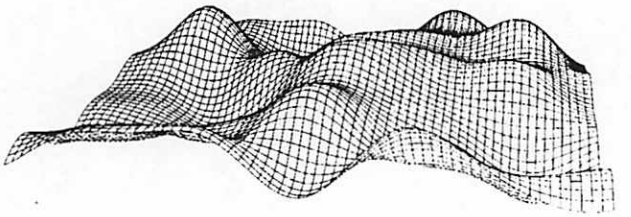

Fig. 4 First Surface.

Fig. 5 Singular points in first image.

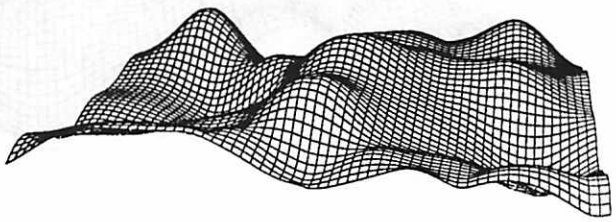

Fig. 6 First surface reconstruction.

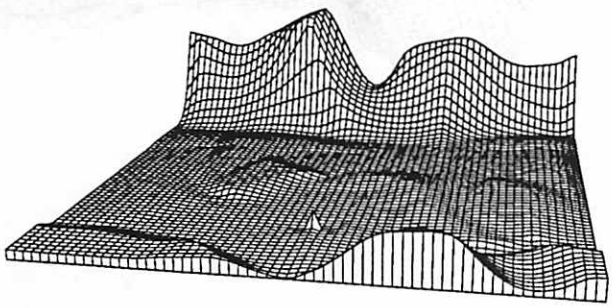

Fig. 7 Difference surface for first image.

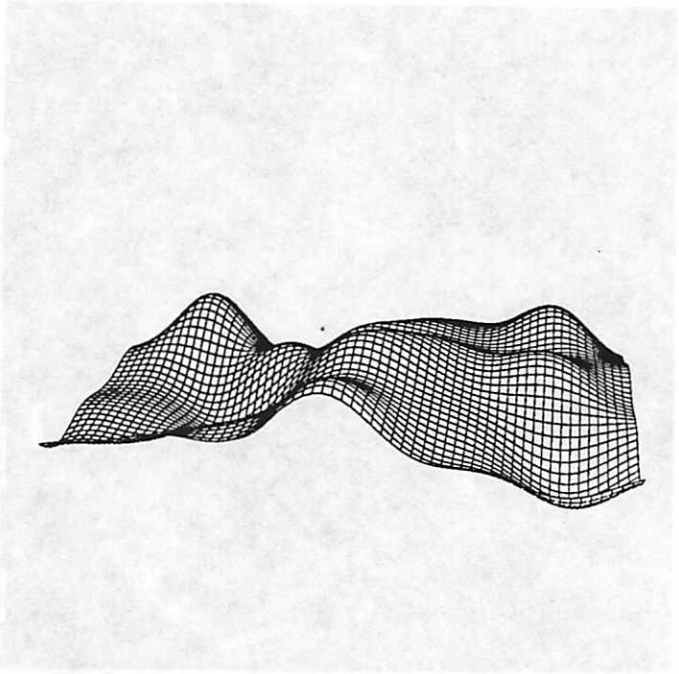

Fig. 8 First surface section.

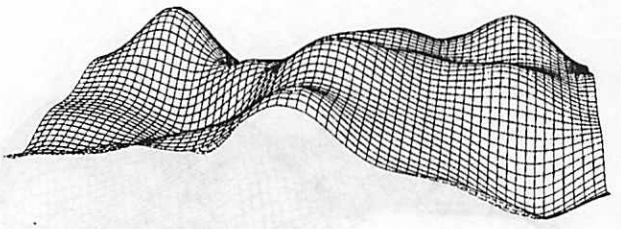

Fig. 9 Reconstructed first surface section.

Fig. 10 Second image.

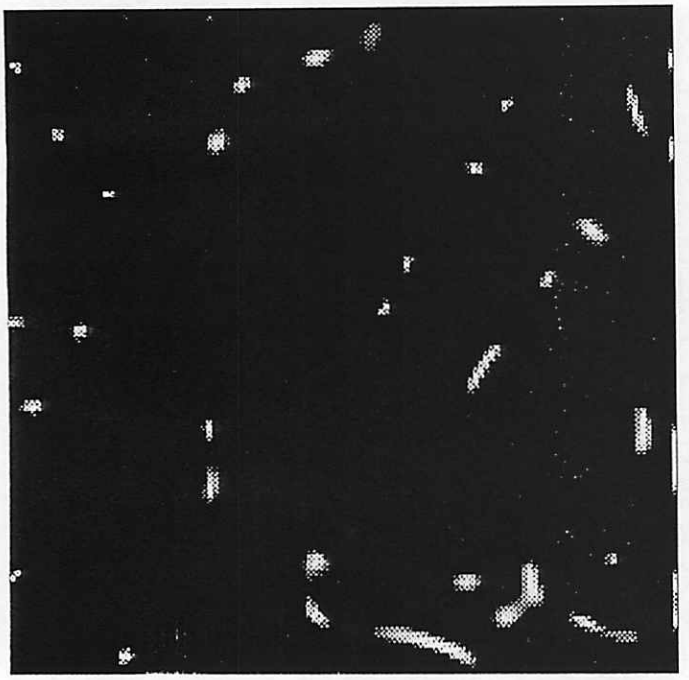

Fig. 11 Singular points for second image.

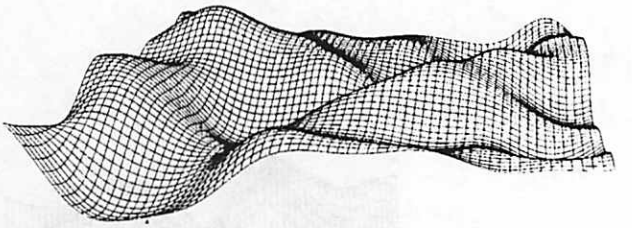

Fig. 12 Second surface.

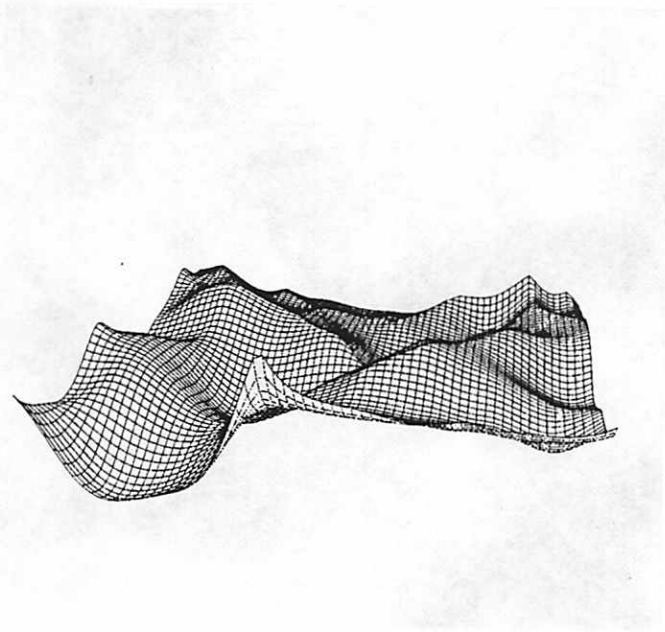

Fig. 13 Reconstructed second surface.

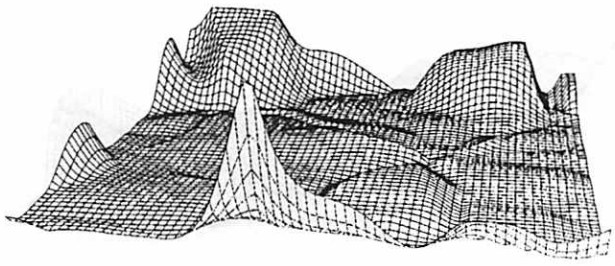

Fig. 14 Difference for second surface. point.

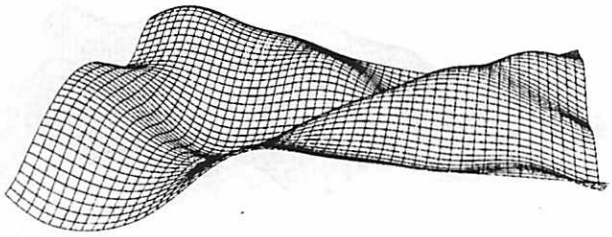

Fig. 15 Second surface section.

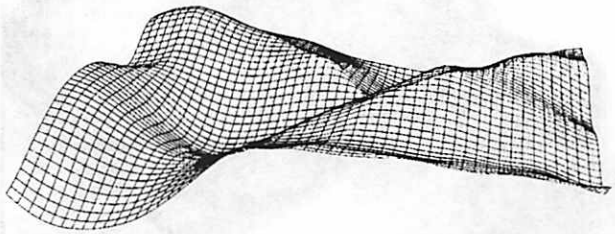

Fig. 16 Reconstruction section.

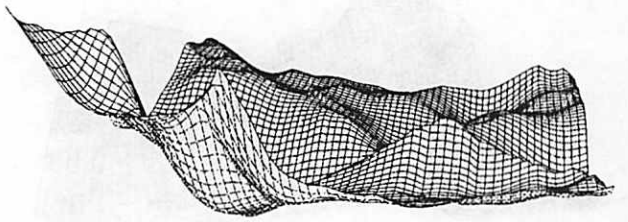

Fig. 17 Reconstructed surface for noisy image.

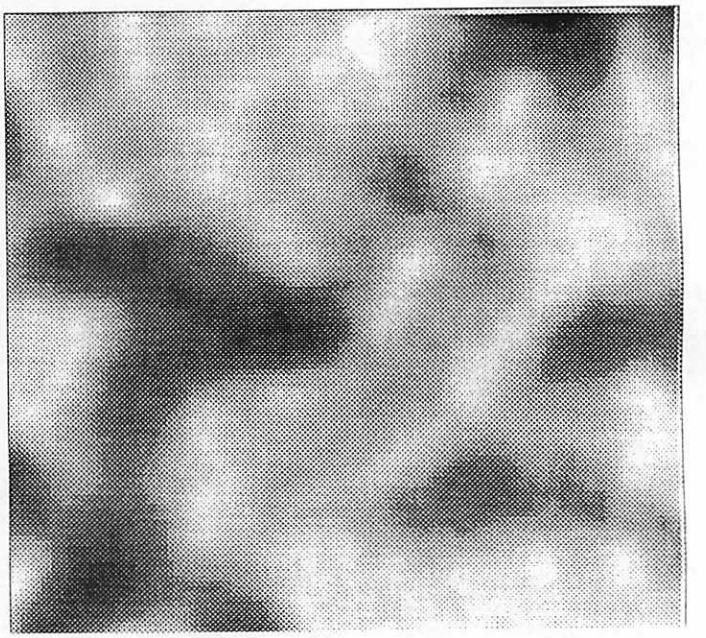

Fig. 18 Noisy second image.

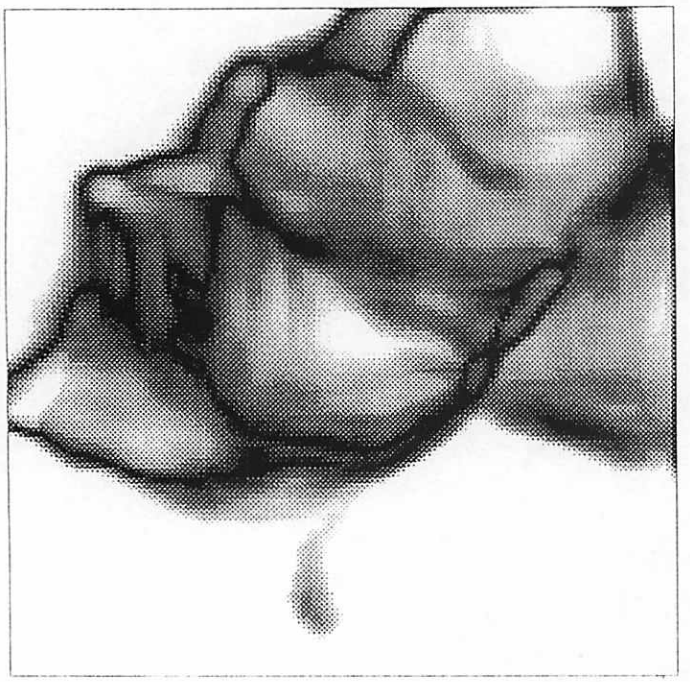

Fig. 19 Error in noisy reconstruction.

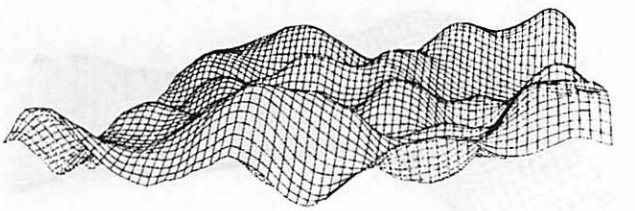

Fig. 20 Third Surface.

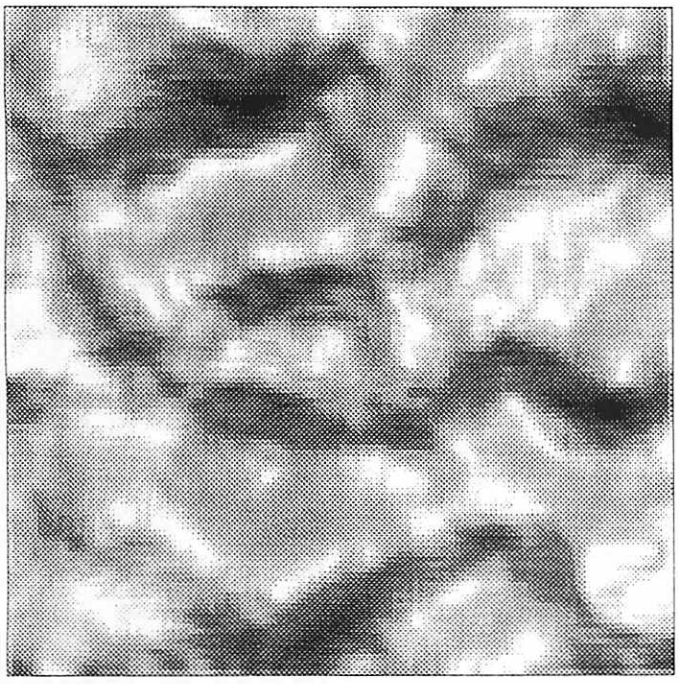

Fig. 21 Third image.

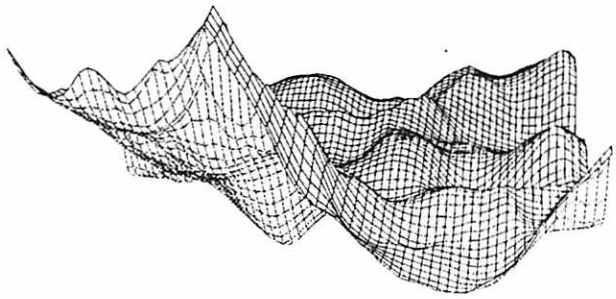

Fig. 22 Third surface reconstruction.

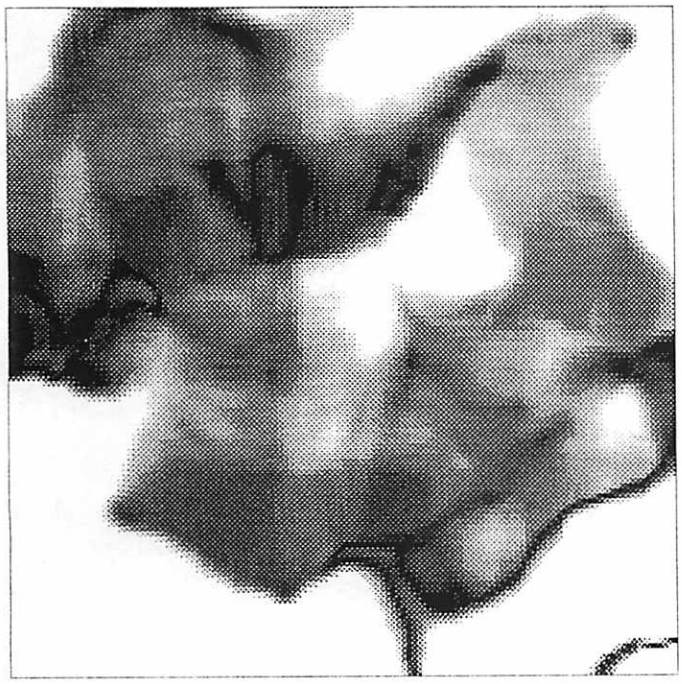

Fig. 23 Difference surface.