International Journal of Computer Vision, 5:3, 271–301 (1990)

Bayesian Modeling of Uncertainty in Low-Level Vision

RICHARD SZELISKI
Digital Equipment Corporation, Cambridge Research Lab, One Kendall Square, Bldg. 700, Cambridge, MA 02139

Abstract
The need for error modeling, multisensor fusion, and robust algorithms is becoming increasingly recognized in computer vision. Bayesian modeling is a powerful, practical, and general framework for meeting these requirements. This article develops a Bayesian model for describing and manipulating the dense fields, such as depth maps, associated with low-level computer vision. Our model consists of three components: a prior model, a sensor model, and a posterior model. The prior model captures a priori information about the structure of the field. We construct this model using the smoothness constraints from regularization to define a Markov Random Field. The sensor model describes the behavior and noise characteristics of our measurement system. We develop a number of sensor models for both sparse and dense measurements. The posterior model combines the information from the prior and sensor models using Bayes' rule. We show how to compute optimal estimates from the posterior model and also how to compute the uncertainty (variance) in these estimates. To demonstrate the utility of our Bayesian framework, we present three examples of its application to real vision problems. The first application is the on-line extraction of depth from motion. Using a two-dimensional generalization of the Kalman filter, we develop an incremental algorithm that provides a dense on-line estimate of depth whose accuracy improves over time. In the second application, we use a Bayesian model to determine observer motion from sparse depth (range) measurements. In the third application, we use the Bayesian interpretation of regularization to choose the optimal smoothing parameter for interpolation. The uncertainty modeling techniques that we develop, and the utility of these techniques in various applications, support our claim that Bayesian modeling is a powerful and practical framework for low-level vision.

1 Introduction

Over the last decade, many low-level vision algorithms have been devised for extracting depth from intensity images. The output of such algorithms usually contains no indication of the uncertainty associated with the scene reconstruction. The need for such error modeling, however, is becoming increasingly recognized. This modeling is necessary because of the noise inherent in real sensors, and the desire to optimally integrate information from different sensors or viewpoints.

This article presents a Bayesian model that captures the uncertainty associated with low-level vision processes and is applicable to two-dimensional dense fields such as depth maps. Our model consists of three components. The prior model describes the world or its properties that we are trying to estimate. The sensor model describes how any one instance of this world is related to the data (such as images) that we acquire. The posterior model, which is obtained by combining the prior and sensor models using Bayes' rule, describes our current estimate of the world given the data we have observed.

The main thesis of this paper is that Bayesian modeling of low-level vision is both feasible and useful. In the paper, we develop a Bayesian framework for low-level vision problems such as surface interpolation and depth-from-motion. To show that our approach is feasible, we build computationally tractable Bayesian models using Markov random fields. To show that these models are useful, we develop representations and algorithms that yield significant improvements in capability and accuracy over existing regularization- and energy-based low-level vision algorithms.

The computationally tractable versions of Bayesian models we use involve estimating first- and second-order statistics. The first-order statistics of a probability distribution are simply its *mean* values. Many low-level vision algorithms already perform this estimation, either by explicitly using Bayesian models, or by using

optimization to find the best estimate (the best and mean estimates often coincide in these problems). The second-order statistics, which encode the *uncertainty* or the *variance* in these estimates, are used much less frequently. The application of uncertainty estimation in computer vision and robotics has previously been limited to systems that have a small number of parameters, such as the position and orientation of a mobile robot or the location of discrete image features. In this article, we extend uncertainty modeling to dense correlated fields such as the depth or optical flow maps commonly used in low-level computer vision.

1.1 Modeling Uncertainty in Low-Level Vision

Low-level visual processing is often characterized as the extraction of *intrinsic images* [Barrow and Tenenbaum 1978] such as depth, orientation, or reflectance from the visual input (figure 1). A characteristic of these images is that they usually represent *dense fields*, that is, the information is available at all points in the two-dimensional visual field. This dense, retinotopic information is then segmented and grouped into coherent surfaces, parts, and objects by later stages of processing.

Intrinsic images form a useful intermediate representation and facilitate the task of higher-level processing. Intrinsic characteristics such as depth or reflectance are more useful than raw intensities for scene understanding or object recognition since they are closer to the true physical characteristics of the scene. This intermediate representation also provides a framework for integrating information from multiple low-level vision modules such as stereo, shading, occluding contours, motion, and texture, and for integrating information over time.

Much of the processing that occurs in these early stages of vision deals with the solution of *inverse problems* [Horn 1977]. The physics of image formation confound many different phenomena such as lighting, surface reflectance, and surface geometry. Low-level visual processing attempts to recover some or all of these features from the intensity image by making assumptions about the world being viewed. For example, when solving the surface interpolation problem—the determination

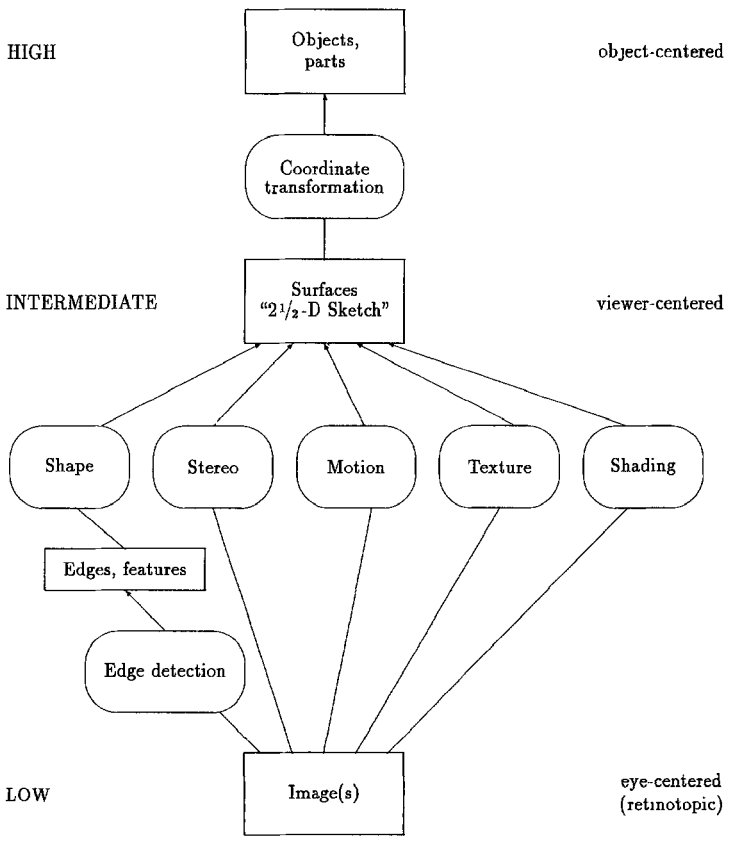

Fig. 1. Visual processing hierarchy.

of a dense depth map from a sparse set of depth values—the assumption is made that surfaces vary smoothly in depth (except at object or part boundaries).

The inverse problems arising in low-level vision are generally *ill-posed* [Poggio et al. 1985], because the data insufficiently constrains the desired solution. One approach to overcoming this problem, called *regularization* [Tikhonov and Arsenin 1977], imposes weak smoothness constraints on the solution in the form of stabilizers. Another approach, *Bayesian modeling* [Geman and Geman 1984], assumes a prior statistical distribution for the data being estimated and models the image formation and sensing phenomena as stochastic or noisy processes. This latter approach is the one we examine in this article.[1]

Currently, both regularization and Bayesian modeling techniques are used only to determine a single (optimal) estimate of a particular intrinsic image. Bayesian modeling, however, can also be used to calculate the uncertainty in the estimate. Modeling this uncertainty, which is the main subject of this article, is important for several reasons. First, because the sensors used in vision applications are inherently noisy, the resulting estimates are themselves uncertain, and we must quantify this uncertainty if we are to develop robust higher-level algorithms. Second, the prior models used in low-level vision can be uncertain (because of unknown parameters) or inaccurate (due to oversimplification). Third, the data obtained from the sensors and subsequent calculations may be insufficient to uniquely constrain the solution, or it may require integration with other measurements. The Bayesian approach allows us to handle both of these cases, namely underdetermined and overdetermined systems, in a single unified framework. Lastly, uncertainty modeling is essential for dynamic estimation algorithms whose accuracy improves over time. These dynamic algorithms can then be applied to problems such as the on-line extraction of depth from motion [Matthies et al. 1989].

The Bayesian approach to low-level vision has other advantages as well. We can use this approach to estimate statistically optimal values for the global parameters that control the behavior of our algorithms (section 8.3). We can generate sample elements from our prior distributions to determine if they are consistent with our intuitions about the visual world or the class of objects being modeled (section 4.1). We can also integrate probabilistic descriptions of our sensors—which can often be obtained by calibration or analysis—into our estimation algorithms (section 5).

The specific Bayesian models we develop in this article are based on Markov Random Fields (MRFs), which describe complex probability distributions over dense fields in terms of local interactions. Markov random fields have several features that make them attractive for low-level vision. The MRF description is very compact, requiring the specification of only a few local interaction terms, while the resulting correlations can have infinite range. MRFs can also easily encode (and estimate) the location of discontinuities in the visual surface. Markov random fields are amenable to massively parallel algorithms for computing the most likely or mean estimates. As we will see in this article, these same algorithms can be used for computing uncertainty estimates. Using such algorithms, we can directly exploit the increasing parallelism that is becoming available in image processing architectures.

1.2 Previous Work

The formulation of low-level vision as a transformation from input images to intermediate representations was first advocated by Barrow and Tenenbaum [1978] and Marr [1978]. Marr [1982] particularly stressed the importance of representations in developing theories about vision. Marr's idea of a "2½-D sketch" was further formalized when Terzopoulos [1988] proposed visible surface representations as a uniform framework for interpolating and integrating sparse depth and orientation information. More recently, Blake and Zisserman [1987] have suggested that discontinuities in the visible surface are the most stable and important features in the intermediate-level description, a view that seems to be echoed by Poggio et al. [1988].

The computational theories used in conjunction with these surface representations were formulated first in terms of variational principles by Grimson [1983] and Terzopoulos [1983], then later formalized using regularization theory [Poggio, Torre, and Koch 1985; Terzopoulos 1988]. Several methods have been proposed for discontinuity detection, including continuation [Terzopoulos 1986b], Markov random fields [Marroquin 1984], weak continuity constraints [Blake and Zisserman 1987], and minimum-length encoding [Leclerc 1989]. Similar energy-based models have also been extended to full three-dimensional surfaces by Terzopoulos et al. [1987].

The common element in these computational theories is the minimization of a global energy function composed

of many local energy components. This minimization is usually implemented using iterative algorithms. The earliest cooperative algorithms were applied to the stereo matching problem [Julesz 1971; Dev 1974; Marr and Poggio 1976]. A different class of iterative algorithms called relaxation labeling [Waltz 1975; Rosenfeld et al. 1976; Hinton 1977] was used to find solutions to symbolic constraint satisfaction problems. The idea of constraint propagation for numerical problems was first suggested by Ikeuchi and Horn [1981], and has been used in many subsequent low-level vision algorithms [Horn and Schunck 1981; Grimson 1983]. Multigrid methods [Terzopoulos 1983], which are based on multiple resolution representations [Rosenfeld 1980], have been used to speed up the convergence of numerical relaxation.

The application of Bayesian modeling to low-level vision has received less attention. Markov random-field models have been used to characterize piecewise constant images [Geman and McClure 1987] or surfaces [Marroquin 1985]. Error models have been developed for stereo matching [Matthies and Shafer 1987] and for more abstract sensors [Durrant-Whyte 1987]. The use of different loss functions to derive alternative optimal posterior estimators has been studied by Marroquin [1985]. In the domain of real-time processing of dynamic data, the Kalman filter has been applied to the tracking of sparse features such as points or lines [Faugeras et al. 1986; Rives et al. 1986], and has recently been extended to dense fields [Matthies et al. 1989]. Here, we apply the Bayesian approach to low-level vision by analyzing the uncertainty inherent in dense estimates and by developing a number of new algorithms based on the Bayesian framework.

Energy-based and Bayesian models have been applied to a variety of low-level vision problems. The one that we examine in detail here is surface interpolation. Additional low-level vision problems include stereo [Barnard and Fischler 1982], motion [Horn and Schunck 1981], and shape from shading [Ikeuchi and Horn 1981]. While these latter applications are not examined in this article, the same uncertainty modeling techniques that are developed here can be applied to these problems.

Surface interpolation is often seen as a post-processing stage that integrates the sparse output of independent low-level vision modules [Marr 1982], although it has recently been used in conjunction with other algorithms such as stereo [Hoff and Ahuja 1986; Chen and Boult 1988]. Surface interpolation was first studied in the context of stereo vision [Grimson 1981]. An interpolation algorithm based on variational principles was developed by Grimson [1983], then extended to use multiresolution computation by Terzopoulos [1983], and finally reformulated using regularization [Poggio, Torre, and Koch 1985]. Recent research has focused on using Markov random fields [Marroquin 1984], continuation methods [Terzopoulos 1986b], and weak continuity constraints [Blake and Zisserman 1987] to detect discontinuities in the visible surface.

1.3 Overview

This article first reviews the representations used with low-level vision and the Bayesian modeling framework. Next, this framework is instantiated by developing prior models, sensor models, and posterior models, and extended to a dynamic environment using the Kalman filter. Finally, we describe a number of applications where our Bayesian modeling framework has been used. A more detailed overview of the article follows.

In section 2, we introduce visible surface representations as a framework for sensor integration and dynamic vision. A discrete implementation of this representation is presented, and the cooperative solution of regularized problems is explained. In section 3, we introduce Bayesian models and Markov random fields and explain the role of prior models, sensor models, and posterior models in the context of low-level vision. In section 4, we use the stabilizer from regularization to define our probabilistic prior model. Using Fourier analysis, we show that the prior model is correlated Gaussian noise with a fractal power spectrum. In section 5, we apply probabilistic modeling to the sensors used in low-level vision. We develop the equivalence between a point sensor with Gaussian noise and a simple spring constraint, and show how to extend this model to other uncertainty distributions and to dense measurements such as correlation-based optical flow.

After developing the prior and sensor models, we examine the characteristics of the posterior model in section 6. We show how the uncertainty in the posterior estimate can be calculated from the energy function of the system, and we devise two new algorithms to perform this computation. In section 7, we extend our model to temporally varying data by developing a two-dimensional generalization of the Kalman filter that can be used to track a time-evolving surface.

In section 8, we describe a number of problems to which our Bayesian framework has been applied. The

first application is an incremental depth-from-motion algorithm [Matthies et al. 1989]. This algorithm computes a dense estimate of scene depth that improves as more images are acquired. The second application computes the observer or object motion, when given two or more sets of sparse depth measurements, without using any correspondence between the sensed points [Szeliski 1988]. The third application estimates the optimal amount of smoothing to be used in regularization by maximizing the likelihood of the data points that were observed [Szeliski 1989]. To conclude, we discuss the relative merits of mechanical and probabilistic models (section 9) and present some open questions and areas of future research (section 10).

2 Representations for Low-Level Vision

Representations play a central role in the study of any visual processing system [Marr 1982]. The representations and algorithms that describe a visual process are a particular instantiation of a general computational theory, and are constrained by the hardware that is available for their implementation. Representations make certain types of information explicit, while requiring that other information be computed when needed. For example, a depth map and an orientation map may represent the same visible surface, but complex computations may be required to convert from one representation to the other. The choice of representation becomes crucial when the information being represented is uncertain [McDermott 1980].

In this section, we examine representations suitable for modeling visible surfaces. In the context of the hierarchy of visual processing (figure 1), these representations are at the interface between the low and intermediate stages of vision. We first review retinotopic visible surface representations and discuss their use. We then examine the use of regularization, finite element analysis, and relaxation for specifying and solving low-level vision problems. This examination is followed by a review of multiresolution algorithms and discontinuity detection.

2.1 Visible Surface Representations

The visible surface representations tht we use here are related to Marr's *2½-dimensional (2½-D) sketch* [Marr 1978] and Barrow and Tenenbaum's *intrinsic images* [Barrow and Tenenbaum 1978]. The 2½-D sketch is a retinotopic map that encodes local surface orientation and distance to the viewer as well as discontinuities in the orientation and distance maps. Intrinsic images represent scene characteristics such as distance, orientation, reflectance, and illumination in multiple retinotopic maps.

Visible surface representations can be used to integrate the output of different vision modules or different sensors (figure 1). They can also be used to integrate information from different viewpoints and to fill in or smooth out information obtained from low-level processes. Two possible techniques for performing this integration and interpolation are regularization and Markov random-field modeling.

Before proceeding with a description of these techniques, we should briefly discuss the question: "Are visible surface representations necessary?" The early work on intermediate representations [Barrow and Tenenbaum 1978; Marr 1978] was motivated by a disappointment with feature-based approaches to vision and a desire to incorporate computational models of image formation. Some of the recent research in computer vision, however, has suggested that image features can be grouped and matched directly to a model [Lowe 1985] or to a more general parts description [Pentland 1986].

Psychophysical studies and recent computational modeling suggest that both models of visual processing (hierarchical and direct) are present in human vision and can be used in computer vision applications. The human ability to obtain depth perception from random-dot stereograms [Julesz 1971] strongly suggests an independent stereo-vision module that produces an intermediate depth map. Studies in neurophysiology show the existence of multiple visual maps in the cortex [Van Essen and Maunsell 1983]. These multiple maps may be the structure used by intermediate-level processes involving visual attention and preattentive grouping. In this article, we will concentrate on the formation and representation of intermediate-level maps and ignore the problems associated with higher levels of visual processing.

2.2 Regularization

Intensity images and visible surface representations define the input and output representations for low-level vision processes. To complete the description of a low-level vision module, we must define the algorithm that maps between these two representations. A number of general techniques have been proposed for this task, including constraint propagation [Ikeuchi and Horn

1981], variational principles [Grimson 1983], and regularization [Poggio, Torre, and Koch 1985]. Here, we will use regularization since it subsumes most of the previous methods and provides a general framework for many low-level vision problems.

The inverse problems arising in low-level vision are generally *ill-posed* [Poggio, Torre, and Koch 1985], that is, the data insufficiently constrains the desired solution. Regularization is a mathematical technique, used to solve ill-posed problems, that imposes weak smoothness constraints on possible solutions [Tikhonov and Arsenin 1977]. Given a set of data d from which we wish to recover a regularized solution u, we define an energy function $\mathcal{E}_d(u, d)$ that measures the compatibility between the solution and the sampled data. We then add a *stabilizing* function $\mathcal{E}_p(u)$ that embodies the desired smoothness constraint, and we find the solution u^* that minimizes the total energy

$$\mathcal{E}(u) = (1 - \lambda)\mathcal{E}_d(u, d) + \lambda\mathcal{E}_p(u) \qquad (1)$$

The regularization parameter λ controls the amount of smoothing performed. In general, the data term d and the solution u can be vectors, discrete fields (two-dimensional arrays of data such as images or depth maps), or analytic functions (in which case, the energies are functionals).

For the surface interpolation problem, the data is usually a sparse set of points $\{d_i\}$, and the desired solution is a two-dimensional function $u(x, y)$. The data compatibility term can be written as a weighted sum of squares

$$\mathcal{E}_d(u, d) = \frac{1}{2}\sum_i c_i[u(x_i, y_i) - d_i]^2 \qquad (2)$$

where the confidence c_i is inversely related to the variance of the measurement d_i, that is, $c_i = \sigma_i^{-2}$. Two examples of possible smoothness functionals (taken from Terzopoulos [1986b]) are the *membrane*

$$\mathcal{E}_p(u) = \frac{1}{2}\iint (u_x^2 + u_y^2)\, dx\, dy, \qquad (3)$$

which is a small deflection approximation of the surface area, and the *thin plate*

$$\mathcal{E}_p(u) = \frac{1}{2}\iint (u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2)\, dx\, dy \qquad (4)$$

which is a small deflection approximation of the surface curvature (note that here the subscripts indicate partial derivatives). These two models can be combined into a single functional

$$\mathcal{E}_p(u) = \frac{1}{2}\iint \rho(x, y)\{[1 - \tau(x, y)][u_x^2 + u_y^2] + \tau(x, y)[u_{xx}^2 + 2u_{xy}^2 + u_{yy}^2]\}\, dx\, dy \qquad (5)$$

where $\rho(x, y)$ is a *rigidity* function, and $\tau(x, y)$ is a *tension* function. The rigidity and tension functions can be used to allow depth ($\rho(x, y) = 0$) and orientation ($\tau(x, y) = 0$) discontinuities. The minimum energy solutions of systems that use the above smoothness constraint are "generalized piecewise continuous splines under tension" [Terzopoulos 1986b].

As an example, consider the nine data points shown in figure 2a. The regularized solution using a continuous thin plate model is shown in figure 2b. We can also manually introduce two depth discontinuities and two orientation discontinuities to obtain the solution shown in figure 2c. These figures show some of the flexibility available with controlled-continuity splines, but do not address the problem of automatic discontinuity detection.

The stabilizer $\mathcal{E}_p(u)$ described by (5) is an example of the more general controlled-continuity constraint

$$\mathcal{E}_p(u) = \frac{1}{2}\sum_{m=0}^{p}\int w_m(\mathbf{x}) \sum_{j_1+\ldots+j_d=m} \frac{m!}{j_1!\ \ldots\ j_d!} \times \left| \frac{\partial^m u(x)}{\partial x_1^{j_1}\ \ldots\ \partial x_d^{j_d}} \right|^2 dx \qquad (6)$$

where $\mathbf{x}$ is the (multidimensional) domain of the function u.

Regularization has been applied to a wide variety of low-level vision problems [Poggio, Torre, and Koch 1985]. In addition to surface interpolation, it has been used for shape from shading [Horn and Brooks 1986], stereo matching [Barnard 1989; Witkin et al. 1987], and optical flow [Anandan 1989]. Problems such as surface interpolation and optical-flow smoothing have a quadratic energy function, and hence have a single energy minimum. Other problems, such as stereo matching, may have many local minima and may require different algorithms for finding the optimum solution [Szeliski 1986; Barnard 1989; Witkin et al. 1987].

2.3 Finite Element Discretization

To find the minimum energy solution on a digital or analog computer, it is necessary to discretize the domain of the surface $u(\mathbf{x})$ using a finite number of *nodal variables*. The usual and most flexible approach is to use finite element analysis [Terzopoulos 1988]. Here, we

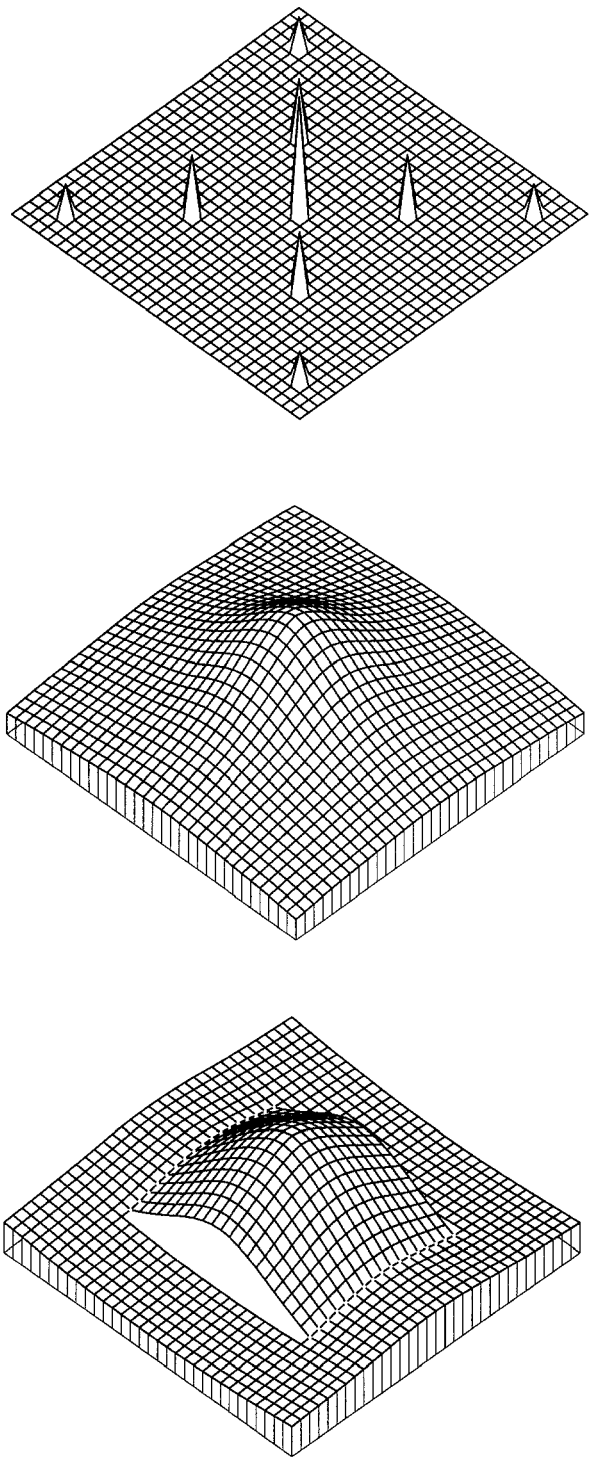

Fig. 2. Sample data and interpolated surface: (a) data points; (b) continuous thin plate solution; (c) thin plate solution with two depth and two orientation discontinuities. The depth discontinuities are shown as missing line segments, while the orientation discontinuities appear as white dots at the nodes.

restrict our attention to rectangular domains on which a rectangular fine-grained mesh has been applied. The topology of this mesh is fixed and does not depend on the location of the data points. It can thus be used for integrating data from various sensors or from various viewpoints. The fine-grained nature of the mesh leads to a natural implementation on a massively parallel array of processors. This kind of massively parallel network is similar to the visual processing architecture of the retina and primary visual cortex.

When we apply finite element analysis to the functionals used in surface interpolation, using a triangular conforming element for the membrane and a nonconforming rectangular element for the thin plate [Terzopoulos 1988], we obtain the energy equations

$$E_{\mathrm{p}}(\mathbf{u}) = \frac{1}{2} \sum_{(i,j)} [(u_{i+1,j} - u_{i,j})^2 + (u_{i,j+1} - u_{i,j})^2] \tag{7}$$

for the membrane (the subscripts indicate spatial position) and

$$\begin{aligned} E_{\mathrm{p}}(\mathbf{u}) = \frac{1}{2} h^{-2} \sum_{(i,j)} [&(u_{i+1,j} - 2u_{i,j} + u_{i-1,j})^2 \\ &+ 2(u_{i+1,j+1} - u_{i,j+1} - u_{i+1,j} + u_{i,j})^2 \\ &+ (u_{i,j+1} - 2u_{i,j} + u_{i,j-1})^2] \end{aligned} \tag{8}$$

for the thin plate, where $h = |\Delta x| = |\Delta y|$ is the size of the mesh (isotropic in x and y). These equations hold at the interior of the surface. Near border points or discontinuities some of the energy terms are dropped or replaced by lower continuity terms [Szeliski 1989; Terzopoulos 1988]. The equation for the data-compatibility energy is simply

$$E_{\mathrm{d}}(\mathbf{u}, \mathbf{d}) = \frac{1}{2} \sum_{(i,j)} c_{i,j}(u_{i,j} - d_{i,j})^2 \tag{9}$$

with $c_{i,j} = 0$ at points where there is no input data.

We can concatenate all the nodal variables $\{u_{i,j}\}$ into a single vector $\mathbf{u}$, and write the prior energy model as one quadratic form

$$E_{\mathrm{p}}(\mathbf{u}) = \frac{1}{2} \mathbf{u}^T \mathbf{A}_{\mathrm{p}} \mathbf{u} \tag{10}$$

This quadratic form is valid for any controlled-continuity stabilizer, though the coefficients will vary according to the structure of the discontinuities. The stiffness matrix $\mathbf{A}_{\mathrm{p}}$ is typically very sparse, but it is not tightly banded because of the two-dimensional structure of the

field. The rows of $\mathbf{A}_p$ are fields of the same dimensionality and extent as the discretized field $\mathbf{u}$ and can be described in terms of computational molecules [Terzopoulos 1988]. For the membrane and thin plate, typical molecules are

$$\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix} \text{ and } h^{-2} \begin{bmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{bmatrix}$$

For the data-compatibility model we can write

$$E_d(\mathbf{u}, \mathbf{d}) = \frac{1}{2}(\mathbf{u} - \mathbf{d})^T \mathbf{A}_d (\mathbf{u} - \mathbf{d}) \tag{11}$$

where $\mathbf{A}_d$ is usually diagonal (for uncorrelated sensor noise) and may contain zeros along the diagonal. The resulting overall energy function $E(\mathbf{u})$ is quadratic in $\mathbf{u}$,

$$E(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathbf{A}\mathbf{u} - \mathbf{u}^T\mathbf{b} + c \tag{12}$$

with

$$\mathbf{A} = \mathbf{A}_p + \mathbf{A}_d \quad \text{and} \quad \mathbf{b} = \mathbf{A}_d\mathbf{d} \tag{13}$$

The energy function has a minimum at $\mathbf{u}^*$, the solution to the linear system of algebraic equations

$$\mathbf{A}\mathbf{u} = \mathbf{b} \tag{14}$$

It can thus be rewritten as

$$E(\mathbf{u}) = \frac{1}{2}(\mathbf{u} - \mathbf{u}^*)^T\mathbf{A}(\mathbf{u} - \mathbf{u}^*) + k \tag{15}$$

2.4 Relaxation

Once the parameters of the energy function have been determined, we can calculate the minimum energy solution $\mathbf{u}^*$ using relaxation. This approach has two advantages over direct methods such as Gaussian elimination or triangular decomposition. First, direct methods do not preserve the sparseness of the $\mathbf{A}$ matrix and thus require more than just a small amount of storage per node. Second, relaxation methods can be implemented on massively parallel, locally connected arrays of processors (or even on analog networks [Koch et al. 1986]). A number of relaxation methods such as Jacobi, Gauss-Seidel, successive overrelaxation (SOR), and conjugate gradient have been used for visible surface interpolation [Terzopoulos 1988; Blake and Zisserman 1987; Choi 1987].

The simplest algorithm to implement is Gauss-Seidel relaxation, where nodes are updated one at a time. This method converges faster than the parallel Jacobi method and can easily be converted to a stochastic version known as the Gibbs Sampler (see section 4.2). At each step, a selected node is set to the value that locally minimizes the energy function. For node u_i, this local energy (with all other nodes fixed) is

$$E(u_i) = \frac{1}{2}a_{ii}u_i^2 + \left(\sum_{j \in N_i} a_{ij}u_j - b_i\right)u_i + k \tag{16}$$

where the subscripts i and j are actually two-element vectors that index the image position, and N_i is the neighborhood of i (the indexes of nonzero entries in row i of matrix $\mathbf{A}$). The node value that minimizes this energy is therefore

$$u_i^+ = a_{ii}^{-1}\left(b_i - \sum_{j \in N_i} a_{ij}u_j\right) \tag{17}$$

Note that it is possible to use a parallel version of Gauss-Seidel relaxation so long as nodes that are dependent (have a nonzero a_{ij} entry) are not updated simultaneously. This parallel version can be implemented on an array of processors for greater computational speed.

2.5 Multiresolution Representations and Algorithms

Unfortunately, the straightforward application of Gauss-Seidel relaxation to surface interpolation usually results in an extremely slow convergence rate (see [Terzopoulos 1983; Szeliski 1989, 1990b] for examples). This slow convergence may not be a problem in a dynamic system (section 7) where iteration can proceed in parallel with data acquisition and the system can converge to a good solution over time.[2] However, for one-shot interpolation problems, the convergence speed may be critical.

To accelerate the convergence of relaxation algorithms, we can use a number of multiresolution algorithms. These algorithms, which operate on pyramid image structures, have proved to be very useful for efficiently solving many image-processing tasks [Rosenfeld 1984]. A popular multiresolution algorithm for visible surface interpolation is multigrid relaxation, which was first applied to this problem by Terzopoulos [1983]. Multigrid algorithms are based on the observation that local iterative methods are good at reducing the high-frequency components of the interpolation error, but are poor at reducing the low-frequency components [Hackbush 1985; Briggs 1987]. By solving related problems

on coarser grids, this low-frequency error can be reduced more quickly. To develop a multigrid algorithm, several components must be specified: the method used to derive the energy equations at the coarser levels from the fine-level equations; a *restriction* operation that maps a solution at a fine level to a coarser grid; a *prolongation* operation that maps from the coarse to the fine level; and a *coordination scheme* that specifies the number of iterations at each level and the sequence of prolongations and restrictions.

Conventional multigrid algorithms keep a full copy of the current depth estimate at each level. An alternative to this absolute multiresolution representation is a *relative* representation [Szeliski 1989], where each level encodes details pertinent to its own scale, and the *sum* of all the interpolated levels provides the overall depth map estimate. The relative representation is thus similar to a band-pass image pyramid [Burt and Adelson 1983; Crowley and Stern 1982], while the absolute representation is similar to a low-pass pyramid. The relative multiresolution representation offers several possible advantages over the usual absolute representation. Fully parallel relaxation can be used with this representation, and yields a multiscale decomposition of the visible surface. Discontinuities can be assigned to just one level, thus permitting a better description of the scene. The use of relative representations can also increase the descriptive power of a method when uncertainty is being modeled [McDermott 1980]. Unfortunately, designing a set of spline energies for each level that decompose the surface into a reasonable multiresolution description while maintaining the faithfulness to the original energy function is a difficult task. A solution based on Bayesian modeling is presented in [Szeliski 1989]. Additional examples of relative representations can also be found in [Szeliski and Terzopoulos 1989b].

Perhaps the most promising fast relaxation algorithm for visual surface interpolation is a multiresolution extension to conjugate gradient descent [Szeliski 1990b]. In this approach, the usual nodal basis set $\mathbf{u}$ is replaced by a hierarchical basis set $\mathbf{v}$ [Yserentant 1986]. Because certain elements of the hierarchical basis set have larger support than the nodal basis elements, the relaxation algorithm converges more quickly (this also manifests itself as a lowered condition number [Yserentant 1986; Szeliski 1990b]). The hierarchical basis conjugate gradient approach allows us to minimize exactly the same energy as the original one derived from the finest finite element grid. It does this by using the pyramid only to smooth the residual inside the conjugate gradient computation. As a result, this new approach is applicable to nonlinear problems such as shape from shading [Szeliski 1990a].

Multiresolution algorithms are an essential component of relaxation-based low-level vision algorithms and the application of Bayesian modeling to these problems. For reasons of brevity, however, we cannot present them here in more detail than what we have sketched above. The reader is referred to [Szeliski 1989] for a more detailed exposition of these algorithms.

2.6 Discontinuities

Representing and localizing discontinuities is an important component of surface interpolation and other low-level vision processes. The detection of intensity discontinuities (*edge detection*) has a long history, dating back to the earliest days of the computer vision field [Roberts 1965; Hueckel 1971; Marr and Hildreth 1980; Canny 1986]. The estimation of depth and orientation discontinuities in parallel with surface interpolation was first studied by Terzopoulos [1986b] using continuation methods, which gradually introduce discontinuities at locations of high curvature. Markov random fields have been used in conjunction with stochastic optimization by Geman and Geman [1984] and Marroquin [1984]. A deterministic approximation to these stochastic algorithms that uses analog "neural nets" was studied by Koch et al. [1986] and more recently by Geiger and Girosi [1989]. Weak continuity constraints, which are similar to Markov random-field descriptions, have been used in the graduated nonconvexity (GNC) algorithm developed by Blake and Zisserman [1987]. The use of intensity edges for constraining the location of depth discontinuities has been studied by Gamble and Poggio [1987].

The accurate localization of depth discontinuities is an important element of visible surface estimation. Without discontinuities, regularization-based methods tend to over-smooth the data, and the accuracy of the reconstruction is reduced. Discontinuity detection can also be combined with surface segmentation [Leclerc 1987], which is an important first step in higher-level analysis. It has even been recently suggested that discontinuities in the visible surface are more important than the depth values themselves [Blake and Zisserman 1987; Poggio et al. 1988]. In this article, we ignore the problem of discontinuity detection. While this problem fits in well with the Bayesian framework [Geman and

Geman 1984; Marroquin 1984], it adds an extra level of complexity to both the implementation and exposition of the interpolation algorithms. We consider the automatic detection and localization of discontinuities to be an important extension that should be added to the work described here.

3 Bayesian Models and Markov Random Fields

In the early days of computer vision, Bayesian modeling was a popular technique for formulating estimation and pattern classification problems [Duda and Hart 1973]. This probabilistic approach fell into disuse as the focus of computer vision research shifted to understanding the physics of image formation and the solution of inverse problems. Bayesian modeling has had a recent resurgence, due in part to the increased sophistication available from Markov random-field models, and due to a realization of the importance of sensor and error modeling. In this section, we briefly review the general Bayesian modeling framework. This is followed by an introduction to Markov random fields and their implementation. We then discuss the utility of probabilistic models in later stages of vision and preview the use of Bayesian modeling in the remainder of the article.

3.1 Bayesian Models

A Bayesian model is a statistical description of an estimation problem that consists of two separate components. The first component, the *prior model*, $p(\mathbf{u})$, is a probabilistic description of the world or its properties before any sensed data is collected. The second component, the *sensor model*, $p(\mathbf{d}|\mathbf{u})$, is a description of the noisy or stochastic processes that relate the original (unknown) state $\mathbf{u}$ to the sampled input image or sensor values $\mathbf{d}$. These two probabilistic models can be combined to obtain a *posterior model*, $p(\mathbf{u}|\mathbf{d})$, which is a probabilistic description of the current estimate of $\mathbf{u}$ given the data $\mathbf{d}$. To compute this posterior model we use Bayes' rule

$$p(\mathbf{u}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{u})\,p(\mathbf{u})}{p(\mathbf{d})} \tag{18}$$

where

$$p(\mathbf{d}) = \sum_{\mathbf{u}} p(\mathbf{d}|\mathbf{u})$$

In its usual application [Geman and Geman 1984], Bayesian modeling is used to find the *maximum a posteriori* (MAP) estimate, that is, the value of $\mathbf{u}$ that maximizes the conditional probability $p(\mathbf{u}|\mathbf{d})$. In the more general case (section 6.1), the optimal estimator $\mathbf{u}^*$ can be the solution that minimizes the expected value of a loss function $L(\mathbf{u}, \mathbf{u}^*)$ with respect to this conditional probability. As we will show in section 3.3, additional useful information (such as the uncertainty in our estimates) can be extracted from the posterior distribution.

To use the Bayesian framework in conjunction with visible surface representations, we must somehow encode the smoothness inherent in these fields. We can do this by using the prior model to describe the correlation between adjacent pixels. A simple method for modeling such correlation is presented next.

3.2 Markov Random Fields

A Markov random field is a probability distribution defined over a discrete field where the probability of a particular variable u_i depends only on a small number of its neighbors,

$$p(u_i|\mathbf{u}) = p(u_i|\{u_j\}), \qquad j \in N_i \tag{19}$$

We can use MRFs to model the correlated structure of dense fields or the smoothness inherent in visible surfaces.

The conditional probabilities $p(u_i|\mathbf{u})$ can be used to generate a prior model $p(\mathbf{u})$. However, calculating $p(\mathbf{u})$ such that all of the marginal distributions are correct is in general a difficult problem. Fortunately, there exists a simple—though indirect—way of specifying a probability distribution for which the conditional probabilities are Markovian. As shown by Geman and Geman [1984], we can use a Gibbs (or Boltzmann) distribution of the form

$$p(\mathbf{u}) = \frac{1}{Z_p} \exp(-E_p(\mathbf{u})/T_p) \tag{20}$$

where T_p is the *temperature* of the model and Z_p is the *partition function*

$$Z_p = \sum_{\mathbf{u}} \exp(-E_p(\mathbf{u})/T_p) \tag{21}$$

The energy function $E_p(\mathbf{u})$ can be written as a sum of local clique energies

$$E_p(\mathbf{u}) = \sum_{c \in C} E_c(\mathbf{u})$$

where each clique energy $E_c(\mathbf{u})$ depends only on a few neighboring points. Thus, to build up our conditional probabilities, we use a linear summation of simple energy terms. These local energies (or cost functions) can be though of as a set of *weak constraints* [Hinton 1977] that penalize unlikely configurations of our prior model.

Computing the probability of any configuration $\mathbf{u}$ using (20) is straightforward, but may be prohibitively expensive due to the exponential complexity of the partition function. For most applications, however, this computation is not necessary. If we wish to generate a random sample from the distribution (20), we can use an algorithm called the *Gibbs sampler* [Geman and Geman 1984]. This iterative algorithm successively updates each state variable u_i by randomly picking a value from the local Gibbs distribution

$$p(u_i|\mathbf{u}) = \frac{1}{Z_i} \exp(-E_p(u_i|\mathbf{u})/T_p) \tag{22}$$

where

$$Z_i = \sum_{u_i} \exp(-E_p(u_i|\mathbf{u})/T_p)$$

This random updating rule is guaranteed to converge (in the ensemble) to a representative sample from the Gibbs distribution. To speed up this convergence, simulated annealing [Metropolis et al. 1953; Kirkpatrick et al. 1983; Hinton and Sejnowski 1983] can be used. The stochastic multigrid techniques discussed in section 4.2 can also be used to to speed up convergence.

The measurement model can usually be written as another Gibbs distribution

$$p(\mathbf{d}|\mathbf{u}) = \frac{1}{Z_d} \exp(-E_d(\mathbf{u}, \mathbf{d})) \tag{23}$$

with

$$E_d(\mathbf{u}, \mathbf{d}) = \sum_i E_d^i(u_i, d_i)$$

For example, the distribution for white Gaussian noise has this form, with $E_d^i = \frac{1}{2}(u_i - d_i)^2/\sigma_i^2$.

We are now in a position to derive the posterior distribution $p(\mathbf{u}|\mathbf{d})$ using Bayes' rule. From equations (18), (20), and (23) we have

$$p(\mathbf{u}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{u})p(\mathbf{u})}{p(\mathbf{u})} = \frac{1}{Z} \exp(-E(\mathbf{u})) \tag{24}$$

where

$$E(\mathbf{u}) = E_p(\mathbf{u})/T_p + E_d(\mathbf{u}, \mathbf{d}) \tag{25}$$

We thus see that the posterior distribution is itself a Markov random field. To compute the MAP estimate, we need only to minimize $E(\mathbf{u})$.

The energy function described by (25) may have many local minima, in which case we must use simulated annealing to perform the optimization. The Gibbs sampler algorithm (using $E(\mathbf{u})$ as the energy function) can be used directly to find the MAP estimate, so long as the system is frozen at the end of the annealing [Geman and Geman 1984]. Alternatively, we could calculate the *maximizer of posterior marginals* [Marroquin 1985], which minimizes the expected number of misclassified pixels.

Comparing (25) to the regularization equation (1) developed in the previous section, we see that regularization is an example of the more general Bayesian approach to optimal estimation. This observation has been made previously in both the numerical analysis literature [Kimeldorf and Wahba 1970] and in the computer vision field [Terzopoulos 1986b; Bertero et al. 1987]. Some newly discovered implications of the relationship will be discussed in section 4.1. The Bayesian interpretation of regularization will also be used in sections 6 and 8 to develop uncertainty estimation and parameter estimation techniques.

Markov random fields have recently been used for image restoration [Geman and Geman 1984; Marroquin 1985], for solving the stereo correspondence problem [Marroquin 1985; Szeliski 1986; Barnard 1989], and for determining discontinuities in visible surfaces [Marroquin 1984]. In this latter application, line processes can be used to represent the discontinuities [Geman and Geman 1984]. The use of line processes to encode and localize discontinuities is currently one of the chief attractions of the MRF approach to low-level vision [Poggio et al. 1988].

Despite their attractive computational properties and their flexibility, Markov random fields have some limitations. They represent distributions with a particularly simple structure, and may be unsuited for modeling more complicated distributions. MRFs are good at modeling fields or surfaces such as terrain maps that have a certain smoothness or coherence but that can have many bumps or wiggles. They are less appropriate for modeling surfaces with more global properties such as piecewise planar surfaces.[3] The direct estimation and modeling of global geometric parameters may be more appropriate in such cases [Durrant-Whyte 1987].

3.3 Using Probabilistic Models

The Bayesian models and Markov random fields that we have introduced in this section have previously been used to obtain single optimal estimates. We show here that additional useful information can be extracted from the posterior distribution, and that a probabilistic development of prior and sensor models can yield new insights into the solution of low-level vision problems.

A simple way to make better use of a posterior distribution is to calculate higher-order statistics (such as variance) in order to quantify the uncertainty in our estimates. The variance of each point can be calculated independently using

$$\text{var}\,(u_i) = \sigma_i^2 = \int (u_i - u_i^*)^2\, p(\mathbf{u}|\mathbf{d})\, d\mathbf{u} \qquad (26)$$

The full covariance matrix of the field $\mathbf{u}$ can also be calculated using

$$\text{cov}\,(\mathbf{u}) = \Sigma_{\mathbf{u}} = \int (\mathbf{u} - \mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)^T p(\mathbf{u}|\mathbf{d})\, d\mathbf{u} \qquad (27)$$

but this information may be too voluminous to store for reasonably sized fields. Higher-order statistics could also be estimated, but these are even more voluminous and hard to compute. In many cases, the distributions that we deal with will be multivariate Gaussians, so that the first- and second-order statistics completely capture the information about the distribution.

Maintaining a probabilistic description of our current estimate is particularly useful in the context of dynamic vision. In such a system, new information is continually being acquired due to either observer or scene motion, and estimates are continually being updated. A useful formalism for modeling such a system is the Kalman filter, which we will examine in section 7. The general Bayesian modeling framework presented in this section can be instantiated in many ways, depending on the particular visual task, visual domain, and sensing strategies being studied. In the next three sections, we examine, in turn, prior models, sensor models, and posterior models. The prior models we study are based on Markov random fields and regularization. A variety of sensor models are then developed, including sparse depth sensors and dense flow estimators. Finally, probabilistic posterior models are developed, along with new techniques for estimating posterior uncertainty, estimating regularization parameters, and estimating observer motion.

4 Prior Models

As we have seen in the previous section, prior models play an essential role in the formulation of Bayesian estimators. When applied to low-level vision, prior models encode the smoothness or coherence of the two-dimensional fields being estmated from the image. In this section, we examine the spectral characteristics of our prior models, and develop algorithms for efficiently generating random samples.

The use of Markov random fields for modeling smooth fields was first suggested by Geman and Geman [1984]. In their implementation, they used discrete values for the intensity and an energy function that favored piecewise constant surfaces. They were also the first to use line processes in conjunction with a MRF representation. Subsequent research has used fields with energies resembling the one obtained from discretizing the membrane model [Marroquin 1984]. In section 4.1, we show how this choice of energy function determines the power spectrum of the prior model.

The ability to generate sample elements from our model space is one of the attractions of the probabilistic approach. This capability allows us to determine if these random samples are consistent with our intuitions about the domain we are modeling. To generate these typical samples, we use the Gibbs sampler algorithm described in section 3.2. As we show in section 4.2, the implementation of this algorithm for models such as the membrane and thin plate is particularly simple and only requires adding a controlled amount of Gaussian noise to the usual Gauss-Seidel relaxation algorithm. We examine how multiresolution (coarse-to-fine) stochastic relaxation can help speed up the approach of the Gibbs sampler toward equilibrium.

The prior models that we study are commonly used to describe intrinsic images and can thus be thought of as "intrinsic models" (this term was coined by Gudrun Klinker). In the hierarchy of visual processing (figure 1), intrinsic models span the middle ground between the object models used in high-level vision and the physical models that describe image formation. Object models are normally used to determine the identity and pose (position and orientation) of a three-dimensional object. These models are typically described by a small number of lumped parameters, such as the pose, the relative positions of parts for articulated objects, and perhaps some shape parameters for models such as superquadrics [Pentland 1986]. In certain cases, the

parameters of these models can be determined directly from the image data [Lowe 1985; Pentland 1986]. Intrinsic models, on the other hand, have a large number of distributed parameters, such as the depth value at each node for a surface model. If we are to recover these parameters from the limited data available in the image, we have to specify a prior distribution and thus restrict the space of possible models.

4.1 Regularization and Fractal Priors

In constructing a Markov random field prior model, we must first choose the energy function defining the Gibbs distribution. As we saw in section 3.2, choosing the regularization smoothness constraint as the energy function results in a MAP estimate that is identical to the one obtained from regularization. While this observation has been used as a statistical justification for regularization [Kimeldorf and Wahba 1970], the characteristics of the prior model have not previously been investigated.

One way of analyzing the prior model is to generate some typical random samples using the Gibbs sampler algorithm described in section 3.2 (the exact implementation details are given in the next section). Using the thin plate whose energy is given in equation (4) as our model, we can generate a typical sample from the prior distribution as shown in figure 3. This surface has an interesting rough or bumpy structure that is quite different from the smooth shape that one might expect. A convenient way to characterize this roughness is to compute the spectral characteristics of the surface using Fourier analysis.

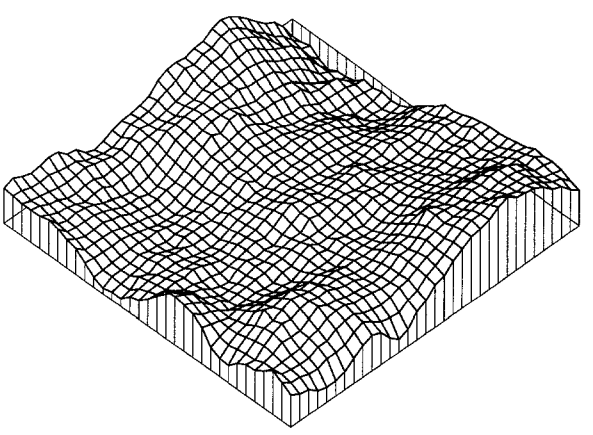

Fig. 3. Typical sample from the thin-plate prior model.

The Fourier transform [Bracewell 1978] of a multidimensional signal $v(\mathbf{x})$ is defined by

$$\mathfrak{F}\{v\} \equiv \int v(\mathbf{x}) \exp\,(2\pi i\mathbf{f}\cdot\mathbf{x})\, d\mathbf{x} = V(\mathbf{f}) \tag{28}$$

and the transform of its partial derivative is given by

$$\mathfrak{F}\left\{\frac{\partial v(\mathbf{x})}{\partial x_j}\right\} = (2\pi i f_j)V(\mathbf{f}) \tag{29}$$

Using Rayleigh's energy theorem

$$\int |v(\mathbf{x})|^2\, d\mathbf{x} = \int |V(\mathbf{f})|^2\, d\mathbf{f} \tag{30}$$

we can rewrite the smoothness functional $\mathcal{E}_p(u)$ in terms of the Fourier transform $U(\mathbf{f}) = \mathfrak{F}\{u\}$ to obtain the new energy function $\mathcal{E}_p'(U)$.

For our smoothness functional, we will use the general form given in (6) with the simplifying assumption that the weighting functions $w_m(\mathbf{x})$ are constant. While this assumption does not strictly apply to the general case of piecewise continuous interpolation, it provides an approximation to the local behavior of the regularized system away from boundaries and discontinuities. Applying (29) and (30) to (6) we obtain

$$\mathcal{E}_p'(U) = \frac{1}{2}\sum_{m=0}^{p}\int w_m \sum_{j_1+\ldots+j_d=m}\frac{m!}{j_1!\,\ldots\,j_d!}$$
$$|(2\pi i f_1)^{j_1}\,\ldots\,(2\pi i f_d)^{j_d}U(\mathbf{f})|^2\, d\mathbf{f}$$

or

$$\mathcal{E}_p'(U) = \frac{1}{2}\int |H_p(\mathbf{f})|^2|U(\mathbf{f})|^2\, d\mathbf{f} \tag{31}$$

where

$$|H_p(\mathbf{f})|^2 = \sum_{m=0}^{p} w_m|2\pi\mathbf{f}|^{2m} \tag{32}$$

For the membrane interpolator, $|H_p(\mathbf{f})|^2 \propto |2\pi\mathbf{f}|^2$, and for the thin plate model, $|H_p(\mathbf{f})|^2 \propto |2\pi\mathbf{f}|^4$.

To derive the spectral characteristics of the prior model, we note that since the Fourier transform is a linear operation, if $u(\mathbf{x})$ has a Gibbs distribution with energy $\mathcal{E}_p(u)$, then $U(\mathbf{f})$ has a Gibbs distribution with energy $\mathcal{E}_p'(U)$.[4] We thus have

$$p(U) \propto \exp\left(-\frac{1}{2}\int |H_p(\mathbf{f})|^2|U(\mathbf{f})|^2\, d\mathbf{f}\right)$$

from which we see that the probability distribution at any frequency $\mathbf{f}$ is

$$p(U(\mathbf{f})) \propto \exp\left(-\frac{1}{2}\,|H_p(\mathbf{f})|^2|U(\mathbf{f})|^2\right)$$

Thus, $U(\mathbf{f})$ is a random Gaussian variable with variance $|H_p(\mathbf{f})|^{-2}$, and the signal $u(\mathbf{x})$ is correlated Gaussian noise with a spectral distribution

$$S_u(\mathbf{f}) = |H_p(\mathbf{f})|^{-2} \tag{33}$$

From this analysis, we conclude that using a regularization-based smoothness constraint is equivalent to using a correlated Gaussian field as the Bayesian prior. The spectral characteristics of this Gaussian field are determined by the choice of stabilizer. For the membrane and the thin-plate models, we have

$$S_{\text{membrane}}(\mathbf{f}) \propto |2\pi\mathbf{f}|^{-2} \tag{34}$$

and

$$S_{\text{thin plate}}(\mathbf{f}) \propto |2\pi\mathbf{f}|^{-4} \tag{35}$$

These equations are interesting because they correspond in form to the spectra of Brownian fractals.

Fractals are a class of mathematical objects that exhibit self-similarity over a range of scales [Mandelbrot 1982]. Fractals have been used to generate intricate geometric designs, to study the statistical properties of coastlines and structured noise, and to generate realistic images of terrain. A stochastic fractal is a random process or a random field that exhibits self-affine statistics over a range of scales. A common way to characterize such a fractal is to note that it follows a power law in its spectral density

$$S_v(f) \propto 1/f^{\beta} \tag{36}$$

This spectral density characterizes a fractal Brownian function $v_H(\mathbf{x})$ with a fractal dimension of $D = E + 1 - H$, where $2H = \beta - E$, and E is the dimension of the Euclidean space [Voss 1985]. A function that satisfies (36) may also be fractional Gaussian noise [Rensink 1986].

Comparing (34) or (35) to (36), we conclude that the smoothness assumptions embedded in standard regularization methods are equivalent to assuming that the underlying processes are fractal [Szeliski 1987]. When regularization techniques are used, it is usual to find the minimum energy solution (figure 2c), which also corresponds to the mean value solution for those cases where the energy functions are quadratic. Therefore, the fractal nature of the process is not evident. A far more *representative* solution can be generated if a random (fractal) sample is taken from this distribution (figure 4). The amount of noise (and hence roughness) that is desirable or appropriate can be derived from the data (see section 8.3).

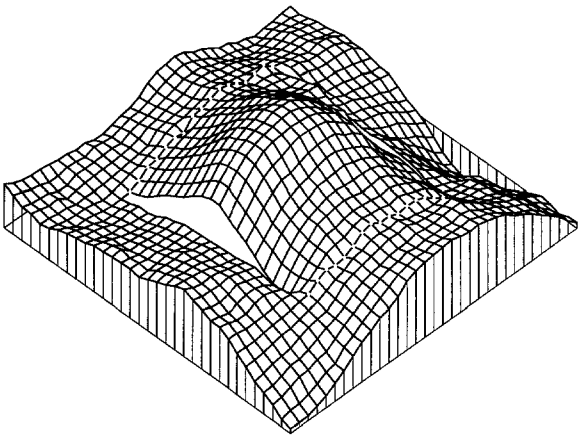

Fig. 4. Fractal (random) solution corresponding to figure 2c.

The fractal nature of the membrane and thin-plate models suggests that we could use priors with in-between (truly fractional) degrees of smoothness. In theory, this is straightforward, since we can specify the prior model to be a Brownian fractal field with an arbitrary β. In practice, implementing the resulting interpolator is difficult. Boult [1986] has implemented such fractional interpolators using reproducing kernel splines. The prior models that we use also need not be isotropic or homogeneous. In general, we can choose a prior model with any arbitrary correlation function and thus model mountain ridges or terrain with locally varying degrees of smoothness. We will explore how to implement such arbitrary prior models in the next section.

4.2 Generating Random Samples

To generate the random samples from either the prior or posterior models, we can use the Gibbs sampler algorithm described in section 3.2. In the Gibbs sampler, each state variable u_i is updated asynchronously (sequentially) by picking a value from the local Gibbs distribution (22). For the surface interpolation problem that we studied in section 2.2, the local energy function (16) is quadratic, with a minimum value u_i^+ given by (17) and a second derivative equal to a_{ii}. The local Gibbs distribution is therefore

$$p(u_i|\mathbf{u}) \propto \exp\left(-\frac{a_{ii}(u_i - u_i^+)^2}{2T_p}\right) \tag{37}$$

which is a Gaussian with mean u_i^+ and variance T_p/a_{ii}. We thus see that the Gibbs sampler is equivalent to the usual Gauss-Seidel relaxation algorithm with the addition

of some locally controlled Gaussian noise at each step [Szeliski 1987]. The temperature parameter T_p controls the amount of roughness in the random sample. In section 8.3, we will present a method for determining the appropriate value of T_p from the sampled data.

As with deterministic relaxation, the above algorithm may converge very slowly toward its equilibrium distribution (the point at which the system exhibits negligible statistical dependence on its starting configuration [Ackley et al. 1985]). To speed up this convergence, we can use a coarse-to-fine technique similar to the one used with deterministic relaxation. We simply generate a random sample using the Gibbs sampler at a coarser level, and then use the interpolated sample as a starting configuration for the finer level. This starting configuration will already be closer to equilibrium than a nonrandom configuration such as the zero state. More importantly, it will contain more of the low-frequency components of the random field than can be obtained by iterating for a long time on the fine level [Szeliski 1989]. Multiresolution stochastic relaxation has also been studied by Barnard [1989] and Konrad and Dubois [1988].

We can use the multiresolution Gibbs sampler algorithm that we have just described to generate constrained fractals with arbitrary discontinuities. Using the same data points as we used for the thin-plate interpolation example (figure 2a) and also the same energy equations, we can apply the Gibbs sampler to the posterior distribution defined by (24). A typical sample generated by this approach is shown in figure 4. While this sample is not truly fractal since it depends on the data points, it is a typical sample from the fractal prior distribution conditioned on the data points that were observed. We can thus shape the fractal by imposing arbitrary depth constraints, orientation constraints [Terzopoulos 1988], depth discontinuities or creases. For a detailed description of our new fractal generation algorithm and a comparison with previous algorithms, see [Szeliski and Terzopoulos 1989a].

5 Sensor Models

Modeling the error inherent in sensors and using these error models to improve performance are becoming increasingly important in computer vision [Matthies and Shafer 1987]. In this section, we present two different sensor models that describe both sparse (symbolic) and dense (iconic) measurements, in order to demonstrate the usefulness of our Bayesian modeling framework. Some additional sensor models are presented in [Szeliski 1989].

5.1 Sparse Data: Spring Models

A noisy depth measurement, such as the three-dimensional location of a feature obtained by stereo triangulation, can be characterized by a three-dimensional probability distribution. Although the shape of this distribution may be quite complex, it can often be approximated by a 3-D Gaussian [Matthies and Shafer 1987]. An advantage of using a Gaussian is that the position vector $\mathbf{p}_i = (x_i, y_i, z_i)$ and the 3×3 covariance matrix $\mathbf{C}_i$ completely specify the distribution.

To determine the interaction between a data point and the visible surface that we compute from our depth measurements, we must first convert this three-dimensional distribution in space into a one-dimensional distribution in depth. Assuming that x and y are the underlying natural coordinates of our visible surface representation, we set $d_i = z_i$ and use the Gaussian probability distribution

$$p(d_i|\mathbf{u}) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(u(x_i, y_i) - d_i)^2}{2\sigma_i^2}\right) \tag{38}$$

If the errors for the various depth measurements are uncorrelated, which is usually the case, this distribution corresponds to the usual data-compatibility equation (2). When our retinotopic representation is not aligned with the sensor reference frame, the position and covariance measurements must first be transformed using simple matrix algebra [Matthies and Shafer 1987]. The effect of these depth constraints on the visible surface is similar to tying the surface to the depth values through springs [Terzopoulos 1988]. The strength of the spring constant is inversely proportional to the variance of the depth measurement.

An alternative way to derive the discrete form of the data constraint is to write the measurement equation

$$d_i = \mathbf{H}_i\mathbf{u} + r_i \tag{39}$$

where $r_i \sim N(0, \sigma_i^2)$. The measurement matrix $\mathbf{H}_i$ encodes how the surface point $u(x_i, y_i)$, which gives rise to the measurement d_i, is obtained from the nodal variables $\mathbf{u}$. The $\mathbf{H}_i$ matrix thus depends on the choice of interpolator. If we choose block (constant) interpolation, the discrete data constraint equation is as before (i.e., we associate the depth constraint with the nearest nodal variable). If we choose bilinear interpolation, we have

$$u(x, y) = h_{00}u_{i,j} + h_{01}u_{i,j+1} + h_{10}u_{i+1,j} + h_{11}u_{i+1,j+1}$$

where $u_{i,j}, \ldots, u_{i+1,j+1}$ are the four nodal variables nearest to (x, y), and $h_{00}, \ldots, h_{11}$ are interpolation constants that depend on x and y. The data constraint equation thus becomes

$$E_d(d, \mathbf{u}) = \frac{1}{2}\sigma^{-2}(d - h_{00}u_{i,j} - h_{01}u_{i,j+1} - h_{10}u_{i+1,j} - h_{11}u_{i+1,j+1})^2 \quad (40)$$

This introduces off-diagonal terms into our data-compatibility matrix $\mathbf{A}_d$, but does not reduce the sparseness of the combined stiffness matrix $\mathbf{A}$ since the prior model matrix $\mathbf{A}_p$ already has such off-diagonal terms.

The probability distribution used to characterize the uncertainty in our depth measurement need not be Gaussian. The advantages of using a Gaussian are that it is characterized completely by its mean and variance values (first- and second-order statistics) and that the resulting constraint energy is quadratic. A Gaussian distribution is appropriate when the error in the measurement is the result of the aggregation of many small random disturbances. Many sensors, however, have a normal operating range characterized by a small σ^2 but also occasionally produce gross errors. A more appropriate model for such a sensor is the *contaminated Gaussian* used by Durrant-Whyte [1987], which has the form

$$p(d_i|\mathbf{u}) = \frac{1-\epsilon}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{(u(x_i, y_i) - d_i)^2}{2\sigma_1^2}\right) + \frac{\epsilon}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{(u(x_i, y_i) - d_i)^2}{2\sigma_2^2}\right) \quad (41)$$

with $\sigma_2^2 \gg \sigma_1^2$ and $0.05 < \epsilon < 0.1$. This model behaves as a sensor with small variance σ_1^2 most of the time, but occasionally generates a measurement with a large variance σ_2. By taking the negative logarithm of the probability density function, we can obtain the constraint energy shown in figure 5. This energy is similar in shape to the weak springs that arise in the weak continuity models of Blake and Zisserman [1987], the ϕ function of Geman and McClure [1987], and the effective potentials of Geiger and Girosi [1989].

Gaussians and contaminated Gaussians are just two of the many possible distributions that can be used to characterize sensors. The advantage of using a Bayesian approach to visual processing is that any sensor model we develop can be incorporated directly into the estimation algorithm. In practice, finding good sensor models involves a tradeoff between the fidelity of the model, the compactness of its representation, and the tractability of its equations.

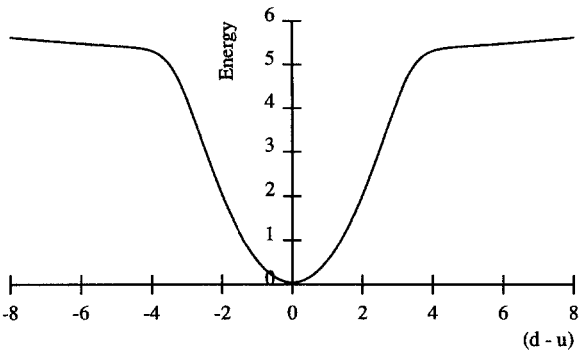

Fig. 5. Constraint energy for contaminated Gaussian.

5.2 Dense Data: Optical Flow

Probabilistic sensor modeling need not be restricted to sparse measurements obtained directly from sensors. We can also apply error analysis to low-level vision algorithms and characterize these algorithms as *virtual sensors* with their own associated error models. For example, such analysis has recently been applied to intensity-based optical-flow estimators. The work was originally done for 1-D (scalar) displacements by Matthies et al. [1989] and later extended to 2-D (vector) displacements by Szeliski [1989]. Both of these analyses show how the uncertainty in the flow measurement at each point can be determined from local measurements already present in the optical-flow algorithm, and how this information can be used in an incremental depth-from-motion algorithm (section 8.1).

The problem of extracting optical flow from a sequence of intensity images has been extensively studied in computer vision. Early approaches used the ratio of the spatial and temporal image derivatives [Horn and Schunk 1981], while more recent approaches have used correlation between images [Anandan 1989] or spatio-temporal filtering [Heeger 1987]. The error analysis we present here uses the simple version of correlation-based matching developed by Anandan [1989], which he calls the *sum of squared differences* (SSD) method. This algorithm integrates the squared intensity difference between two shifted images over a small area to obtain an error measure

$$e_t(\mathbf{d}; \mathbf{x}) = \int w(\boldsymbol{\lambda})[f_t(\mathbf{x} + \mathbf{d} + \boldsymbol{\lambda}) - f_{t-1}(\mathbf{x} + \boldsymbol{\lambda})]^2\, d\boldsymbol{\lambda} \quad (42)$$

where f_t and f_{t-1} are the two intensity images, $\mathbf{x}$ is the image position, $\mathbf{d}$ is the displacement (flow) vector, and $w(\lambda)$ is a windowing function. The SSD measure is computed at each pixel $\mathbf{x}$ for a number of possible flow values $\mathbf{d}$. The resulting error surface $e_t(\mathbf{d}; \mathbf{x})$ is used to determine the best displacement estimate $\hat{\mathbf{d}}$ and the confidence in this estimate.

In analyzing the SSD algorithm, Anandan and Weiss [1985] observed that the shape of the error surface differs depending on whether both, one, or none of the displacement components are known (corresponding to an intensity corner, an edge, or a homogeneous area). They proposed an algorithm for computing the confidence measures based on the principal curvatures and the directions of the principal axes in the vicinity of the error surface minimum. The analysis in [Szeliski 1989, Appendix C] supports this heuristic by demonstrating that the variance of the flow measurement $\hat{\mathbf{d}}$ is $2\sigma_n^2\mathbf{A}^{-1}$, where $\mathbf{A}$ is obtained by fitting a quadratic to the error surface, and σ_n^2 is the variance of the image noise. This analysis can also be used to derive the correlation between adjacent flow estimates and between flow estimates obtained from successive frames.

From these results, we see how the statistical analysis of an optical-flow algorithm can provide an error model for its output. This output can then be treated as a virtual sensor that can be incorporated into a Bayesian estimation framework. This approach permits us to take into account the spatially varying uncertainties that are often inherent in low-level visual processes. A similar analysis can be applied to other low-level vision algorithms with similar benefits.

6 Posterior Models

In the previous two sections, we have developed a prior model for visible surfaces and a number of sensor models for low-level vision algorithms. In this section, we show how the prior and sensor models can be combined using Bayes' rule to obtain a posterior model, and study how to compute optimal estimates of the visible surface from the posterior distribution. We also show how to calculate the uncertainty inherent in a visible surface estimate from this distribution, and discuss why such uncertainty modeling is important.

6.1 MAP Estimation

The probabilistic prior models and sensor models we have developed in this paper are instances of Markov random fields. From the results we obtained in section 3.2, we know that the posterior distribution is also a MRF. This field can be described by a Gibbs distribution with an associated energy

$$E(\mathbf{u}) = E_{\mathrm{p}}(\mathbf{u}) + E_{\mathrm{d}}(\mathbf{u}, \mathbf{d}) \tag{43}$$

where $E_{\mathrm{p}}(\mathbf{u})$ is the energy function associated with the prior model, and $E_{\mathrm{d}}(\mathbf{u}, \mathbf{d})$ is the energy function that describes the sensor model. Computing the *maximum a posteriori* estimate is thus equivalent to minimizing the energy $E(\mathbf{u})$.

Several techniques can be used for performing this minimization, depending on the application. For surface interpolation or optical-flow smoothing, the energy functions $E_{\mathrm{p}}(\mathbf{u})$, $E_{\mathrm{d}}(\mathbf{u}, \mathbf{d})$, and hence $E(\mathbf{u})$ are quadratic. Performing the minimization is thus equivalent to solving a large set of sparse linear equations.[5] As discussed in section 2.2, we can use one of several relaxation techniques to find the minimum energy solution. The advantage of using such iterative techniques over direct solution methods is that they can be implemented on massively parallel architectures.

The MAP estimate, however, is not the only estimate that can be computed from the posterior distribution $p(\mathbf{u}|\mathbf{d})$. As has been observed by Marroquin [1985], any loss function $L(\mathbf{u}, \mathbf{u}^*)$ can be used to define the optimal estimate. Given such a loss function, the optimal estimate $\mathbf{u}^*$ is the one that minimizes the expectation of loss

$$\langle L \rangle = \int L(\mathbf{u}, \mathbf{u}^*) p(\mathbf{u}|\mathbf{d})\, d\mathbf{u}$$

For MAP estimation, the loss function is a negative delta (we win one for guessing the correct estimate, but all other estimates are equally bad). A more sensible loss function for applications such as terrain classification or image restoration is one that counts the number of misclassified pixels. This leads to the maximizer of posterior marginals (MPM) estimator [Marroquin 1985]. For many applications, we can also compute the *minimum mean squared error* (MMSE) estimate.[6]

The advantage of using loss functions to define the optimal estimate is that we can tailor the loss function to our particular application. In addition to allowing the development of task-specific algorithms, this approach allows some top-down influence to be exerted on the low-level process. For example, in a robot navigation application where we want to avoid hitting obstacles, we can use a loss function that penalizes overestimates in distance more than underestimates.

Using different loss functions can increase the power of probabilistic methods over simple energy minimization approaches. Having a single optimal estimate, however, still does not tell us how certain, accurate, or typical such an estimate might be. Ideally, we would like to pass the whole probability distribution on to the next level of processing. In practice, however, we usually have to restrict ourselves to a more parsimonious description.

6.2 *Uncertainty Estimation*

To characterize the uncertainty inherent in the output of a low-level vision algorithm, we can compute the second-order statistics (covariance matrix) of the estimate. This uncertainty measure can be used to integrate new data, to guide search (e.g., to set disparity limits in stereo matching), or to indicate where more sensing is required. For many distributions, second-order statistics do not capture all of the useful information present in the distribution, but they are a good start. In this section, we examine how uncertainty can be derived from the energy function characterizing the posterior distribution, and we present two new algorithms for computing this uncertainty.

When we combine the regularization-based prior models developed in section 4.1 with the simple sensor models developed in section 5.1, we obtain posterior models that are Markov random fields with a quadratic energy function. This energy can be rewritten in the form given in (15), that is,

$$E(\mathbf{u}) = \frac{1}{2}(\mathbf{u} - \mathbf{u}^*)^T\mathbf{A}(\mathbf{u} - \mathbf{u}^*) + k \quad (44)$$

where $\mathbf{u}^*$ is the minimum energy solution. The Gibbs distribution corresponding to this quadratic form is a multivariate Gaussian with mean $\mathbf{u}^*$ and covariance $\mathbf{A}^{-1}$. Thus, to obtain the covariance matrix, we need only invert the information matrix $\mathbf{A}$.[7] One way of doing this is to use the multigrid algorithm presented in section 2.2 to calculate the covariance matrix one row at a time. To obtain a single covariance field, we set $\mathbf{b} = \mathbf{e}_i$ in (13), that is, we set all but one of the data points to 0 without modifying $\mathbf{A}$, and solve as before.

Figures 6a and 6b show two covariance fields, one for the point in the left corner, and one for the point in the middle of the top crease. These fields are identical in shape (but not in magnitude) to the Green's functions (blending functions or shift-*variant* filters) that can be used to solve the interpolation problem. Their shape does not depend on the data values d_i, but only on the smoothing function (prior model) and the data confidence measures c_i. Intuitively, these covariance

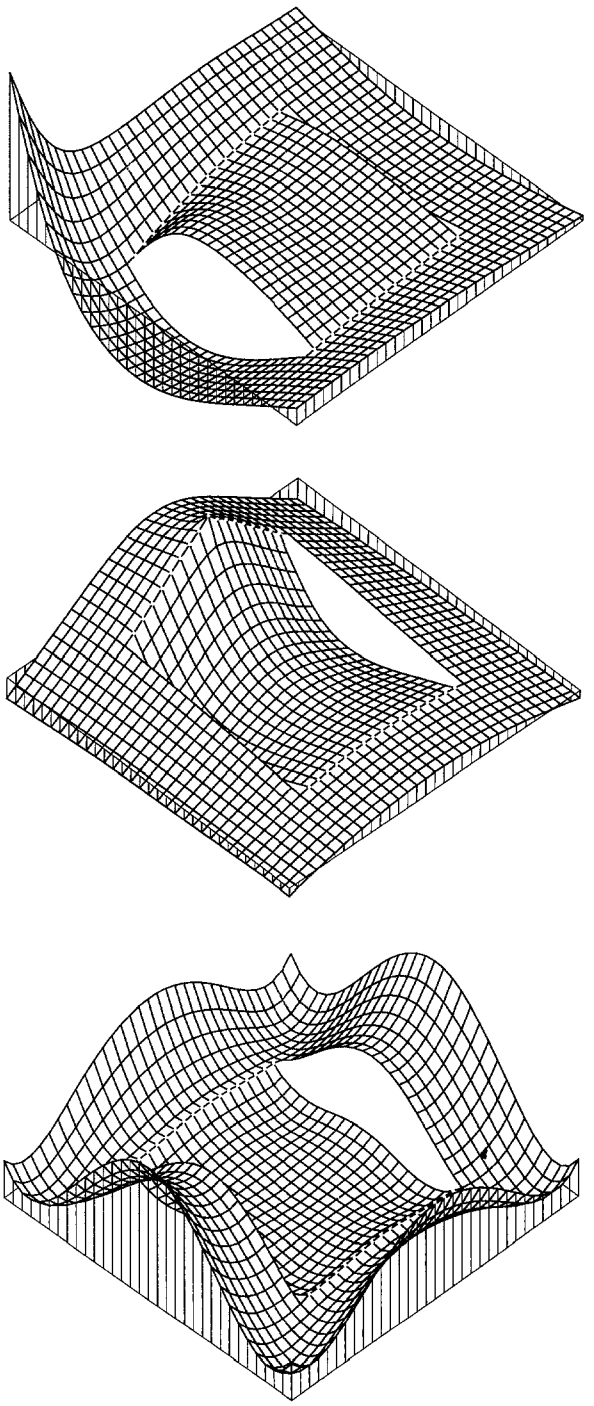

Fig. 6. Sample covariance and variance fields: (a) covariance field at (0, 0); (b) covariance field at (6, 16); (c) variance field.

fields show how the overall surface would wiggle if one particular point was moved up and down. For the special case of isotropic (shift-invariant) smoothing, the Green's function is equivalent to the smoothing filter $h_s(\mathbf{x})$ derived by Poggio et al. [1985].

Storing all of the covariance fields is impractical because of their large size (for a 512×512 image, the covariance matrix has over 10^{10} entries). We can, however, keep only the variance at each point, that is, the diagonal elements of the covariance matrix. These variance values are an estimate of the confidence associated with each point in the regularized solution (e.g., the residual uncertainty in optical flow *after* smoothing). Alternatively, they can be viewed as the amount of fluctuation at a point in the Markov random field (in the ensemble of typical fractal solutions). Figure 6c shows the variance estimate corresponding to the regularized solution of figure 2c (this variance has been magnified for easier interpretation). The variance of the field increases near the edges and discontinuities; this is as expected, if we interpret the variance as the wobble or inverse global stiffness of the thin plate. This variance field gives us a *dense*, distributed error model for the visible surface representation.

Calculating the variance field using the above deterministic algorithm requires resolving the system for each point in the field and is therefore very time consuming. An alternative to this approach is to run the multiresolution Gibbs sampler at a nonzero temperature, and to estimate the desired statistics using a Monte Carlo approach. For example, we can estimate the variance at each point (the diagonal of the covariance matrix) by simply keeping a running total of the depth values and their squares. Unfortunately, the straightforward application of the Gibbs sampler results in estimates that are biased or take extremely long to converge (figure 7a). This is because the Gibbs sampler is a multidimensional version of the Markov random walk. Successive samples are therefore highly correlated, and time averages are ergodic only over very long time scales (even if the system is already at equilibrium). To help decorrelate the signal, we can use successive coarse-to-fine iterations and gather only a few statistics at the fine level each time. Examples of the variance field estimates obtained with such a stochastic algorithm are shown in figures 7b and 7c.

This stochastic estimation technique can also be used with systems that have nonquadratic (and nonconvex) energy functions. In this case, the mean and covariance

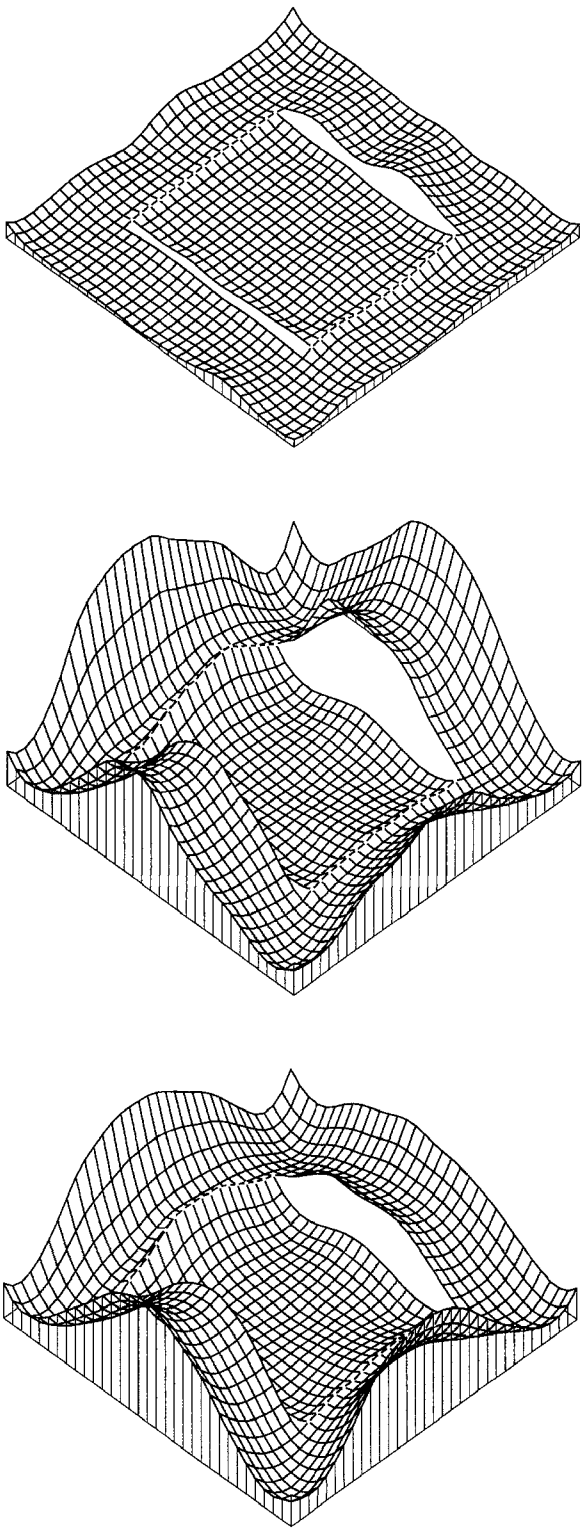

Fig. 7. Stochastic estimates of variance field: (a) single level, 1000 iterations; (b) multiresolution, 100 iterations; (c) multiresolution, 2500 iterations.

are not sufficient to completely characterize the distribution, but they can still be estimated. For stereo matching, once the best match has been found (say by using simulated annealing), it may still be useful to estimate the variance in the depth values. Alternatively, stochastic estimation may be used to provide a whole distribution of possible solutions, perhaps to be disambiguated by a higher-level process.

Once we have calculated the variance field, we can use it to grow a confidence region around the mean or minimum energy solution. This confidence region can be used in applications such as path planning or navigation. For example, we can use a 95% confidence interval if we wish to be "95% certain" of not hitting the surface. We can also look at the size of the confidence region (which is related to the local variance) to decide where additional active sensing may be required. Figure 8 shows a one-dimensional example of such a confidence region built around a cubic spline solution. Note that—just as for the thin plate—the uncertainty grows as we extrapolate away from known data.

Modeling the uncertainty in the visible surface is important if we are using this representation to aggregate data. Uncertainty modeling is also important if we will be matching the visible surface to models from an object database or to other intermediate representations. In section 8.2 we show how including such uncertainty modeling is essential to developing a surface-to-surface matching algorithm that can handle occlusions, limited areas of overlap, and sparse data.

The uncertainty representation scheme developed in this section differs in several important ways from pre-

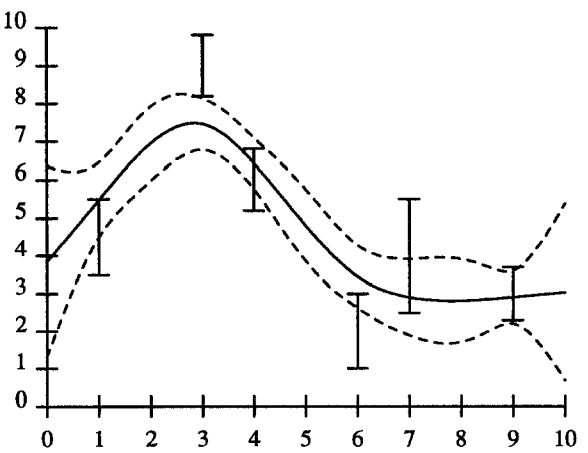

Fig. 8. Cubic spline with confidence interval. The vertical lines are the error bars (1 standard deviation) around the data points, and the dashed lines are the confidence interval for the whole curve.

viously developed representations. The spatial likelihood map developed by Christ [1987] uses spherical coordinates to represent the likelihood of a surface patch at a particular location. This method does not have an explicit smoothness constraint; instead, it uses a local planar model to extrapolate the surface away from known data points. Occupancy maps [Elfes and Matthies 1987; Moravec 1988] use a two- or three-dimensional array of scalar values to indicate the occupancy of different portions of space. This method is well suited to path planning and can represent three-dimensional objects and obstacles. However, the resolution of the method is limited to the grid size. Finally, the approach used by Wahba [1983] to obtain confidence intervals on splines is similar to ours. However, Wahba's method computes a single variance estimate for the whole curve, rather than having a spatially varying variance. This method thus fails to capture some important characteristics of the uncertainty, such as the increase in variance as we extrapolate away from known data.

7 Dynamic Models

The Bayesian models we have described so far allow us to obtain optimal estimates of static visible surfaces, to integrate information from multiple viewpoints, and to analyze the uncertainty in our estimates. Many computer vision applications, however, deal with dynamic environments. This may involve tracking moving objects or updating the model of the environment as the observer moves around. Recent results by Aloimonos et al. [1987] suggest that taking an active role in vision (either through eye or observer movements) greatly simplifies the complexity of certain low-level vision problems.

In this section, we develop a two-dimensional generalization of the Kalman filter suitable for modeling visible surfaces. This framework can be used to construct incremental (on-line) vision algorithms. The advantages of using the incremental approach are that rough depth measurements are available immediately and that the quality of these estimates improves over time as more data is acquired. Incremental processing also has lower storage requirements than batch processing.

7.1 Kalman Filtering

The Kalman filter is a Bayesian estimation technique used to track a stochastic dynamic system being observed with noisy sensors. The filter is based on three separate

probabilistic models. The first model, the *prior model*, describes the knowledge about the system state $\hat{\mathbf{u}}_0$ and its covariance $\mathbf{P}_0$ before the first measurement is taken,

$$\mathbf{u} \sim N(\hat{\mathbf{u}}_0, \mathbf{P}_0) \tag{45}$$

where the notation $\mathbf{x} \sim N(\mathbf{m}, \mathbf{P})$ denotes that $\mathbf{x}$ is a multivariate normal variable with mean $\mathbf{m}$ and covariance $\mathbf{P}$ [Gelb 1974]. As we have seen in section 4, this model allows us to capture the smoothness constraint associated with visible surfaces by setting $\mathbf{P}_0^{-1} = \mathbf{A}_p$. The second model, the *measurement* (or *sensor*) *model*, relates the measurement vector $\mathbf{d}_k$ to the current state through a measurement matrix $\mathbf{H}_k$ and the addition of Gaussian noise $\mathbf{r}_k$,

$$\mathbf{d}_k = \mathbf{H}_k\mathbf{u}_k + \mathbf{r}_k \qquad \mathbf{r}_k \sim N(0, \mathbf{R}_k) \tag{46}$$

When applied to surface estimation, the measurement matrix $\mathbf{H}_k$ is used to convert the dense map $\mathbf{u}_k$ into the sparse measurement $\mathbf{d}_k$ (section 5.1). These two models form the basis for the Bayesian estimation framework we have developed here. The Kalman filter introduces a third model, the *system model*, which describes the evolution of the current state vector $\mathbf{u}_k$ over time. The transition between states is characterized by the known transition matrix $\mathbf{F}_k$ and the addition of Gaussian noise $\mathbf{q}_k$,

$$\mathbf{u}_k = \mathbf{F}_k\mathbf{u}_{k-1} + \mathbf{q}_k \qquad \mathbf{q}_k \sim N(0, \mathbf{Q}_k) \tag{47}$$

In the case of depth-from-motion, the transition matrix $\mathbf{F}_k$ describes the mapping of surface estimates from one coordinate frame to the next as the observer changes position.

The above three models describe the evolution of the state $\mathbf{u}_k$ and its relationship to the measurements $\mathbf{d}_k$. To obtain an optimal estimate $\hat{\mathbf{u}}_k$ of the current state, the Kalman filter operates in two phases. The extrapolation phase predicts the new state given the previous best estimate and updates the covariance matrix associated with the predicted estimate. The correction phase updates the state estimate using the new measurements. It does this by first computing the Kalman filter-gain matrix, and then updating the state estimate by adding the residual between the observed and predicted measurements scaled by the Kalman filter gain [Gelb 1974]. The usual Kalman filter updating equations are not given here since we use the alternative formulation described below.

Kalman filtering is usually applied to systems with a fairly small number of state variables. In the domain of motion sequence analysis, it has previously been used to track edges [Rives et al. 1986; Matthies and Shafer 1987; Baker and Bolles 1989], but has not been used in conjunction with *dense* (iconic) fields such as depth maps. When modeling dense maps, the information matrixes (inverse covariance matrixes) are sparse and banded (because of the nature of the prior information matrix $\mathbf{A}_p$), while the covariance matrixes are not. Bierman [1977] discusses a number of efficient techniques for doing the Kalman filter update that rely on using matrix decomposition or factorization methods. These various decompositions (including the *square root information filter*) do not, however, result in matrixes that are as sparse as the original information matrix. Thus, for applications where the prior and posterior distributions are Markov random fields (of reasonable size), factorization methods are not useful since they require too much storage space ($O(N\sqrt{N})$ or $O(N^2)$ where N is the number of pixels).

For these reasons, we use the information matrix $\mathbf{A}_k \equiv \mathbf{P}_k^{-1}$ and the cumulative weighted data vector $\mathbf{b}_k \equiv \mathbf{A}_k\hat{\mathbf{u}}_k$ as the quantities to be updated. The advantage of this formulation is that the updating equations are particularly simple,

$$\mathbf{A}_k^+ = \mathbf{A}_k^- + \mathbf{H}_k^T\mathbf{R}_k^{-1}\mathbf{H}_k^T \tag{48}$$

and

$$\mathbf{b}_k^+ = \mathbf{b}_k^- + \mathbf{H}_k^T\mathbf{R}_k^{-1}\mathbf{d}_k \tag{49}$$

and they require no matrix inversions. The current estimate $\hat{\mathbf{u}}_k^+$ can be computed at any time by solving

$$\hat{\mathbf{u}}_k^+ = (\mathbf{A}_k^+)^{-1}\mathbf{b}_k^+$$

using multigrid relaxation. In practice, we can use the previous state estimate $\hat{\mathbf{u}}_{k-1}^+$ or the predicted state estimate $\hat{\mathbf{u}}_k^-$ as the starting point for the relaxation and only iterate for a few steps. This may not yield the optimal solution for the given data but, given enough time, the estimate will converge to such an optimal solution. Thus, a tradeoff can be made between the desired accuracy of the data and the amount of computation performed.

The prediction equations for our depth estimation system are somewhat more difficult to implement. This is because the mapping from one depth map $\mathbf{u}_{k-1}$ to the next is not a predetermined linear operation. Instead, the whole depth map is warped according to the local disparity to obtain the new depth map. The exact form of this warping is explained in [Matthies et al. 1989]. For now, we assume that we can compute the transition matrix $\mathbf{F}_{k-1}$ by finding the linear mapping that defines how each point in the new field $\mathbf{u}_k$ is obtained as a

weighted combination of points in the previous field $\mathbf{u}_{k-1}$ (the alternative is to use the *extended Kalman filter* [Gelb 1974] to compute $\mathbf{F}_{k-1}$ from the Jacobian of the state transition function).

Using some simplifying assumptions, which are explained in [Szeliski 1989], we can write the prediction equations as

$$\mathbf{A}_k^- = (1 + \epsilon)^{-1}\mathbf{F}_{k-1}\mathbf{A}_{k-1}^+\mathbf{F}_{k-1}^T$$
$$\mathbf{b}_k^- = (1 + \epsilon)^{-1}\mathbf{F}_{k-1}\mathbf{b}_{k-1}^+$$

In practice, we apply the warping $\mathbf{FAF}^T$ to the data component of the information matrix and leave the prior model component invariant [Szeliski 1989].

The Kalman filtering framework we have developed in this section is specially matched to the structure of the visible surface estimation problem. By using information matrixes rather than covariance matrixes, we can keep the representations sparse and the computations simple. This framework has been used as the basis of the incremental depth-from-motion algorithm reviewed in the next section.

8 Applications

The Bayesian modeling framework developed here can be applied to many low-level vision problems. To demonstrate the feasibility of this approach and its usefulness, we present three algorithms developed using this framework. The first algorithm extracts an on-line estimate of depth from an image sequence taken from a moving camera. The second computes observer motion by matching sparse range data. The third computes the optimal amount of smoothing from the sampled data. All three of these applications use a Bayesian formulation to model the uncertainty in the sensors and to statistically derive optimal algorithms.

8.1 Incremental Iconic Depth-from-Motion

The study of depth-from-motion has long been an active area of research in computer vision. Early work concentrated on extracting the optical-flow field from a pair of images, using either gradient-based [Horn and Schunck 1981] or correlation-based [Anandan 1989] techniques. More recent motion algorithms have attempted to use batch processing of the whole image sequence, either by fitting lines to the spatiotemporal data [Bolles et al. 1987] or by using spatiotemporal filtering [Adelson and Bergen 1985; Heeger 1987].

In this section, we briefly review the incremental algorithm developed by Matthies, Kanade, and Szeliski [1989] that produces a dense on-line estimate of depth from the motion image sequence. As mentioned before, the incremental approach produces rough depth estimates whose quality improves over time. A similar algorithm that extends the analysis to arbitrary motion has recently been developed by Heel [1989].

The algorithm developed by Matthies et al. is based on the Kalman filter we developed in section 7.1. It consists of four main stages (figure 9). The first stage uses correlation to compute an estimate of the displacement vector and its associated covariance (section 5.2). It converts this estimate into a disparity (inverse-depth) measurement using the known camera motion. The second stage integrates this information with the disparity map predicted at the previous time step. The third stage uses regularization-based smoothing to reduce measurement noise and to fill in areas of unknown disparity.

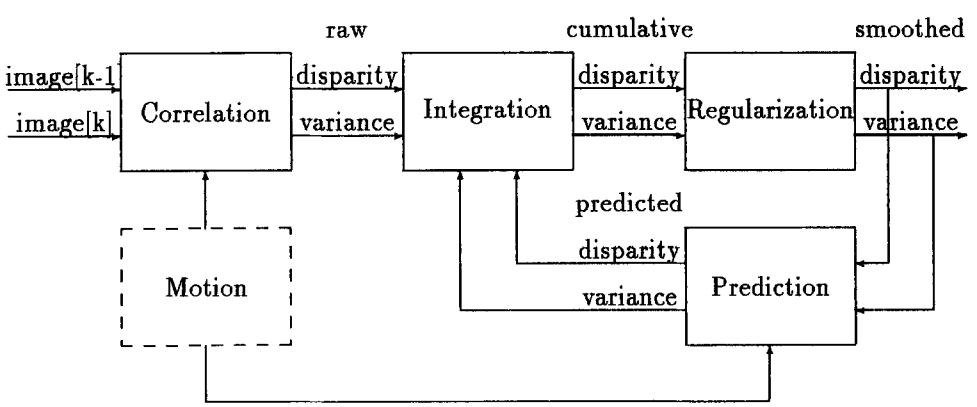

Fig. 9. Iconic depth estimation block diagram.

The last stage uses the known camera motion to predict the disparity field that will be seen in the next frame, and resamples the field to keep it iconic (pixel-based).

The information propagated between these four stages consists of two fields (iconic maps). The first field is the disparity estimate computed at each pixel in the image. The second field is the variance associated with each disparity estimate. Modeling the variance at every pixel is essential because it can vary widely over the image, with low variance near edges and in textured areas, and high variance in areas of uniform intensity. These two fields roughly correspond to the cumulative data vector $\mathbf{b}_k$ and the information matrix $\mathbf{A}_k$ used in the previous section.

The incremental depth-from-motion algorithm was tested on a number of image sequences acquired in the Calibrated Imaging Laboratory at Carnegie Mellon University. To measure the accuracy of the algorithm and to determine its rate of actual convergence, Matthies et al. digitized an image sequence of a flat-mounted poster. The ground truth value for the depth was determined by fitting a plane to the measured values, and the accuracy of the estimates was determined by computing the RMS deviation of the measurements from the plane fit. These experiments showed that the algorithm converged to an error level of approximately 0.5% percent after processing eleven images. Since the poster was 20 inches from the camera, this equates to a depth error of 0.1 inches, whereas the overall baseline between the first and the eleventh image was only 0.44 inches. The incremental depth-from-motion algorithm was also tested on complicated, realistic scenes obtained from the Calibrated Imaging Laboratory (figure 10). From these experiments, it was found that the main structures of the scene were recovered quite well [Matthies et al. 1989]. This depth-from-motion algorithm is therefore

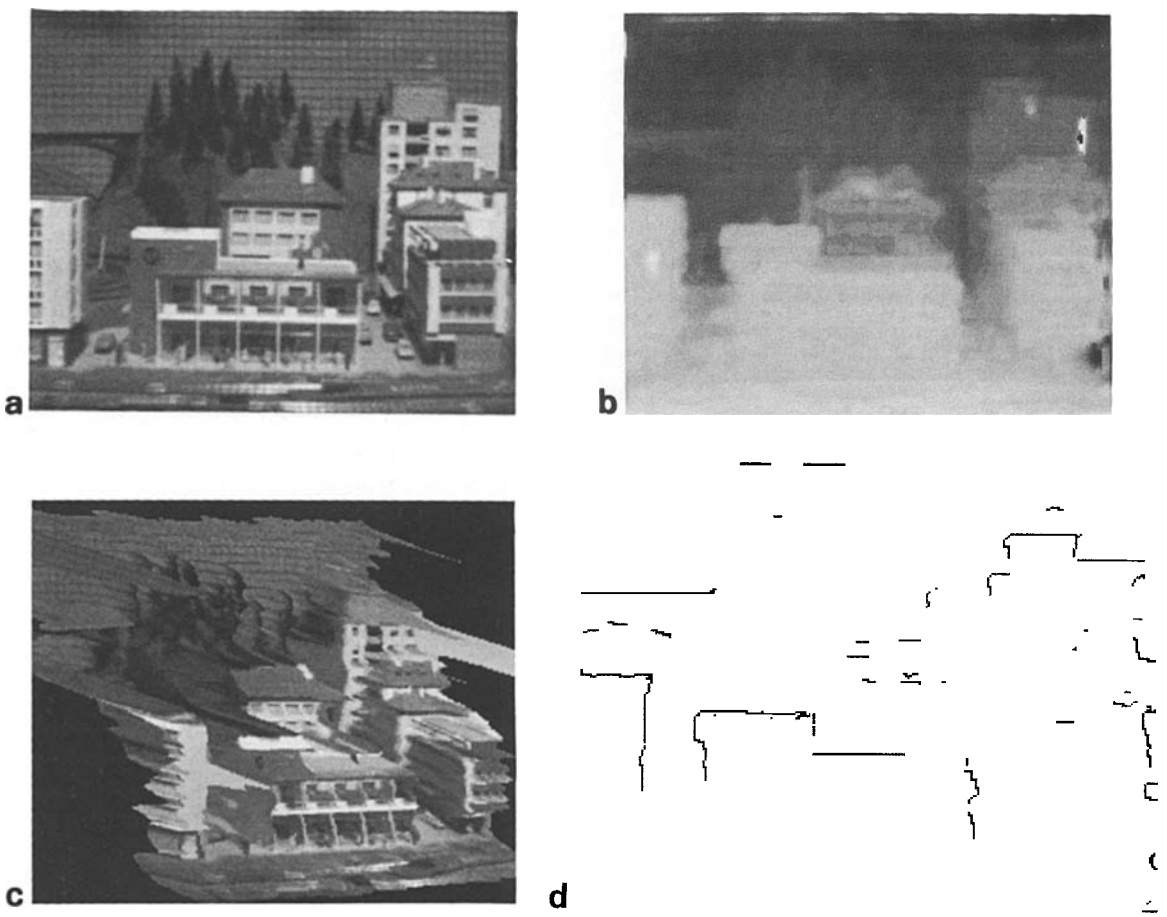

Fig. 10. Depth map computed from CIL image sequence: (a) first frame of image sequence; (b) intensity-coded depth map computed from combined sequence of horizontal and vertical motions; (c) perpsective view of intensity image texture-mapped onto depth map; (d) occluding boundaries computed during motion analysis.

a powerful demonstration of the advantages of using the Bayesian modeling framework for solving low-level vision problems.

8.2 Motion Estimation Without Correspondence

The probabilistic framework developed in this article shows how a sparse set of measurements can be converted into a dense iconic map, and how the uncertainty in this map can be modeled and estimated. This same framework can be used to solve an extended version of the motion estimation problem: given two sets of points that come from the same surface but from different viewing directions, what is the most likely coordinate transformation between the two sets? This question is important in robot navigation and manipulation applications where the motion of the observer or object is to be determined.

Traditionally, motion estimation and pose determination problems have assumed that a correspondence is given or is computable between the two sets of points to be matched [Ullman 1979]. The problem is then to find a transformation $T(\mathbf{p}, \Theta)$ such that the distance between the transformed points and the original points is minimized [Tsai and Huang 1984; Faugeras and Hebert 1987]. The method developed in [Szeliski 1988], which is based on our Bayesian framework, shows how to estimate this transformation even when no such correspondence exists. The two point sets can have a different number of points and a limited area of overlap. The approach is thus well suited for use with laser range finders or other active range sensors that do not sample the same points from different viewing positions. It is also particularly well suited for terrain maps, since it can handle data points that are irregularly spaced (from perspective de-projection), and it can incorporate prior knowledge from cartographic data. We will briefly summarize the motion estimation algorithm here; a more detailed description of the new algorithm is given in [Szeliski 1988].

In describing our algorithm, we use the notation introduced in section 7.1 and assume the same prior and measurement models. In general, the measurement vector $\mathbf{d}_k$ and the measurement matrix $\mathbf{H}_k$ depend (perhaps nonlinearly) on some coordinate transformation parameter vector Θ_k. For now, we assume that $\mathbf{d}_1$ and $\mathbf{H}_1$ are known, and that only the second set of measurements $\mathbf{d}_2(\Theta)$ and measurement matrix $\mathbf{H}_2(\Theta)$ are parameterized.

A simple approach for determining the motion parameters would be to interpolate the first set of data and to then measure the distance between the new set of points and the interpolated solution. Unfortunately, this approach has several problems. The matching of new data points to the extrapolated parts of the surface is inaccurate, since little is known about the surface in these areas. This is symptomatic of the more general problem with this technique, which is that it does not incorporate any knowledge about the uncertainty in the original interpolated surface. For example, range data will often have "shadowed" areas where the extrapolated data can be extremely uncertain. To overcome these problems, we must use a Bayesian model to derive the optimal motion estimator.

To compute the optimal estimate of the motion, we find the value Θ that makes it most likely that the two sets of data points $\mathbf{p}_1$ and $\mathbf{p}_2$ came from the same smooth surface. Skipping the details of the derivation [Szeliski 1988], we find that the new log likelihood $E(\mathbf{d}_2)$ can be written as

$$E(\mathbf{d}_2) = E_1(\mathbf{d}_2) + E_2(\mathbf{d}_2) \tag{50}$$

where

$$E_1(\mathbf{d}_2) = \frac{1}{2}\log|2\pi\mathbf{R}_2^{-1}| + \frac{1}{2}\log|\mathbf{P}_1^{-1}| - \frac{1}{2}\log|\mathbf{P}_2^{-1}| \tag{51}$$

and

$$E_2(\mathbf{d}_2) = \frac{1}{2}(\mathbf{d}_2 - \mathbf{H}_2\hat{\mathbf{u}}_1)^T\mathbf{R}_2^{-1}(\mathbf{d}_2 - \mathbf{H}_2\hat{\mathbf{u}}_2) \tag{52}$$

The first component of the energy, E_1, measures the reduction in likelihood due to the sensor noise as traded off against the increase in posterior information. In practice, this component of the energy varies fairly slowly with the transformation parameter Θ and can usually be ignored. The second part of the energy, E_2, measures the distance between the new data points $\mathbf{d}_2$ and the surfaces $\hat{\mathbf{u}}_1$ and $\hat{\mathbf{u}}_2$, where $\hat{\mathbf{u}}_1$ is the surface interpolated through the first set of points, and $\hat{\mathbf{u}}_2$ is the surface interpolated through both sets of points. This term is similar to a simple squared distance between the points and the old interpolated surface $\hat{\mathbf{u}}_1$, except that one side of the quadratic form uses the *new* surface estimate $\hat{\mathbf{u}}_2$. Points that lie closer to the new surface than to the old estimate are thus penalized less by the optimal energy measure. In this way, areas where the surface values are originally uncertain (because the data is uncertain,

the area is shadowed, or the surface is being extrapolated) contribute less to the matching criterion.

An additional advantage of the Bayesian model for motion estimation is that we can compute the uncertainty in the motion estimate directly from the shape of the error surface $E(\mathbf{d}_2; \Theta)$ [Szeliski 1988]. This variance estimate can be used to integrate the motion estimate provided by the new algorithm with other motion estimates, such as those provided by dead reckoning or inertial navigation systems.

8.3 Regularization Parameter Estimation

One of the recurring problems associated with regularization and other energy-based estimation techniques is the need to select good values for the global parameters that control the algorithm. Some progress has been made in this direction [Craven and Wahba 1979], but mostly these parameters are still adjusted by hand. The advantage of using a Bayesian approach to low-level vision is that the unknown parameters can often be derived from knowledge of the problem domain or from the data itself. In particular, the Bayesian interpretation of regularization has been used to develop a maximum likelihood estimator for the regularization parameter λ [Szeliski 1989]. A brief description of this approach follows.

Consider the case where the measurement noise can be determined from knowledge about the sensors. In this case, we can use the parameterization

$$p(\mathbf{u}|\mathbf{d}) \propto \exp - [E_d(\mathbf{u}, \mathbf{d}) + E_p(\mathbf{u})/\sigma_p^2] \tag{53}$$

where $\lambda^{-1} = \sigma_p^2$ simply encodes the overall variance in the prior model. Intuitively, if σ_p is very low, then typical surfaces are extremely flat or planar, and it is unlikely that the given (nonflat) data sample would actually occur. Similarly, if typical surfaces are very rough, then the probability of a given data sample occurring becomes small. There exists some optimal value of σ_p that maximizes the probability $p(\mathbf{d})$ of actually having observed the given data.

To compute the maximum likelihood estimate of σ_p, we write

$$p(\mathbf{d}) = |2\pi(\mathbf{H}\mathbf{P}_0\mathbf{H}^T + \mathbf{R})|^{-1/2} \exp\left[-\frac{1}{2}\mathbf{d}^T(\mathbf{H}\mathbf{P}_0\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{d}\right] \tag{54}$$

where $\mathbf{H}$, $\mathbf{R}$, and $\mathbf{d}$ are defined by the usual measurement equation (46), and $\mathbf{P}_0^{-1} = \sigma_p^{-2}\mathbf{A}_p$ is the information matrix parametrized by σ_p. The negative logarithm of this distribution can be written as

$$E(\mathbf{d}) = -\log p(\mathbf{d}) = E_1(\mathbf{d}) + E_2(\mathbf{d}) \tag{55}$$

where $E_1(\mathbf{d})$ is the logarithm of the partition function and $E_2(\mathbf{d})$ is the energy (quadratic form) associated with the Gaussian. From [Szeliski 1989], we have

$$E_1(\mathbf{d}) = \frac{1}{2}\log|\sigma_p^{-2}\mathbf{A}_p + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}| - \frac{1}{2}\log|2\pi\mathbf{R}^{-1}| - \frac{1}{2}\log|\sigma_p^{-2}\mathbf{A}_p| \tag{56}$$

and

$$E_2(\mathbf{d}) = \frac{1}{2}\mathbf{d}^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\hat{\mathbf{u}}_1) \tag{57}$$

$$= \frac{1}{2}(\mathbf{d} - \mathbf{H}\hat{\mathbf{u}}_1)^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\hat{\mathbf{u}}_1) + \frac{\sigma_p^{-2}}{2}\hat{\mathbf{u}}_1^T\mathbf{P}_0^{-1}\hat{\mathbf{u}}_1 \tag{58}$$

The log partition function $E_1(\mathbf{d})$ is minimized as $\sigma_p \to 0$, where its value approaches $\log|2\pi\mathbf{R}|$. Note that for a membrane or a thin plate, $\mathbf{A}_p$ is singular, so we must add a small diagonal element $\epsilon\mathbf{I}$ to this matrix when evaluating $E_1(\mathbf{d})$. The energy function $E_2(\mathbf{d})$, on the other hand, decreases as $\sigma_p \to \infty$, since in this case $\hat{\mathbf{u}}_1 \to \mathbf{d}$ as we approach the interpolated solution. Note that this energy can be written in two different ways. The first form, which is easier to compute, simply measures the weighted dot product of the data points and the residuals to the interpolated surface. The second form shows how this energy can be partitioned into a data-compatibility term and a smoothness term, both of which are evaluated with respect to the minimum energy solution.

The maximum likelihood approach described in this section is just one possible method for estimating the desired degree of smoothing [Cravan and Wahba 1979]. A comparison of our algorithm with previously developed techniques is presented in [Szeliski 1989]. The exact theoretical connection between our new approach and these other methods and their relative performance remains to be investigated.

9 Mechanical vs. Probabilistic Models.

In their book *Visual Reconstruction*, Blake and Zisserman [1987] lay out an elegant approach to the problem

of piecewise continuous surface reconstruction. In explaining their method, Blake and Zisserman argue that a deterministic mechanical (energy-based) approach is preferable to a stochastic probabilistic approach. We believe that the converse is true, that is, that the Bayesian approach has many advantages over the mechanical one.

One of Blake and Zisserman's main arguments in favor of the mechanical viewpoint is that the models should be continuous. Using such continuous models facilitates variational analysis, allows the implementation of viewpoint-invariant interpolators, and matches the continuous nature of surfaces and intensity fields in the real world. Fortunately, continuous models are not incompatible with probabilistic modeling. As we have seen in the previous section, regularization-based models are equivalent to correlated Gaussian fields. Even models such as viewpoint-invariant interpolators that do not have quadratic energy functions can be turned into probability distributions through the use of the Gibbs distribution.

A continuous field, of course, cannot be simulated on a computer without first discretizing the energy or probability equations. The best way to perform this discretization for mechanical models is to use finite element analysis. The same discrete equations that are derived from the mechanical model can then be used to define a Markov random field. We can thus view the MRF as a discretized version of a continuous pseudo-Markovian field. The parameters for this field need not be computed by assigning conditional probabilities. They can be derived the same way as they are for mechanical models, that is, using parameters with natural interpretations or physical correlates [Geiger and Girosi 1989].

Because of the equivalence between energy functions and probability density functions that can be established using the Gibbs distribution, we can design probabilistic models that have the exact same performance as that obtained with mechanical models. The question is then "what possible advantages do probabilistic models offer?" One advantage is that we can develop probabilistic sensor models that closely match the characteristics of real sensors (section 5). Another advantage is that we can choose different loss functions to use with our posterior estimator (section 6.1), whereas mechanical models always find the MAP estimate. With a probabilistic model we can also determine the uncertainty in our estimate (section 6.2), which corresponds to determining the local stiffness of the mechanical model. Additional applications of the probabilistic approach were presented in section 8.

Probabilistic modeling need not be restricted to two-dimensional fields. We can apply the same techniques for converting energy-based models into Bayesian priors to three-dimensional models such as those being investigated by Terzopoulos et al. [1987]. Figure 11 shows a three dimensional elastic net (based on the two-dimensional nets developed by Durbin and Willshaw [1987, 1989]) that was obtained by tessellating a sphere. The energy equation for this net is

$$E(\mathbf{p}_i) = \sum_i \sum_{j \in N_i} |\mathbf{p}_i - \mathbf{p}_j|^2 - \rho \sum_i |\mathbf{p}_i|$$

Applying the Gibbs sampler to this system, we obtain the typical sample shown in figure 11. This figure resembles the examples of fractal textured spheres shown in [Mandelbrot 1982] and [Pentland 1986] that were generated by adding fractal texture onto the surface of a sphere.

From this example, we see that the difference between intrinsic models that describe visible (retinotopic) surfaces and three-dimensional energy-based models that describe objects may not be that large. Bayesian modeling may thus serve as a common mathemetical framework for describing the multiple transformations that occur in going from images to three-dimensional models. Moreover, the uncertainty computed at an earlier stage of processing can be used to derive the uncertainty in later estimates, for example, the uncertainty in the object model shape or position parameters can be computed from the probabilistic description of the

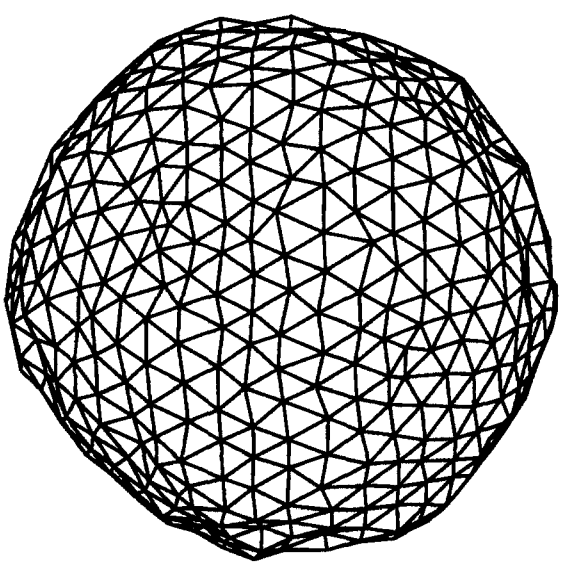

Fig. 11. Random sample from a three-dimensional elastic net.

surface. We believe that the development of appropriate intrinsic models for intermediate-level visual representations will prove to be an interesting and important research topic.

In the end, the difference between mechanical and probabilistic models may not be that large, since probabilistic systems based on MRFs have mechanical analogues and vice versa [Geiger and Girosi 1989]. The mechanical approach may be prefereable for developing energy equations and specifying model parameters. For MAP estimation, specially tailored deterministic algorithms—such as those developed by Witkin et al. [1987] and Blake and Zisserman [1987]—should perform better than general stochastic optimization. On the other hand, the probabilistic approach enables the development of more sophisticated estimates, including the use of different loss functions and the estimation of model parameters.

10 Conclusions

The main focus of this article has been the development of a Bayesian framework for modeling dense fields and their associated uncertainties. Such fields are used in low-level vision to represent visible surfaces and intrinsic images. These retinotopic maps form a useful intermediate representation for integrating information from different low-level vision modules and sensors. Modeling the uncertainty in these maps is an essential component of the integration process and provides a richer description for later stages of processing.

The Bayesian framework we have developed is based on three separate probabilistic models. The prior model describes the a priori knowledge that we have about the structure of the visual world. The sensor model describes how individual measurements (such as image intensities) are obtained from a particular scene. The posterior model is derived from the first two models using Bayes' rule and describes our current estimate of the scene given the measurements. By examining each of these models in turn, we have developed new algorithms for low-level vision problems and new insights into existing algorithms.

In studying the prior model, we have analyzed the statistical assumptions of regularization-based smoothing. The prior model captures the smoothness or coherence assumptions associated with a visible surface. We construct this model using the smoothness constraint (stabilizer) from regularization as the energy function of a Markov random field. Using Fourier analysis, we have shown how the choice of the stabilizer determines the power spectrum (and hence the correlation function) of the prior model.

In studying sensor models, we have reviewed the equivalence between Gaussian sensor noise and a simple spring constraint, and shown how to extend this analysis to a mixture of Gaussians model. We have also applied sensor modeling to correlation-based optical flow measurements, where the uncertainty in the estimates is derived from the shape of the local error surface, thereby accounting for the spatially varying reliability of the estimates.

In studying the posterior model, we have shown how to calculate the uncertainty in the posterior estimate from the energy function of the system, and we have developed two new algorithms to perform this computation. The first algorithm uses deterministic relaxation to calculate the uncertainty at each point separately. The second algorithm generates typical random samples from the posterior distribution and calculates statistics based on these samples. The uncertainty map obtained from these algorithms can be used to set confidence limits on our measurements or to suggest where further active sensing is required.

Our Bayesian framework has been extended to temporal seuqences using a two-dimensional generalization of the Kalman filter. By paying careful attention to computational issues and to alternative representations, we obtain simple formulations for the updating equations.

To demonstrate the usefulness of the Bayesian framework developed here, we have presented three novel computer vision algorithms. The first algorithm estimates optical flow from successive pairs of images and incrementally refines the resulting disparity estimates and their associated confidences. This algorithm produces a dense on-line estimate of depth that improves over time. Experiments with real images have demonstrated the improved accuracy that can be obtained with this approach and have shown that the reconstructed depth maps of the scene are quite realistic. The second algorithm determines observer or object motion given two or more sets of sparse depth measurements. The algorithm determines this motion—without requiring any correspondence between the sensed points—by maximizing the likelihood that point sets come from the same smooth surface. The third algorithm estimates the optimal amount of smoothing to be used with regularization. This estimate is obtained by maximizing the likelihood of the data points that were observed given

a particular (parameterized) prior model, which results in a statistically valid, data-driven method for determining this important parameter.

10.1 Future Research

The Bayesian framework we have developed has been applied to visible surfaces (2½-D sketches) and to surface interpolation and depth-from-motion. In future work we plan to extend our Bayesian approach to other visual representations and to other computer vision problems. These include the extension to full three-dimensional models and to multiple intrinsic images, as well as the development of more general depth-from-motion and shape reconstruction algorithms.

The extension to viewpoint-invariant surface models and energy-based three-dimensional models should be straightforward. To perform this extension, we use the smoothness energy associated with the surfaces to define the prior model through a Gibbs distribution. While the resulting distributions are no longer correlated Gaussians (because the energy functions are not quadratic), they are still Markov random fields (because of the local structure of the energy). Applying the Bayesian approach to these representations should produce similar benefits to those demonstrated here, including the ability to use better sensor models and the ability to characterize the uncertainty in the estimates. We could also examine the extension to locally tensioned splines and to nonspline models such as the constant curvature sign models suggested by Blake and Zisserman [1987].

We plan to apply the Bayesian modeling approach to multiple intrinsic images, thus providing a unified framework for describing many different low-level vision algorithms. For example, we can enhance our new depth-from-motion algorithm by estimating the reflectance functions (albedos) of the visual surfaces, and thus incorporate shading cues into the reconstruction process. We also plan to study the more general idea of intrinsic models—probabilistic descriptions of intrinsic images—and how to link these models to higher-level three-dimensional models. In particular, we should examine how to use the uncertainty in the intermediate-level estimates to determine the uncertainty in three-dimensional model parameters. For this approach to be viable, however, we will first have to solve the problems of grouping, segmentation, and discontinuity detection.

The extension of our depth-from-motion algorithm to general motion is another area of future research. Combining this idea with full three-dimensional models, we could construct an active vision system that builds a three-dimensional description of its environment by roaming around. The Bayesian modeling of surfaces that we have developed would be an essential component of such a system, allowing information from many viewpoints and sensor modalities to be integrated in a natural and statistically optimal fashion. These are just some of the directions in which we plan to extend our Bayesian model, which has already proved to be a powerful, practical, and general framework for low-level vision.

Acknowledgments

This article is an abridged version of my Ph.D. thesis from Carnegie Mellon University. I would would like to thank my thesis advisors, Geoffrey Hinton and Takeo Kanade, for their ideas, encouragement and guidance during my five year stay at Carnegie Mellon. Jon Webb and Alex Pentland read an early version of this work, and I thank them for their comments and additional ideas and insights. My work was much enriched by discussions with the members of the IUS vision research group at Carnegie Mellon, especially Steve Shafer, Martial Hebert, Larry Matthies, and In So Kweon, and with members of the Boltzmann research group, especially David Plaut. Since my graduation from Carnegie Mellon, I have been fortunate to work at Schlumberger Palo Alto Research, at SRI International, and at Digital Equipment Corporation. The support of these organizations has allowed me to pursue the ideas described here, and I have benefited much from interactions with Jay Tenenbaum, Demetri Terzopoulos, Martin Fischler, Yvan Leclerc, Gudrun Klinker, and other researchers at these institutions.

References

Ackley, D.H., Hinton, G.E., and Sejnowski, T.J. 1985. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9: 147-169.

Adelson, E.H., and Bergen, J.R. 1985. Spatiotemporal energy models for the perception of motion. *J. Opt. Soc. Amer.* A2: 284–299.

Aloimonos, J., Weiss, I., and Bandyopadhyay, A. 1987. Active vision. In *Proc. 1st Intern. Conf. Comput. Vision*, London, pp. 35–54.

Anandan, P. 1989. A computational framework and an algorithm for the measurement of visual motion. *Intern. J. Comput. Vision* 2: 283–310.

Anandan, P., and Weiss, R. 1985. Introducing a smoothness constraint in a matching approach for the computation of displacement fields. In *Proc. DARPA Image Understanding Workshop*, Miami Beach, FL, pp. 186–196.

Baker, H.H., and Bolles, R.C., 1989. Generalizing epipolar-plane image analysis on the spatiotemporal surface. *Intern. J. Comput. Vision* 3: 33–49.

Barnard, S.T. 1989. Stochastic stereo matching over scale. *Intern. J. Comput. Vision* 3: 17–32.

Barnard, S.T., and Fischler, M A. 1982. Computational stereo. *Computing Surveys* 14: 553–572.

Barrow, H.G., and Tenenbaum, J.M. 1978. Recovering intrinsic scene characteristics from images. In Allen R. Hanson and Edward M. Riseman (eds) *Computer Vision Systems*, pp. 3–26. Academic Press: New York.

Bertero, M., Poggio, T., and Torre, V. 1987. Ill-posed problems in early vision. A.I. Memo 924, Massachusetts Institute of Technology.

Bierman, G.J. 1977. *Factorization Methods for Discrete Sequential Estimation*. Academic Press: New York.

Blake, A., and Zisserman, A. 1987. *Visual Reconstruction*. MIT Press: Cambridge, MA.

Bolles, R.C., Baker, H.H., and Marimont, D.H. 1987. Epipolar-plane image analysis: An approach to determining structure from motion. *Intern. J. Comput. Vision* 1: 7–55.

Boult, T.E. 1986. Information based complexity in non-linear equations and computer vision, Ph.D. thesis, Columbia University.

Bracewell, R.N. 1978. *The Fourier Transform and Its Applications*. 2nd ed. McGraw-Hill: New York.

Briggs, W.L. 1987. *A Multigrid Tutorial*. Society for Industrial and Applied Mathematics: Philadelphia.

Burt, P.J., and Adelson, E.H. 1983. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.* COM-31: 532–540.

Canny J. 1986. A computational approach to edge detection. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-8: 679–698.

Chen, L.-H., and Boult, T.E. 1988. An integrated approach to stereo matching, surface reconstruction and depth segmentation using consistent smoothness assumptions. In *Proc. DARPA Image Understanding Workshop*, Cambridge, MA, pp. 166–176.

Choi, D.J. 1987. Solving the depth interpolation problem on a fine grained, mesh- and tree-connected SIMD machine. In *Proc. DARPA Image Understanding Workshop*, Los Angeles, pp. 639–643.

Christ, J.P. 1987. Shape estimation and object recognition using spatial probability distributions. Ph.D. thesis, Carnegie Mellon University.

Craven, P., and Wahba, G. 1979. Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik* 31: 377–403.

Crowley, J.L., and Stern, R.M. 1982. Fast computation of the difference of low-pass transform. Tech. Rept. CMU-RI-TR-82-18: The Robotics Institute, Carnegie Mellon University.

Dev, P. 1974. Segmentation processes in visual perception: a cooperative neural model. COINS Technical Report 74C-5, University of Massachusetts at Amherst.

Duda, R.O., and Hart, P.E. 1973. *Pattern Classification and Scene Analysis*, Wiley: New York.

Durbin, R., Szeliski, R., and Yuille, A. 1989. An analysis of the elastic net approach to the travelling salesman problem. *Neural Computation* 1: 348–358.

Durbin, R , and Willshaw, D. 1987. An analogue approach to the traveling salesman problem using an elastic net method. *Nature* 326: 689–691.

Durrant-Whyte, H.F. 1987. Consistent integration and propagation of disparate sensor observations. *Intern. J. Robotics Res.* 6: 3–24.

Elfes, A., and Matthies, L. 1987. Sensor integration for robot navigation: Combining sonar and stereo range data in a grid-based representation. In *Proc. IEEE Conf. Decision and Control*.

Faugeras, O.D., Ayache, N., and Faverjon B. 1986. Building visual maps by combining noisy stereo measurements. In *Proc. IEEE Intern. Conf. Robotics and Automation*, San Francisco, pp. 1433–1438.

Faugeras, O.D., and Hebert, M. 1987. The representation, recognition and positioning of 3-D shapes from range data. In Takeo Kanade (ed.). *Three-Dimensional Machine Vision*. Kluwer Academic Publishers: Boston, pp. 301–353.

Gamble, E., and Poggio, T. 1987. Visual integration and detection of discontinuities: The key role of intensity edges. A.I. Memo 970, Artif. Intell. Lab., Massachusetts Institute of Technology.

Geiger, D., and Girosi, F. 1989. Mean field theory for surface reconstruction. In *Proc. Image Understanding Workshop*, Palo Alto, CA, pp. 617–630.

Gelb, Arthur (ed.). 1974. *Applied Optimal Estimation*. MIT Press: Cambridge, MA.

Geman, S., and Geman, D. 1984. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. In *IEEE Trans. Patt. Anal. Mach. Intell. PAMI*-6: 721–741.

Geman, S., and McClure, D.E. 1987. Statistical methods for tomographic image reconstruction. In *Proc. 46th Session of the Intern. Statistical Inst.*

Grimson, W.E.L. 1981. *From Images to Surfaces: a Computational Study of the Human Early Visual System*. MIT Press: Cambridge, MA.

Grimson, W.E.L. 1983. An implementation of a computational theory of visual surface interpolation. *Comput. Vision, Graphics, and Image Process.* 22: 39–69.

Hackbusch, W. 1985. *Multigrid Methods and Applications*. Springer-Verlag: Berlin.

Heeger, D.J. 1987. Optical flow from spatiotemporal filters. In *Proc. 1st Intern. Conf. Comput. Vision*, London, pp. 181–190.

Heel, J. 1989. Dynamic motion vision. In *Proc. Image Understanding Workshop*, Palo Alto, CA, pp. 702–713.

Hinton, G.E. 1977. Relaxation and its role in vision. Ph.D. thesis, University of Edinburgh.

Hinton, G.E., Sejnowski, T.J. 1983. Optimal perceptual inference. In *Proc. Conf. Comput. Vision and Patt. Recog.*, Washington, D.C., pp. 448–453.

Hoff, W., and Ahuja, N. 1986. Surfaces from stereo. In *Proc. 8th Intern. Conf. Patt. Recog.*, Paris, pp. 516–518.

Horn, B.K.P. 1977. Understanding image intensities. *Artificial Intelligence* 8: 201–231.

Horn, B.K.P., and Brooks, M.J. 1986. The variational approach to shape from shading. *Comput. Vision, Graphics, Image Process.* 33: 174–208.

Horn, B.K.P., and Schunck, B.-G. 1981. Determining optical flow. *Artificial Intelligence* 17: 185–203.

Hueckel, M.H. 1971. An operator which locates edges in digitized pictures. *J. Assoc. Comput. Mach.* 18: 113–125.

Ikeuchi, K., and Horn B.K.P. 1981. Numerical shape from shading and occluding boundaries. *Artificial Intelligence* 17: 141–184.

Julesz, B. 1971. *Foundations of Cyclopean Perception*. Chicago University Press: Chicago.

Kass, M., Witkin, A., and Terzopoulos, D. 1988. Snakes: Active contour models. *Intern. J. Comput. Vision* 1: 321–331.

Kimeldorf, G., and Wahba, G. 1970. A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *Ann. Math. Stat.* 41: 495–502.

Kirkpatrick, S., Gelatt, C.D., Jr., and Vecchi, M.P. 1983. Optimization by simulated annealing. *Science* 220: 671–680.

Koch, C., Marroquin, J., and Yuille, A. 1986. Analog "neuronal" networks in early vision. *Proc. Nat. Acad. Sci. U.S.A.* 83: 4263–4267.

Konrad, J., and Dubois, E. 1988. Multigrid Bayesian estimation of image motion fields using stochastic relaxation. In *Proc. 2nd Intern. Conf. Comput. Vision*, Tampa, FL, pp. 354–362.

Leclerc, Y.G. 1989. Constructing simple stable descriptions for image partitioning. *Intern. J. Comput. Vision* 3: 75–102.

Lowe, D.G. 1985. *Perceptual Organization and Visual Recognition*. Kluwer Academic Publishers: Boston.

Mandelbrot, B.B. 1982. *The Fractal Geometry of Nature*. W.H. Freeman: San Francisco.

Marr, D. 1978. Representing visual information. In Allen R. Hanson and Edward M. Riseman (eds.), *Computer Vision Systems*, pp. 61–80, Academic Press: New York.

Marr, D. 1982. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W.H. Freeman: San Francisco.

Marr, D., and Hildreth, E. 1980. Theory of edge detection. *Proc. Roy. Soc. London* B 207: 187–217.

Marr, D., and Poggio, T. 1976. Cooperative computation of stereo disparity. *Science* 194: 283–287.

Marroquin, J.L. 1984. Surface reconstruction preserving discontinuities. A.I. Memo 792. Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

Marroquin, J.L. 1985. Probabilistic Solution of Inverse Problems. Ph.D. thesis, Massachusetts of Technology.

Matthies, L., and Shafer, S.A. 1987. Error modeling in stereo navigation. *IEEE J. Robotics Automation* RA-3: 239–248.

Matthies, L.H., Kanade, T., and Szeliski, R. 1989. Kalman filter-based algorithms for estimating depth from image sequences. *Intern. J. Comput. Vision* 3: 209–236.

McDermott, D. 1980. Spatial inferences with ground, metric formulas on simple objects. Department of Computer Science, Yale University, Res. Rept. 173.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E. 1953. Equations of state calculations by fast computing machines. *J. Chem. Physics* 21: 1087–1091.

Moravec, H.P. 1988. Sensor fusion in certainty grids for mobile robots. *AI Magazine* 9: 61–74.

Pentland, A.P. 1986. Perceptual organization and the representation of natural form. *Artificial Intelligence* 28: 293–331.

Poggio, T., Torre, V., and Koch, C. 1985. Computational vision and regularization theory. *Nature* 317: 314–319.

Poggio, T., Voorhees, H., and Yuille, A. 1985. A regularized solution to edge detection. A.I. Memo 833. Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

Poggio, T., et al. 1988. The MIT vision machine. In *Proc. DARPA Image Understanding Workshop*, Boston, pp. 177–198.

Rensink, R.A. 1986. *On the Visual Discrimination of Self-Similar Random Textures*. Master's thesis, The University of British Columbia.

Rives, P., Breuil, E., and Espiau, B. 1986. Recursive estimation of 3D features using optical flow and camera motion. In *Proc. Conf. Intell. Autonomous Systems*. pp. 522–532. (Also in 1987 *Proc. IEEE Intern. Conf. Robotics and Automation*.)

Roberts, L.G. 1965. Machine perception of three-dimensional solids. In Tippett et al., (eds.), *Optical and Electro-Optical Information Processing*, ch. 9, pp. 159–197, MIT Press: Cambridge, MA.

Rosenfeld, A. 1980. Quadtrees and pyramids for pattern recognition and image processing. In *5th Intern. Conf. Patt. Recog.*, Miami Beach, FL, pp. 802–809.

Rosenfeld, A. (ed.). 1984. *Multiresolution Image Processing and Analysis*. Springer-Verlag: New York.

Rosenfeld, A., Hummel, R.A., and Zucker, S.W. 1976. Scene labeling by relaxation operations. *IEEE Trans. Syst., Man, and Cybern.* SMC-6: 420–433.

Szeliski, R. 1986. Cooperative algorithms for solving random-dot stereograms. Tech. Rept. CMU-CS-86-133, Computer Science Department, Carnegie Mellon University.

Szeliski, R. 1987. Regularization uses fractal priors. In *Proc. 6th Nat. Conf. Artif. Intell.*, Seattle, pp. 749–754.

Szeliski, R. 1988. Estimating motion from sparse range data without correspondence. In *Proc. 2nd Intern. Conf. Comput. Vision*, Tampa, FL, pp. 207–216.

Szeliski, R. 1989. *Bayesian Modeling of Uncertainty in Low-Level Vision*. Kluwer Academic Publishers: Boston.

Szeliski, R. 1990a. Fast shape from shading. In *Proc. 1st European Conf. Comput. Vision*, Antibes, France, pp. 359–368.

Szeliski, R. 1990b. Fast surface interpolation using hierarchical basis functions. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-12: 513–528.

Szeliski, R., and Terzopoulos, D. 1989a. From splines to fractals *Computer Graphics* 23: 51–60.

Szeliski, R., and Terzopoulos, D. 1989b. Parallel multigrid algorithms and computer vision applications. In *4th Copper Mountain Conf. on Multigrid Methods*, Copper Mountain, Colorado, pp. 383–398.

Terzopoulos, D. 1983. Multilevel computational processes for visual surface reconstruction. *Comput. Vision, Graphics, Image Process.* 24: 52–96.

Terzopoulos, D. 1986a. Image analysis using multigrid relaxation methods. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-8: 129–139.

Terzopoulos, D. 1986b. Regularization of inverse visual problems involving discontinuities. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-8: 413–424.

Terzopoulos, D. 1987. Matching deformable models to images: Direct and iterative solutions. In *Topical Meeting on Machine Vision*, Washington, D.C., pp. 164–167.

Terzopoulos, D. 1988. The computation of visible-surface representations. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-10: 417–438.

Terzopoulos, D., Witkin, A., and Kass, M. 1987. Symmetry-seeking models and 3D object reconstruction. *Intern. J. Comput. Vision* 1: 211–221.

Tikhonov, A.N., and Arsenin, V.Y. 1977. *Solutions of Ill-Posed Problems*, V.H. Winston: Washington, D.C.

Tsai, R.Y., and Huang, T.S. 1984. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. Patt. Anal. Mach. Intell.* PAMI-6: 13–27.

Ullman, S. 1979. *The Interpretation of Visual Motion*. MIT Press: Cambridge, MA.

Van Essen, D.C., and Maunsell, J.H.R. 1983. Hierarchical organization and functional streams in the visual cortex. *Trends in Neuroscience* 6: 370–375.

Voss, R.F., 1985. Random fractal forgeries. In R.A. Earnshaw (ed.), *Fundamental Algorithms for Computer Graphics*. Springer-Verlag, Berlin.

Wahba, G. 1983. Bayesian "confidence intervals" for the cross-validated smoothing spline *J. Roy. Statist. Soc.* B 45: 133–150.

Waltz, D.L. 1975. Understanding line drawings of scenes with shadows. In P. Winston, (ed.), *The Psychology of Computer Vision*, McGraw-Hill, New York.

Witkin, A., Terzopoulos, D., and Kass, M. 1987. Signal matching through scale space. *Intern. J. Comput. Vision* 1: 133–144.

Yserentant, H 1986 On the multi-level splitting of finite element spaces. *Numerische Mathematik* 49: 379–412.

Notes

[1]We will show later that regularization is a special case of the more general Bayesian approach to the formulation of inverse problems.

[2]Once the system has latched-on to a good solution, it can then track changes in the scene using only a few iterations to correct its estimate. Similar arguments have recently been advanced in support of *dynamic deformable models* by Terzopoulos [1987] and Kass et al. [1988].

[3]A Markov random field can sometimes be designed such that it has the desired surface (e.g., a plane) in its null space [Leclerc 1989]. However, the estimation of the surface using the MRF will be much slower than direct least squares fitting.

[4]This is because the Jacobian $|\partial U/\partial u|$ is a constant for a linear operator.

[5]For stereo matching and many other vision problems, the energy function being minimized has many local minima, so some search technique must be used. Popular iterative search techniques include simulated annealing [Marroquin 1985; Szeliski 1986; Barnard 1989] and continuation methods [Terzopoulos 1988; Koch et al. 1986; Blake and Zisserman 1987; Witkin et al. 1987, Leclerc 1989]

[6]For quadratic energy functions, the MAP, MPM, and MMSE estimates are identical.

[7]The inverse covariance matrix is called the information matrix in the statistical literature. In finite element analysis, **A** is called the stiffness matrix and **b** is the force vector.