CVGIP: IMAGE UNDERSTANDING
Vol. 53, No. 3, May, pp. 237–252, 1991

INVITED PAPER

An Approach to Three-Dimensional Image Segmentation

SHIH-PING LIOU AND RAMESH C. JAIN

Artificial Intelligence Laboratory, Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, Michigan 48109

Received June 19, 1990; accepted October 11, 1990

The development of techniques for interpreting the structure of three-dimensional images, $f(x,y,z)$, is useful in many applications. A key initial stage in the signal to symbol conversion process, essential for the interpretation of the data, is three-dimensional image segmentation involving the processes of *partitioning* and *identification*. Most segmentation and grouping research in computer vision has addressed partitioning of 2D images, $f(x,y)$. In this paper, we present a parallel 3D image segmentation algorithm which, through the use of *α-partitioning* and *volume filtering*, segments 3D images such that the greylevel variation within each volume can be described by a *regression model*. Experimental results demonstrate the effectiveness of this algorithm on several real-world 3D images. © 1991 Academic Press, Inc.

1. INTRODUCTION

The development of techniques for interpreting the structure of three-dimensional images has attracted researchers in the area of medicine for many years [1–5]. One of the most widely known techniques is computed tomography (CT), a technique which allows physicians to diagnose clinical abnormalities from the display of three-dimensional reconstructed organ images. In computer vision, dynamic scene analysis has attracted researchers interested in transforming a sequence of noisy two-dimensional inputs into a description of a scene in terms of objects, their three-dimensional shape, and their motion through space. Even though many techniques have been developed for this purpose, most techniques use just two or three frames of a sequence. Only recently has the so-called *spatio-temporal* approach [6–11] started using longer sequences. This approach pictures time-varying stimuli as occupying a three-dimensional space, in which x and y are two spatial dimensions and t is the temporal dimension. The collected sequence of images is usually referred to as a spatio-temporal *image solid*. In many applications of mobile robot and other autonomous agents, the information about different properties in space must be acquired combining information from disparate sensors obtained from multiple viewpoints at different time instants. This task is facilitated by keeping hierarchical models of the environment. In such a model, the most detailed level will contain properties at every voxel [12]. These property values should be grouped to form higher level concepts.

A three-dimensional image forms an *image solid* which is represented as a three-dimensional matrix of *greylevels*, $f(i,j,k)$. Each greylevel represents certain relevant property associated with the location (i,j,k) in the modeled three-dimensional world. For CT images, the greylevel represents an estimate of the average of a physical parameter (attenuation coefficient) at the point (i,j) in the kth cross section image. For dynamic scenes, the greylevel is proportional to the radiant energy received in the electromagnetic band to which the sensor is sensitive in a small area around (i,j) in the kth frame of a sequence. In the environment model it represents one or more property values at point (i,j,k) in space.

Due to the potential for essentially unbounded complexity of three-dimensional image data, it is often necessary to abstract the sensor-derived signals into a relatively clear-cut three-dimensional description of properties in the modeled three-dimensional world. A key initial stage in this abstraction process is three-dimensional image segmentation which can be viewed as involving the two closely tied activities of *partitioning* and *identification* (Fig. 1a). The partitioning process hypothesizes a partition of 3D images into volumes. To justify whether or not such a hypothesized partition is valid, the identification process matches the underlying greylevel distribution of the partition subsets to a given set of *volume models*. Most segmentation and grouping research in computer vision has addressed the problem for two-dimensional images $f(i,j)$ and very little work has been done for the three-dimensional case. In this paper, we shall present an algorithm which can segment 3D images into coherent volumes and, upon complete segmentation, model each of them by individual description (in terms of a volume model).

1049-9660/91 $3.00

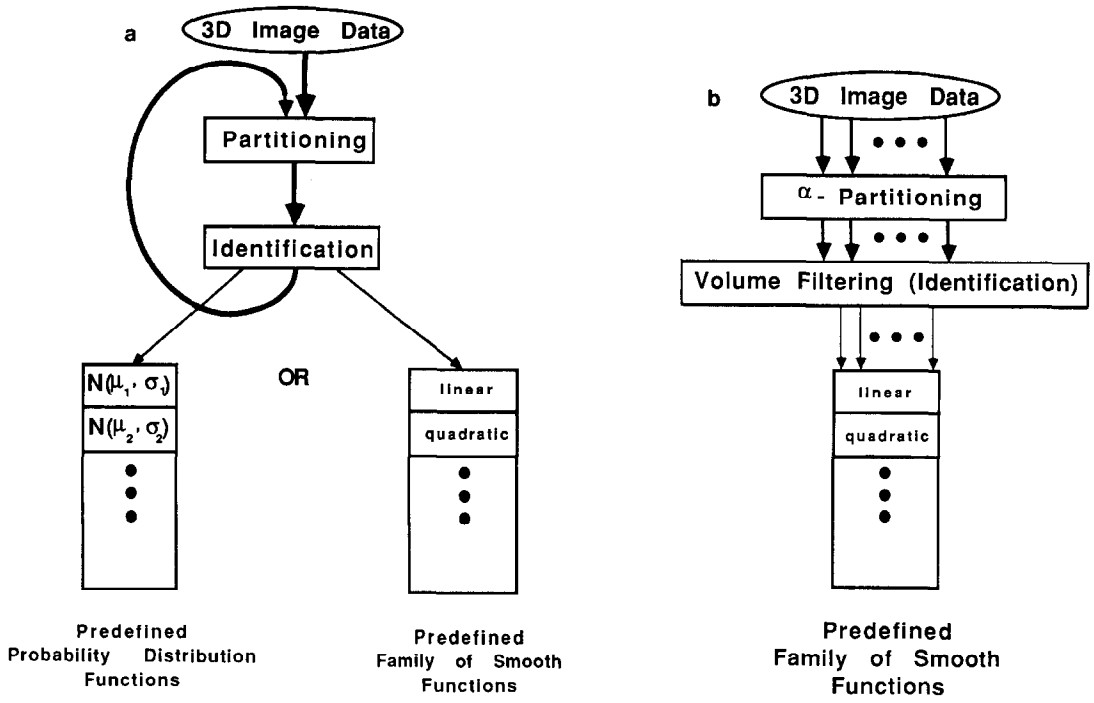

FIG. 1. (a) Approaching the 3D segmentation problem by extending conventional iterative region growing methods to three-dimensional space. (b) Using α-partitioning and volume-filtering to break the iterative structure in the parallel 3D image segmentation approach.

These descriptions are very important for a variety of applications. They can be used for identifying the objects of interest and indexing into a knowledge base to provide a basis for three-dimensional object recognition. In medical applications, they can be used to reconstruct three-dimensional organs and the display of such reconstructed organs can convey useful information to the physician in diagnosing clinical abnormalities. For interpreting dynamic scenes, the functional descriptions (i.e., volume descriptions) for the spatio-temporal 3-surface may be very useful for identifying moving objects in the scene.

The most widely used technique in attempts to address this 3D segmentation problem has been 3D edge detection [3, 4, 7, 13]. Similar to 2-dimensional edge detectors, the 3-dimensional edge detection method tries to locate edgels along the boundaries of volumes in the image solid. These edgels are then linked to form groups of edge segments, which are in turn 2-dimensional surfaces. Such surfaces are not guaranteed to form closed volume boundaries in many cases; therefore, the edge detection technique does not offer the level of abstraction desirable for the analysis of 3D images.

To achieve the objective of 3D image segmentation, the most straightforward way may be to extend the conventional region growing techniques for image segmentation to *volume growing* techniques for segmenting 3D images. Volume growing is a global hypothesis testing technique. Given initially poor or incorrect seed volumes, such techniques usually do not provide any mechanisms for detecting and rejecting local gross errors in situations such as when an initial seed volume spans two separate hypersurfaces. Therefore, the generation and filtering of good seed volumes of high confidence will be essential. The selected seeds also tend to be small in most real 3D images and merely correspond to subsets of the actual surfaces' pixels in the 3D image. The iterative process of volume growing must then be applied in order to recover the hypersurfaces of interest. When a hypothesis is matched to a given model during identification, the result is then sent back to the partitioning process for growing volumes at the next iteration. The process terminates whenever there are no more pixels to be identified as belonging to a volume model.

Conventionally, volume models have been represented using the *probabilistic approach* where a 3D image is modeled as a three-dimensional random field. The *functional approach* [14–18], which tires to capture the underlying greylevel distribution of an image using a family of smooth functions, has been used only for the segmentation of 2D images. Our 3D image segmentation algorithm is the 3D version of our earlier algorithm [19] and is based on a unified framework for modeling 3D images. Under this new framework, the probabilistic nature of the image formation process can be captured by assigning a functional description to each volume, effectively unifying the above probabilistic and functional approaches.

Similar to the earlier algorithm [19] which does not have any apparent disadvantage of conventional region growing techniques, this 3D image segmentation algorithm breaks the iterative structure of conventional volume growing through the use of *α-partitioning* and *volume filtering* (Fig. 1b). The mechanism of α-partitioning generates volume hypotheses, while volume filtering detects and rejects gross errors. By using gradient information, α-partitioning removes the need for volume growing so that our algorithm can be implemented on a parallel architecture. The proposed 3D image segmentation techniques segment 3D images such that the greylevel variation within each volume cluster can be described by a *regression model.*

This paper is organized as follows. The key ideas behind the 3D image segmentation algorithm are first described: the regression model (Section 2) and the techniques for breaking the iterative structure (Section 3). The implementation details are given next (Section 4) and experimental results show the algorithm's performance on three 3D images (Sections 5). It concludes with comments on future improvements (Section 6).

2. A REGRESSION MODEL FOR IMAGE SOLID

The modeling of the underlying greylevel variation in an image solid is essential in the 3D image segmentation process. What is an effective image solid model? Consider the sampling process in which an image solid has been obtained over a short period of time. Even in such a simple environment, if we repeat the process several times with the same sensor, the observational image solids will not be identical. Such a phenomenon can be attributed to the inherent measurement errors in the imaging process and the occurrences of certain unavoidable random events in this dynamic, though controlled, environment. In general, the stochastic nature of the imaging process cannot be modeled deterministically. An image solid has therefore to be treated as a three-dimensional discrete random field, a collection of random variables where values of each denotes the greylevel of the pixel in the image solid.

Every random variable which is *finite* and has *compact support* [20] can be uniquely determined by its complete set of moments $\{\mu_n\}$. For example, μ_1 and μ_2 determine the mean and standard deviation, respectively. In the general case, an image solid can then be treated as a realization of a vector random process $\mathbf{W}$ described by a set of moments $\vec{\mu}_n$. Therefore, by specifying all finite moments of this vector random process, we can always obtain an unique model for the underlying greylevel variation. In the special case where this random process has a Gaussian distribution, only the mean vector $E[\mathbf{W}]$ (the first moment vector) and the covariance matrix $\mathbf{R_W}$ (the second moment about the mean) are necessary for describing $\mathbf{W}$. The multivariate Gaussian

$$p(\vec{w}) = K \exp(-\tfrac{1}{2}(\vec{w} - E[\mathbf{W}])^T \mathbf{R}_{\mathbf{W}}^{-1}(\vec{w} - E[\mathbf{W}])) \quad (1)$$

has the property of being able to embody the correlation between the components of the vector $\vec{w}$

$$\vec{w} = (w_{\vec{r}_1}, w_{\vec{r}_2}, \ldots, w_{\vec{r}_n})$$

directly into the probability distribution function (pdf) in a compact and mathematically tractable fashion. ($w_{\vec{r}_i}$ is the ith vector random process sample associated with the pixel $r_i = (x_i,y_i,z_i)$.) Thus, for most practical purposes, it is the only pdf to be used as a model for multivariate vector data [21].

2.1. *The Nonstationary Mean and Stationary Autocovariance Assumption*

A wide-sense stationary assumption is usually adopted for the Gaussian pdf in modeling regular 2D images [22]. This assumption can be stated simply as

$$E[\mathbf{W}] = \vec{\mu}_1 = (M, M, \cdots, M)$$

and the autocovariance $R(\vec{r}_1,\vec{r}_2)$ of $\mathbf{W}$ defined by

$$R_{WW}(\vec{r}_1,\vec{r}_2) = E[(\mathbf{W}_{\vec{r}_1} - \mu_1(\vec{r}_1))(\mathbf{W}_{\vec{r}_2} - \mu_1(\vec{r}_2))]$$

$$= \delta_{i,j}\,\sigma^2,$$

where $\delta_{i,j}$ is the Kronecker delta function and σ^2 is a constant.

Hunt [23] demonstrates how greatly this conventional assumption of stationary mean can be violated by real-world images. However, by weakening the above stationary assumptions, Hunt and Cannon [21, 23] show that the Gaussian pdf of most statistical models can be derived on an *experimental* basis. They also claim that a digital image $f(i,j)$ can be modeled as consisting of intensity fluctuations about a non-stationary mean of a context-dependent ensemble

$$f(i,j) = \bar{f}(i,j) + s(i,j), \quad (2)$$

where $\bar{f}(i,j)$ is the low-frequency (or blurred) component and $s(i,j)$ is the high-frequency component. The nonstationary mean $\bar{f}(i,j)$ can be considered as the deterministic component of $f(i,j)$ and is estimated by blurring the individual ensemble sample member with a low-pass spatial filter. The random component $s(i,j)$ is shown to have approximately Gaussian behavior. Our regression model is based on an extension of this nonstationary mean and

stationary autocovariance assumption to the three-dimensional space.

2.2. *The Multiple Distribution Functions Assumption*

The assumption of Section 2.1 provides the basis for treating an image solid as a realization of a single vector random process with a nonstationary mean and stationary autocovariance. However, we believe that the modeling of an entire image solid with one distribution function defeats the purpose of the 3D image segmentation process. Instead, we consider an image solid I as the union of N volumes V_l, each of which is modeled by an independent Gaussian pdf f_{V_l} (l). An $m \times n \times l$ real image solid I can be represented as a continuous function of three variables $f(x,y,z)$. A digital image solid $f(i,j,k)$ is a matrix of samples of this function, $f(i\Delta x, j\Delta y, k\Delta z)$, with sampling intervals Δx, Δy, and Δz, for $0 \le i \le m$, $0 \le j \le n$, and $0 \le k \le l$. We denote $f(i,j,k)$ as an unregistered sample matrix $f(x_i,y_i,z_i)$ or $f(\vec{r}_i)$ where $f(\vec{r}_i)$ is considered as the ith sample from a vector random process. The multiple distribution functions assumption asserts

$$f(x_i,y_j,z_k) \sim f_{V_l} \quad \forall\, (x_i,y_i,z_i) \in V_l \quad \textit{and} \quad I = \cup_{l=1}^{N} V_l. \tag{3}$$

The objective of the 3D image segmentation process is then to recover not a single image solid model, but the individual volume models f_{V_l} for image solid I. During the recovery of a volume model, its associated nonstationary mean and stationary autocovariance have to be estimated.

2.3 *Modeling Nonstationary Means with Approximating Functions*

One way of estimating the nonstationary mean $\hat{f}$ in (2) is by blurring the 3D image with a low-pass three-dimensional filter.[1] In such case, the representation of $\hat{f}$ can be considered as an *mnl* vector for a $m \times n \times l$ image solid. Theoretically, $\hat{f}$ can also be modeled by a single polynomial functional description that best fits the greylevel values in the entire image solid. In practice, the best-fitting models can only be obtained through 3D image segmentation on a per volume model basis as suggested in [14]. We view the use of approximating functions in volume growing as a regression technique that uses a trivariate polynomial function $\hat{f}_{V_l}$ as a nonstationary mean model for f_{V_l}.

From the functional approximation point of view, the digital image solid $f_{V_l}(x_i,y_i,z_i)$ within a volume V_l is approximated by a function $\hat{f}_{V_l}(x,y,z)$ with the addition of measurement errors $\eta_{V_l}(x,y,z)$,

$$f_{V_l}(x,y,z) = \hat{f}_{V_l}(x,y,z) + \eta_{V_{l(x,y,z)}}, \tag{4}$$

under the assumption that the errors are uncorrelated within each volume such that $\eta_{V_l} \sim N(\mathbf{0}, \mathbf{I}\sigma^2_{V_l})$, for all (x,y,z) associated with V_l. (We shall drop the subscript V_l in the rest of our discussion on volume models.)

Obviously, $E[\hat{f}(x,y,z)] = \hat{f}(x,y,z)$. Since η is zero mean independent identically distributed (i.i.d.), $E[\eta] = \mathbf{0}$ and $R_{\eta\eta}(\vec{r}_i,\vec{r}_j) = \delta_{i,j}\sigma^2$. It follows that

$$\begin{aligned} E[f(x,y,z)] &= E[\hat{f}(x,y,z)] + E[\eta(x,y,z)] \\ &= \hat{f}(x,y,z) \end{aligned} \tag{5}$$

and

$$\begin{aligned} &R_{ff}(x_i,y_i,z_i;x_j,y_j,z_j) \\ &= E[(f(x_i,y_i,z_i) - \hat{f}(x_i,y_i,z_i))]E[(f(x_j,y_j,z_j) - \hat{f}(x_j,y_j,z_j))] \\ &= E[\eta(x_i,y_i,z_i)E[\eta(x_j,y_j,z_j)] \\ &= R_{\eta\eta}(x_i,y_i,z_i;x_j,y_j,z_j) \\ &= \delta_{i,j}\sigma^2, \end{aligned} \tag{6}$$

where $\delta_{i,j}$ is the Kronecker delta function and σ is the measurement variance. Therefore, we have shown that the functional approximation approach basically adopts the same nonstationary mean (from Eq. 5) and stationary autocovariance (from Eq. 6) assumptions that we have presented in Section 2.1. The functional and probabilistic approaches in volume modeling are thus unified under a single framework.

2.4. *Modeling Nonstationary Means with Fixed-Order Polynomial Functions*

In the simplest case, the nonstationary mean $\hat{f}$ can be modeled by a linear function

$$E[f(x,y,z)] = ax + by + cz + d, \tag{7}$$

where a, b, c, and d are four free coefficients needed to be estimated. Unfortunately, this usually does not capture the inherent complexity in 3D images of the real world.

Besl and Jain in the conclusion of [14] suggest using different orders of trivariate polynomial functions to embody the knowledge about various levels of smoothness in an image solid. We view this suggestion as modeling the nonstationary mean of the image solid with a vocabulary of variable-order *functional descriptions* from the perspective of our unified regression framework. Furthermore, we shall argue that it is actually unnecessary to find the most appropriate order of the polynomial function for representing this mean. To explain the concept, let us first define the functional space formed by the mod-

[1] An extension of the method suggested by Hunt and Cannon [21].

eling functions as the *P-order space*. The P-order space contains polynomial functions of degree 0, 1, . . . , P, where P is defined as the maximum variation order (MVO). This set of approximating functions can be written in the form of a single equation

$$\hat{f}(\text{MVO}, \vec{a}; x,y,z) = \sum_{i+j+k \le \text{MVO}} a_{ijk} x^i y^j z^k.$$

For example, when MVO = 4, this set of functions includes linear, triquadratic, tricubic, and triquartic polynomials.

There are three properties associated with this P-order space. First, the order P constrains the minimum level of smoothness hypothesized to occur in real image solids. In other words, P is the hypothesized maximum complexity of the underlying smoothness variation within each volume. Second, the experimental results included in this paper (Section 5) show that MVO = 4 (triquartic polynomials) is reasonable for a large class of image solids, i.e.,

$$\begin{aligned}\hat{f}(4,\vec{a};x,y,z) &= \sum_{i+j+k \le 4} a_{ijk} x^i y^j z^k \\ &= a_{000} + a_{100}x + a_{010}y + a_{001}z + a_{110}xy \\ &\quad + a_{011}yz + a_{101}xz + a_{200}x^2 + a_{020}y^2 \\ &\quad + a_{002}z^2 + a_{210}x^2y + a_{120}xy^2 + a_{021}y^2z \\ &\quad + a_{012}yz^2 + a_{201}x^2z + a_{102}xz^2 \\ &\quad + a_{111}xyz + a_{300}x^3 + a_{030}y^3 \\ &\quad + a_{003}z^3 + a_{310}x^3y + a_{220}x^2y^2 \\ &\quad + a_{130}xy^3 + a_{301}x^3z + a_{202}x^2z^2 \\ &\quad + a_{103}xz^3 + a_{031}y^3z + a_{022}y^2z^2 \\ &\quad + a_{013}yz^3 + a_{211}x^2yz + a_{121}xy^2z \\ &\quad + a_{112}xyz^2 + a_{400}x^4 + a_{040}y^4 \\ &\quad + a_{004}z^4.\end{aligned}$$

Third, a MVO-order polynomial function is sufficient for representing a volume since it is an unbiased estimate of any underlying volume models of order less than MVO. The last two properties lead us to conclude, contrary to the conventional viewpoint, that 4th order functions are sufficient for modeling nonstationary means of real image solids.

Previously, when MVO $\ge$ 1 has been selected, techniques such as the extension of the variable order surface-fitting algorithm [14] to three dimensions can be used to recover the so called *best descriptive model* for a volume V_l. Since not all the volumes have exactly MVO order of smoothness variation, overfitting may occur when a MVO-order polynomial function $\hat{f}_{V_l}$ is fitted to a V_l that can be best described by one of a lower order. This very common *model reduction* problem also exists in regression analysis. In general regression analysis, the functional relationships between variables are unknown and many tests have been developed to find this relationship [24]. In computer vision applications, the set of independent variables is fixed and the functional relationship between variables is modeled by a set of polynomial functions. The task is then to find a particular polynomial order and set of coefficients for specifying the functional relationship.

This task becomes unnecessary given the fixed MVO-order property of the P-order space based on regression theory [24]. In general, an estimate $\hat{\beta}$ is an unbiased estimate of the parameter β being estimated if

$$E[\hat{\beta}] = \beta$$

A *fitted model* is a hypothesis used to explain the observational data generated by a *true model*. A *full model* is the largest regression containing all terms; in our case, all the coefficients of the MVO-order polynomial function. Suppose that the greylevels $\mathbf{Y}$ in volume V_l are approximated by a MVO-order full model

$$E[\mathbf{Y}] = \mathbf{X}\Theta$$

with regression matrix $\mathbf{X}$, n data points $\mathbf{Y}$, and p regression coefficient Θ in the least squares sense (see Appendix A). (For example, when MVO = 4, p = 35 unknowns for triquartic polynomials.) Let us assume our underlying (true) model of a lower order, with k unknowns, is

$$E[\mathbf{Y}] = \mathbf{X}_1\Theta_1,$$

where $\mathbf{X}_1$ consists of the first $k < p$ columns of $\mathbf{X}$ so that $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2)$. Since the least squares estimate $\hat{\Theta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$ (by Eq. 11 in Appendix A),

$$\begin{aligned}E[\hat{\Theta}] &= E[(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}] \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T E[\mathbf{Y}] \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}_1\Theta_1 \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\begin{pmatrix}\Theta_1 \\ \mathbf{0}\end{pmatrix} \\ &= \begin{pmatrix}\Theta_1 \\ \mathbf{0}\end{pmatrix}.\end{aligned}$$

We have $\hat{\Theta} = (\hat{\Theta}_1, \hat{\Theta}_2)$ and $\hat{\Theta}_1$ consisting of the first k rows is, therefore, an unbiased estimate of Θ_1. The least squares estimate then allows us to obtain the reconstructed (or predicted) values $\hat{\mathbf{Y}}$ from the full model using

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{\Theta}}.$$

It follows that

$$E[\hat{\mathbf{Y}}] = E[\mathbf{X}\hat{\mathbf{\Theta}}] = \mathbf{X}\begin{pmatrix}\mathbf{\Theta}_1\\ \mathbf{0}\end{pmatrix} = \mathbf{X}_1\mathbf{\Theta}_1;$$

in other words,

$$E[\hat{\mathbf{Y}}] = E[\mathbf{Y}];$$

and so the MVO-order full model for V_l is an unbiased estimate of the underlying true model. This is the same as stating the following property. Assume that p regression coefficients $\{A_1, A_2, \cdots, A_p\}$ denoting a vector random process $\hat{\mathbf{\Theta}}$, are estimated at m different time instances. This set of estimates, obtained from fitting m MVO-order full models to m separate observations of V_l, has the property

$$E[A_{k+1}] = E[A_{k+2}] = \cdots = E[A_p] = 0$$

since $E[\hat{\mathbf{\Theta}}_2] = \mathbf{0}$.

In summary, we have shown that:

- A stochastic 3D image model is necessary for the 3D image segmentation process and the multivariate Gaussian model is selected for an 3D image.
- The nonstationary mean and stationary autocovariance assumption is reasonable for real-world 3D images.
- Due to the complexity of this modeling process, a reasonable 3D image model is the one that assumes different distribution functions (volume models) for different volumes in the image solid.
- The use of approximating functions as volume models in 3D image segmentation is the same as estimating the nonstationary mean of the 3D image through polynomial functional approximation.
- It is unnecessary to use variable-order polynomial functions for modeling the nonstationary mean, only polynomial function of order MVO = 4 is needed.

3. PARTITIONING AND IDENTIFICATION

A commonly used definition [14, 25] states that if I is the set of all pixels, a segmentation of I is a partitioning set of connected subsets, or image volumes, $\{V_1, V_2, \ldots, V_N\}$ such that

$$\cup_{i=1}^{N} V_l = I, \quad \textit{where} \quad V_l \cap V_m = \emptyset \quad \forall\, l \neq m,$$

where a *uniformity predicate* $P(\bullet)$ is defined on groups of connected pixels such that

$$P(V_l) = \textit{TRUE} \qquad \forall\, l$$

$$P(V_l \cup V_m) = \textit{FALSE} \quad \forall\, V_l \textit{ adjacent to } V_m.$$

A partition is defined to be *correct* when every volume hypothesis H_l for V_l satisfies the smoothness predicate $P(\bullet)$ and contains no gross error. A partition is *complete* when every H_l contains *exactly* all pixels $p_{i,j,k} \in V_l$. These two definitions also apply on a per volume basis and are illustrated in Fig. 2. The final partition from the 3D image segmentation process should have the image solid I segmented into the set $\{V_1, V_2, \cdots, V_N\}$ where each V_l is represented by a correct volume model f_{V_l}.

Similar to our formulation for the 2D image case in [19], a 3D image segmentation algorithm can also be viewed as the implementation of an uniformity predicate for the generation and verification of volume hypotheses and which upon completion yields an abstract representation of a 3D image in terms of volume models. Our 3D image segmentation algorithm stems from the regression model for 3D images (Section 2). In particular, its multiple functions assumption (by Eq. 3) is consistent with the above definition of segmentation. MVO-order polynomial functions are used to estimate $\hat{f}_{V_l}$ of the underlying volume models f_{V_l} that have zero mean i.i.d. Gaussian noise (Section 2). The uniformity predicate implemented is viewed as a *smoothness predicate* satisfying the following criteria:

- The smoothness variation of the pixel data in V_l is correctly and completely modeled by an $\hat{f}_{V_l}$.
- Discontinuities between V_l and V_m, where $P(V_l \cup V_m) = \textit{FALSE}$, are preserved to guarantee a correct final partition.

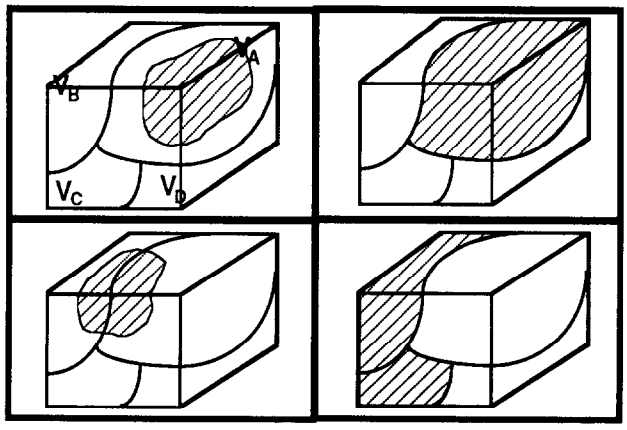

FIG. 2. Types of partitions: (upper left) Shaded area is correct but incomplete for V_A. (upper right) Shaded area is correct and complete for V_A. (lower left) Shaded area is incorrect and incomplete for V_A and V_B. (lower right) Shaded area is incorrect but complete for V_B and V_C. Note that we encounter *gross errors* where a volume hypothesis incorrectly spans two actual volumes.

The two techniques embedded in our implementation of the smoothness predicate are *α-partitioning* and *volume filtering*. α-partitioning generates volume hypotheses by taking advantage of the locations of discontinuities in the image solid. It preserves the discontinuities in the final correct partition and guarantees complete partitioning with accurate detection of such locations. Volume filtering verifies the smoothness variation of volume hypotheses and rejects gross errors by using the fitting variance S^2. S^2 is an unbiased estimate of the measurement error σ^2 when the model fits the data. The conventional iterative structure is not inherent in the 3D image segmentation algorithm because these two methods enable the algorithm to handle volume hypotheses in parallel and to combine the verified results efficiently through a very simple image solid operation (Image-Solid-OR) at the end.

3.1. *α-Partitioning for Generating Volume Hypotheses*

Similarly to our 2D algorithm [19], volume hypotheses are generated simultaneously by using information about the location of discontinuities. These often lie on surfaces that separate volumes of different smoothness variation in an image solid. The area surrounded by each closed boundary surface can then serve as a volume hypothesis. It is natural to use the gradient magnitude of a 3D image to locate the surrounding boundaries of volumes. However, various thresholds for edge strength may be required for different boundary surfaces in an image solid. Since the threshold value(s) for each surface is unknown *a priori*, we treat the segmentation problem as one of *finding the* minimal *set of edge strength thresholds, α_{MIN}-set, through which a* 3D *image can be segmented into volumes according to the smoothness predicate.*

To derive the α_{MIN}-set, we need sufficiently good estimates or we can find all of the edge strengths by searching over a window around these initial values in the measurement space. These *a priori* estimates can be obtained either from the vision system's prior experience or from the output of a high-level vision module in a goal-directed manner. In cases where such information may not be readily available to the segmentation module, it must be derived from the 3D image content instead.

This algorithm considers the latter case. The solution space is first reduced by scaling down, or normalizing, all gradient magnitudes. A greylevel *α_i-image_solid* is produced after the removal of edgels thresholded at an edge strength α_i from the image solid. The 3D connected components in the α_i-image_solid, if any, constitute the corresponding set of volume hypotheses. This process of obtaining independent sets of volume hypotheses from the various *α-image_solids* is called *α-partitioning*. We then apply *volume filtering* on each hypothesis and a simple Image-Solid-OR operation at the end, to verify the correctness of each hypothesis and to derive the α_{MIN}-set, respectively. For clarity of illustration, we introduce the *pseudo α-image_solid* that contains pattern(gray)-coded pixels instead of the actual greylevel values in a true α-image_solid. Examples of the pseudo α-image_solids are shown in Fig. 3.

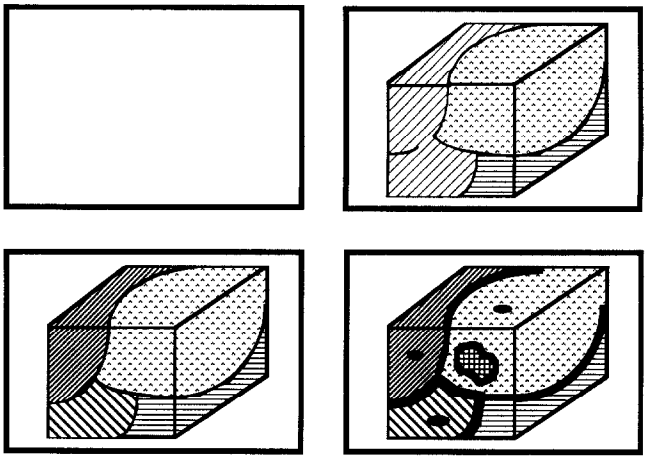

FIG. 3. Examples of pseudo α-image_solids. (upper left) α_1-image_solid, (upper right) α_2-image_solid, (lower left) α_3-image_solid, and (lower right) α_4-image_solid; where $\alpha_1 > \alpha_2 > \alpha_3 > \alpha_4$. Black pixels are edgels that have been removed.

3.2. *Volume Filtering for Detecting and Rejecting Gross Errors*

Not all volume hypotheses generated by the α-partitioning process are correct. The final partition can be obtained if we know how to determine the correctness of a hypothesized model for the given set of data. As a direct extension of our region filtering techniques in [19], the technique of *volume filtering* is used to verify the correctness of hypothesized models in an α-image_solid. Incorrect volume hypotheses are filtered out to produce a binary *filtered α-image_solid*. Each such image_solid is guaranteed to have a correct partition, even though the partition may be incomplete. Two example applications of a volume filter are shown in Fig. 4. The volume filter adopts the *lack-of-fit test* [26] to each volume hypothesis independently:

$$S^2 \leq \sigma^2.$$

S^2 is the unbiased estimate of the true measurement variance σ^2 when there is no lack of fit. The lack-of-fit test is commonly used in applied regression with the assumption of a *correct model* [26] and is defined as

$$S^2 = \frac{S(\hat{\Theta})}{n - p}, \tag{8}$$

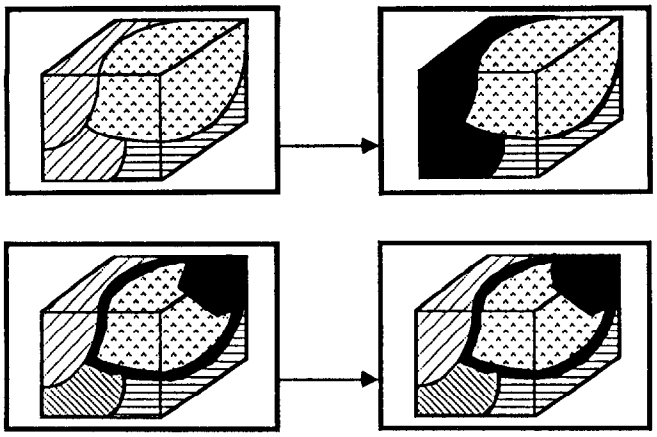

FIG. 4. Two examples illustrating the application of a volume filter. On the left are two pseudo α_image_solids. Edgels are labeled black, while the three volume hypotheses in the first example and four in the second example are patterncoded. On the right are two filtered α_image_solids where the incorrect volume hypothesis (shown in black) is eliminated.

where $S(\hat{\Theta})$ is the sum of squares for residuals, n the number of data points, and p the number of polynomial coefficients (see Appendix A).

In practice, the true measurement error is usually not directly available *a priori* in computer vision and we frequently deal with only a single image solid. Furthermore, σ^2 is unknown for volume V_l since the correctness of the fitted volume model f_{V_l} is supposed to be tested. So the maximum lack-of-fit error σ^2_{est} is estimated instead from the image solid noise variance estimate σ_{img} and the general lack-of-fit regression test has essentially become the so-called RMS fit error test [14]:

$$S^2 \leq \sigma^2_{est} = \omega\sigma_{img}. \tag{9}$$

3.3. *An Efficient Technique for Resolving Descriptions*

Multiple filtered a_image_solids are obtained after volume filtering, each of which is associated with one correct but possibly incomplete partition. In order to obtain the final segmented image solid, the α_image_solids can be resolved to produce the final α_{MIN}-set by simply applying an Image-Solid-OR operation to the various descriptions. This arises naturally from the processes of α-partitioning and volume filtering. We have shown the mathematics for the 2D image case in [19]. It can be shown similarly to the 2D case that an Image-Solid-Inclusive-OR operation on all filtered α_image_solids $\mathcal{F}_{\alpha_i}$ will produce a set of volumes V_l, each of which satisfies the smoothness predicate and is associated with a particular α_i (see Fig. 5). Therefore, the 3D image segmentation algorithm has effectively computed the α_{MIN}-set and generated a correct and complete segmentation of the image solid I.

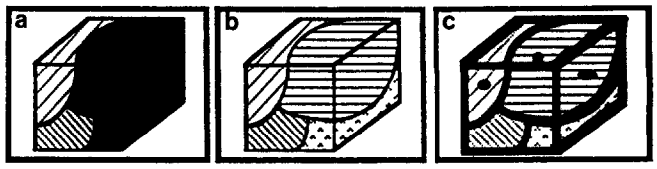

FIG. 5. Examples illustrating the relationships between α values and filtered α_image_solids. (a) $\mathcal{F}_{\alpha_1}$ (b) $\mathcal{F}_{\alpha_2}$ (c) $\mathcal{F}_{\alpha_3}$, where $\alpha_1 > \alpha_2 > \alpha_3$. In these filtered α_image_solids, correct volume hypotheses are labeled with different patterns while edgels and incorrect hypotheses are black. Note that there are two volumes in (a), four volumes in (b) and (c).

3.4. *Achieving Parallelism in Searching the α-Space*

In our 3D image segmentation algorithm, the α_{MIN}-set is derived by performing a search in the *α-space* (or edgeness measurement space). The α-set is either derived from the 3D image content (in our case) or supplied *a priori* somehow. Given this initial set, the algorithm then generates and prunes a search tree, according to the smoothness predicate, in the derivation of the α_{MIN}-set (Fig. 6). In the α-partitioning process, each set element α_i generates a hypothesized partition of the 3D image into volumes enclosed by discontinuities, such as $\{H_{\alpha_i,1}, H_{\alpha_i,2}, \cdots, H_{\alpha_i,a}\}$ in Fig. 6. Each volume hypothesis $H_{\alpha_i,j}$ in the α_i-image_solid is fitted with a MVO-order polynomial function $\hat{f}H_{\alpha_i,j}$. During volume filtering, any hypothesis

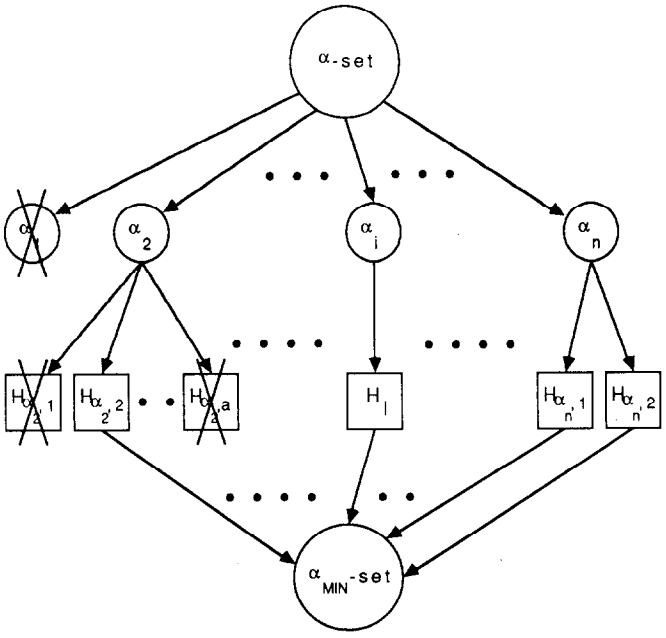

FIG. 6. Searching α-space for α_{MIN}-set. A box node represents a volume hypothesis and a circle node a set of volume hypotheses. Note that deleted nodes are marked with crosses.

that fails the lack-of-fit test is removed and the $H_{\alpha_i,j}$ node deleted (e.g. $H_{\alpha_2,1}$ and $H_{\alpha_2,d}$). It is conceivable that all hypotheses for a particular α_i are invalidated by the smoothness predicate and its associated subtree is thus pruned from the search (e.g. α_1). On the other hand, correct hypotheses in the filtered α_image_solids form the leaves of our search tree (e.g. $H_{\alpha_2,2}$). These are combined to produce the α_{MIN}-set.

Similarly to what we have shown for our 2D algorithm [19], the complexity of our 3D algorithm is also determined by the volume filtering where a least squares fit has to be applied to each volume. If a processor is assigned to the volume filter for each α_image_solid, it is of $O(k\ n^3)$ (based on standard Gaussian elimination), where n is the average number of data points in the fit and k the number of volume hypotheses. Since $k \ll n$, we have $O(n^3)$. The actual time and storage complexity of this implementation thus depends on

1. number of parallel processors utilized,
2. desired accuracy of the solution, and
3. input 3D image content, which determines the number of possible volume hypotheses.

4. IMPLEMENTATION ISSUES

The formulation described in the previous sections results in our new approach to the 3D image segmentation problem (Fig. 7). The 3D image segmentation algorithm allows us to partition an 3D image into correct volumes satisfying the smoothness predicate. We use the techniques of *α-partitioning* and *volume filtering* to pose-and-test various hypothesized edge strengths of boundary surfaces in parallel. Associated with every hypothesized edge strength α_i, we obtain an α_i_image_solid that is sent to a volume filter to remove volume hypotheses within which the smoothness variation is beyond the specified complexity. At the end of these parallel operations, all filtered α_image_solids are combined to obtain

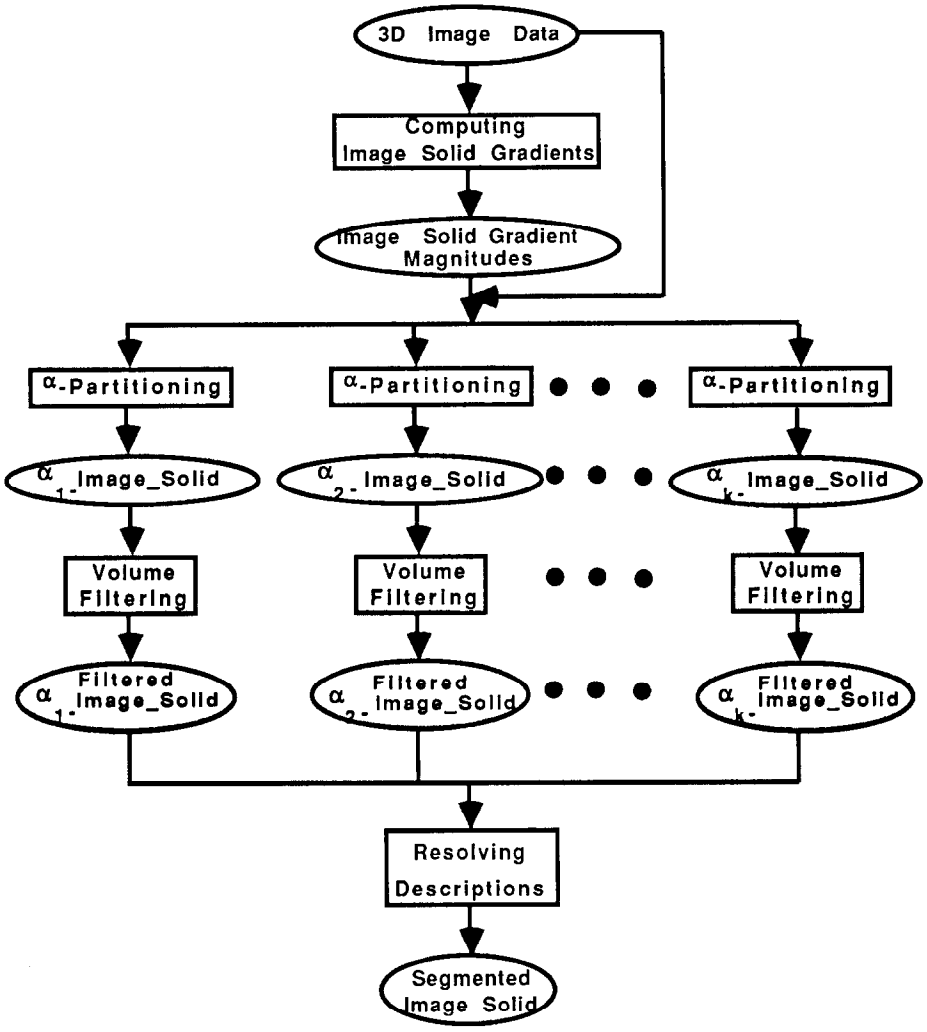

FIG. 7. The control diagram of the parallel 3D image segmentation algorithm.

the minimal set of edge strength thresholds α_{MIN}-set which is used to segment the 3D image.

4.1 *α-Partitioning*

The 3D Sobel edge detector was first applied in $3 \times 3 \times 3$ window neighborhoods to compute the gradient magnitudes on the $n \times n \times 6$ image solid. The *embedding angle* formula [10] is used to scale the gradient magnitudes M_{ijk},

$$\overline{M}_{ijk} = \cos^{-1}\left(\frac{1}{\sqrt{1 + M_{ijk}^2}}\right) \quad 1 \le i,j \le n \; 1 \le k \le 6.$$

The scaled gradient magnitudes $0 \le \overline{M}_{ijk} \le 90$ are stored into a three-dimensional array. The values above a given α_i are subtracted from the original image solid to produce the corresponding greylevel α_i_image_solid.

In our current implementation, the range where the α-set is searched for is chosen to be $\{90, 89, \cdots, 61\}$. The α's below 61 were experimentally determined to be not significantly meaningful for sampling the edgeness measurement space.

4.2. *Volume Filtering*

The volume filter determines all connected volumes in the given α_image_solid. Since the maximum complexity of a volume model f_{V_l} is defined by the unbiased estimate of the nonstationary mean $\hat{f}_{V_l}$ a MVO-order polynomial function, it suggests the following simple *least squares* regression procedure (see [19] for details). The greylevels $\mathbf{Y}$ in each volume V_l of the original image are then approximated by a 4th-order polynomial function $\mathbf{X}$ with regression coefficients $\boldsymbol{\Theta}$ so that the sum of squares of the errors $\mathbf{Y}$ is minimized. The resulting linear model can be written as

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\Theta} + \mathbf{Y}. \tag{10}$$

The volume filter than applies the lack-of-fit test [19] to examine whether or not this 4th-order polynomial fits the image data. The image noise level is estimated and compared with the estimated measurement error obtained from each volume fitting using the RMS fit error test [14]. If its lack-of-fit error is too large, the volume will fail this test and its corresponding area in the given α_image_solid is filtered out. This regression procedure and test are applied to all connected volumes in parallel, producing the binary filtered α_image_solid.

4.3. *Resolving Descriptions*

Thirty filtered α_image_solids (at $\alpha = 90, 89, \cdots, 61$) will be sent back and these binary image solids are then Inclusive-ORed [19] to produce the final segmented image solid.

5. EXPERIMENTAL RESULTS

Our algorithm has been applied successfully to a wide variety of 3D image data. Its performance on one medical 3D tomographic data set and two intensity image sequences is discussed in this section. The following set of figures is displayed for each input 3D image data:

- Original Gray Scale 3D Image of Six Frames
- Original Gray Scale 3D Image: 3D Solid (left) and 3D Sliced Solid (right)
- Gradient Magnitude 3D Image: 3D Solid (left) and 3D Sliced Solid (right)
- Gradient Magnitude 3D Image: 3D Solid Thresholded at Three Different Levels, Where White Pixels Are Those Above the Selected Threshold
- Segmented 3D Image Solid, Where Different Volumes Are Shown Separately

The segmented 3D image solids display the results produced by the 3D image segmentation algorithm. Each volume is an isolated set of connected pixels whose greylevel variation can be represented well by a triquartic polynomial function.

The following 3D image data have been used to test our proposed algorithm.

1. LIVER CT Image (UM Medical Center)
   This 3D data contained six 128×128 16-bit reconstructed cross section CT images obtained from our medical research center. We stack this sequence of cross sections into a $128 \times 128 \times 6$ array. Our algorithm was able to delineate the liver, but failed to catch the areas surrounding the liver and these areas are shown black in the segmented image solid, (Figs. 8–12).
2. ROAD Intensity Image Sequence (Martin Marietta)

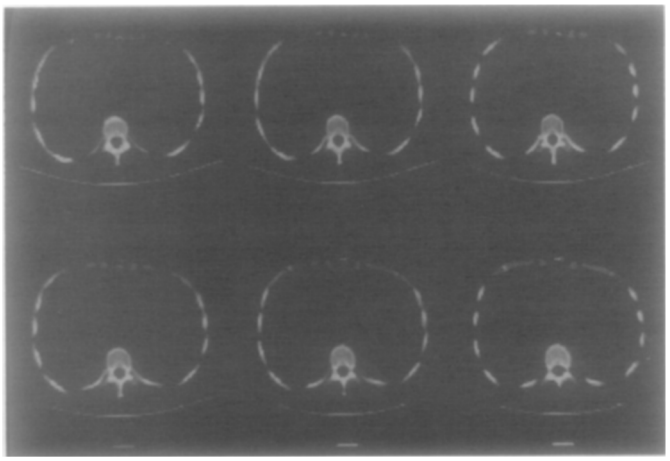

FIG. 8. Original grayscale 3D LIVER CT image of six frames.

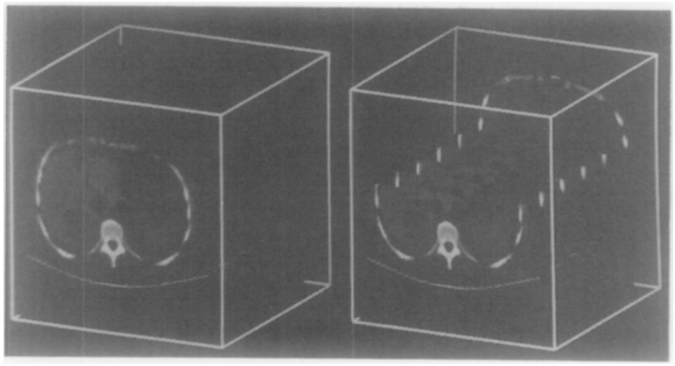

FIG. 9. Original grayscale 3D LIVER CT image: 3D solid (left) and 3D sliced solid (right).

This sequence was obtained from Martin Marietta and contained six 128 × 128 8-bit intensity images. In this sequence, the camera was mounted on a slowly moving vehicle. There are two moving cars in the scene. One car is on the far front of the camera and the other is passing by from the left. Since the camera is moving slowly, the motion between it and the background is not significant. The edge of the road and the edge of a passing car are all clearly delineated. The algorithm has trouble catching some of the highly textured background, which are also labeled black in the segmented image solid, (Figs. 13–17).

3. LAB Intensity Image Sequence (UM AI Laboratory)
This sequence was taken by a camera moving toward the center of the image and contained six 128 × 128 8-bit intensity images. In this sequence, the background is mostly empty and homogeneous and two blocks in the center are stationary and the one on the right is moving left. The toy dog on the left of the image is moving right. They are all on a table. They toy dog, three blocks, and the background are all clearly delineated. The algorithm has trouble catching some areas of the toy dog which are highly textured. These areas are labeled black in the segmented image solid (Figs. 18–22).

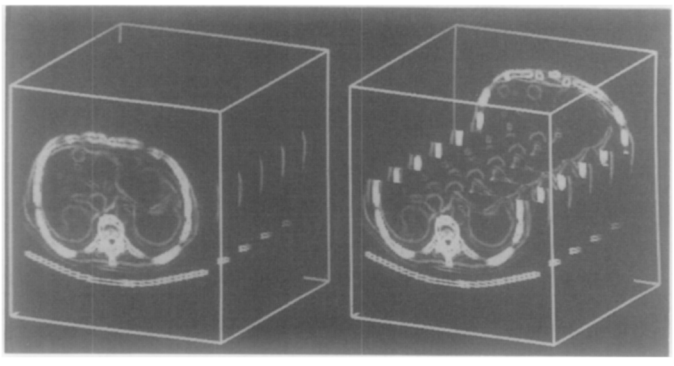

FIG. 10. Gradient magnitude 3D LIVER CT image: 3D solid (left) and 3D sliced solid (right).

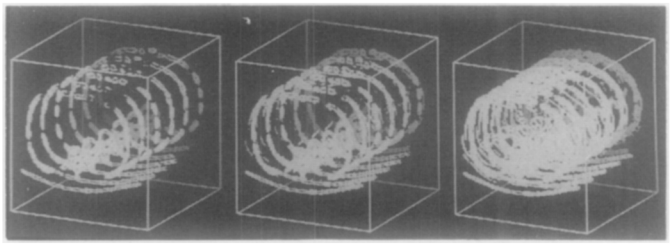

FIG. 11. Gradient magnitude 3D LIVER CT image: 3D solid thresholded at three different levels, where white pixels are those above the selected threshold.

6. CONCLUSIONS AND FUTURE RESEARCH

Three observations are evident from the experimental results. First, we have presented a very powerful segmentation scheme not only for 2D images [19] but also for 3D images as shown in this paper. The use of α-partitioning and volume filters enabled our 3D image segmentation algorithm to produces good segmentation results with tremendous savings in computation time. Further improvements might still be obtained by using surface refinement techniques [14] and developing more sophisticated 3D edge detectors as extensions of current 2D edge detectors [27–29].

Second, our segmentation algorithm depends heavily on how successful the 3D edge detector locates various volume hypotheses. When there is an abrupt change between neighboring frames, the 3D edge detector will usually locate a very thick discontinuity contour surface and thus result in a very small volume. The medical CT 3D image we experimented with causes such effects. How to reduce these effects is the central issue for our future research.

Finally, although our results show that confining the smoothness predicate to the 4th-order space (MVO = 4) is reasonable for real 3D images, there is a need to formulate a representation for highly textured volumes, other than increasing our complexity constraint to higher order arbitrarily. Based on our experience with several real 3D images, we believe that our regression model can be extended for textured images by relaxing the stationary autocovariance assumption and by using models to constrain the maximum complexity a covariance matrix can have. It would then be possible to develop a general 3D image segmentation algorithm that deals effectively with both regular and textured 3D images.

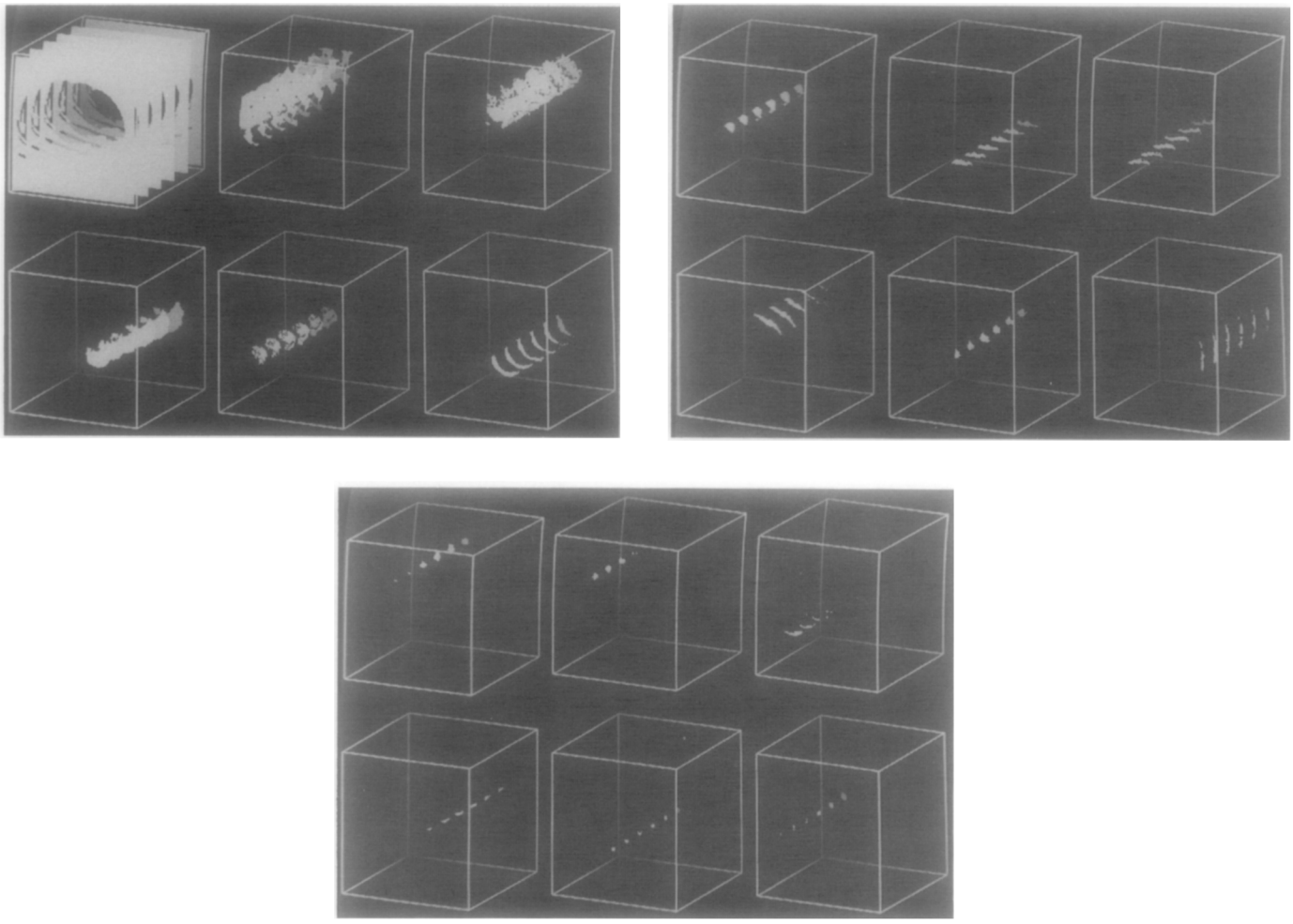

FIG. 12. Segmented 3D LIVER CT image solid, where different volumes are shown separately.

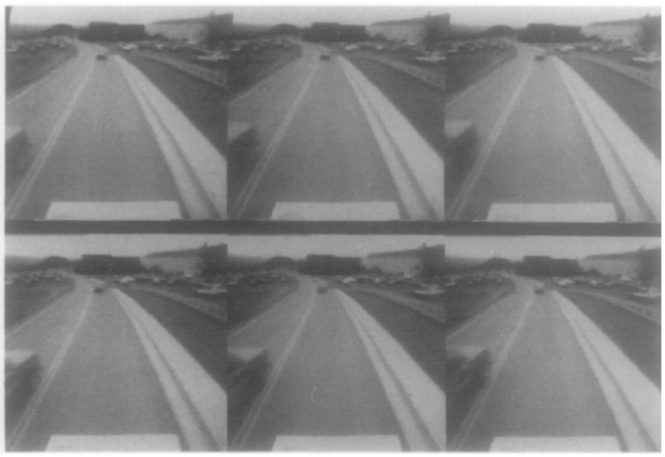

FIG. 13. Original grayscale spatio-temporal ROAD intensity image of six frames.

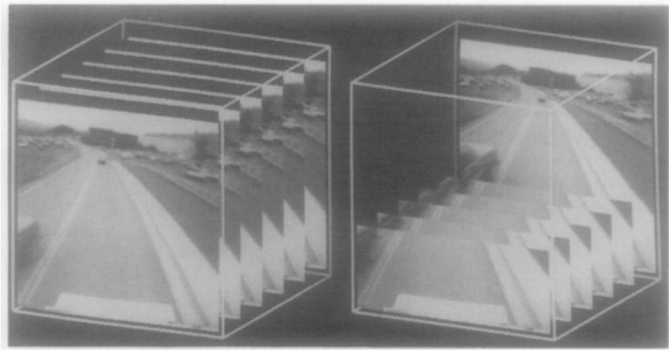

FIG. 14. Original grayscale spatio-temporal ROAD intensity image: 3D solid (left) and 3D sliced solid (right).

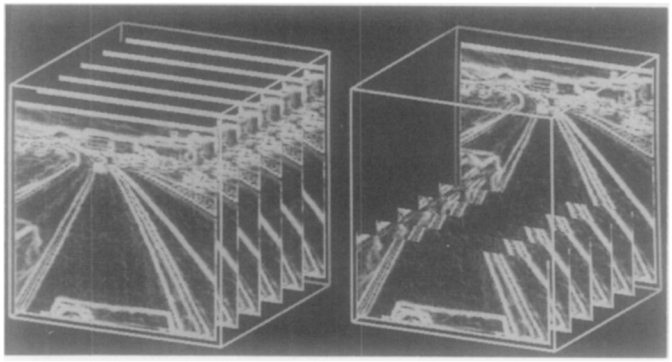

FIG. 15. Gradient magnitude spatio-temporal ROAD intensity image: 3D solid (left) and 3D sliced solid (right).

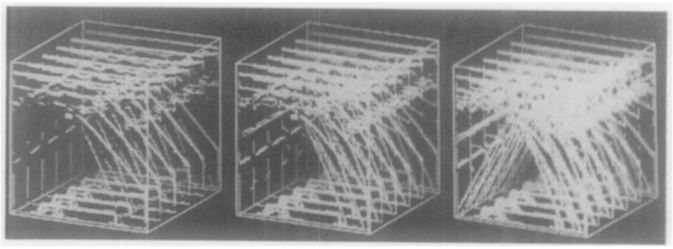

FIG. 16. Gradient magnitude spatio-temporal ROAD intensity image: three 3D solids thresholded at different levels, where white pixels are those above the selected threshold.

APPENDIX A. LINEAR LEAST SQUARES REGRESSION TECHNIQUES

In mathematical analysis of observational results, it is often necessary to extract the *best* or most plausible interpretation from the available data. This problem can be reduced to a formulation as follows. Given the observed (known) quantities Y_v and $X_{1v}, X_{2v}, \cdots, X_{pv}$, for $v = 1, 2, \cdots, n$, it is required to find values for the set of unknown quantities $\theta_1, \theta_2, \cdots, \theta_p$ in such a way that the set of appropriate equations

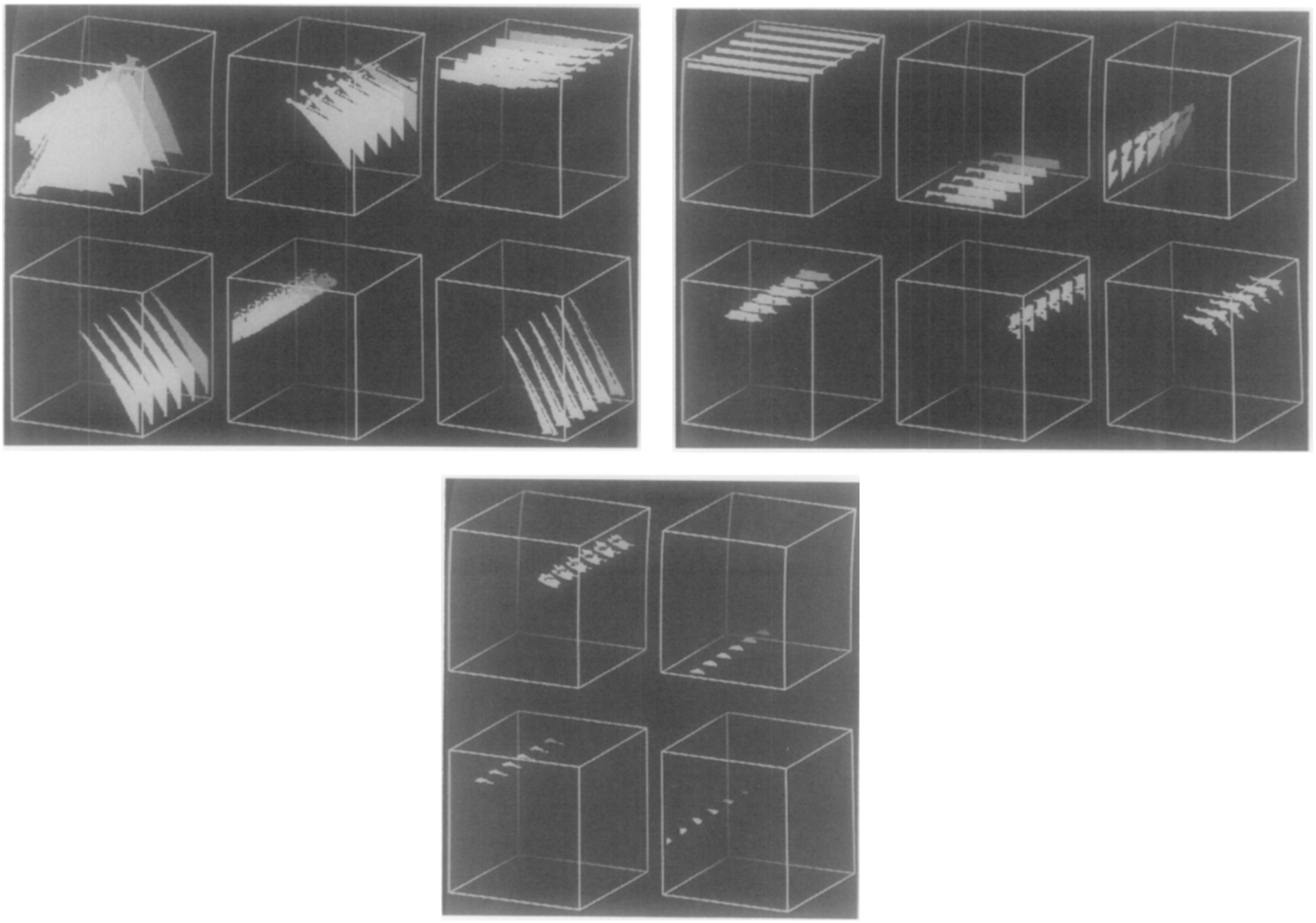

FIG. 17. Segmented spatio-temporal ROAD intensity image solid, where different volumes are shown separately.

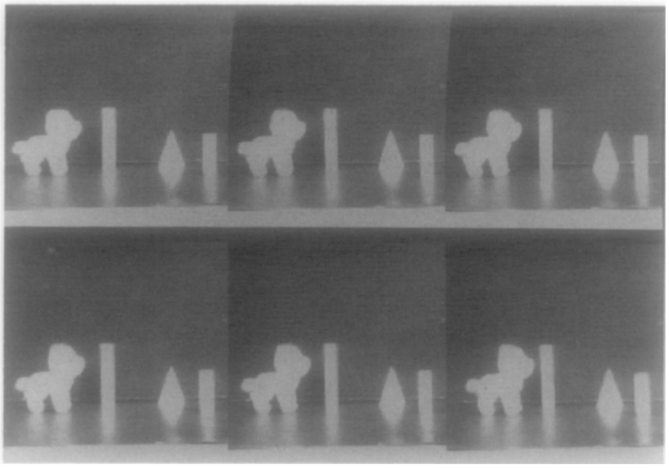

FIG. 18. Original grayscale spatio-temporal LAB intensity image of six frames.

$$
\begin{aligned}
Y_1 &\sim \theta_1 X_{11} + \theta_2 X_{21} + \cdots + \theta_p X_{p1} \\
Y_2 &\sim \theta_1 X_{12} + \theta_2 X_{22} + \cdots + \theta_p X_{p2} \\
&\cdots \\
Y_n &\sim \theta_1 X_{1n} + \theta_2 X_{2n} + \cdots + \theta_p X_{pn}
\end{aligned}
$$

are satisfied as nearly as possible, when the number n of equations is greater than the number p of unknowns.

By saying that the equations are to be satisfied *as nearly as possible*, we mean that the quantities

$$
\begin{aligned}
\varepsilon_1 &= Y_1 - (\theta_1 X_{11} + \theta_2 X_{21} + \cdots + \theta_p X_{p1}) \\
\varepsilon_2 &= Y_2 - (\theta_1 X_{12} + \theta_2 X_{22} + \cdots + \theta_p X_{p2}) \\
&\cdots \\
\varepsilon_n &= Y_n - (\theta_1 X_{1n} + \theta_2 X_{2n} + \cdots + \theta_p X_{pn})
\end{aligned}
$$

are to be as small as possible.

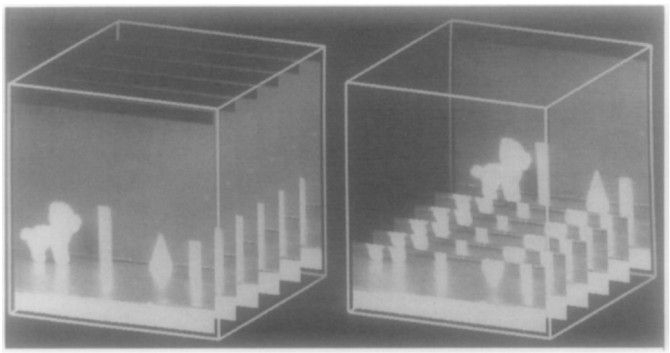

FIG. 19. Original grayscale spatio-temporal LAB intensity image: 3D solid (left) and 3D sliced solid (right).

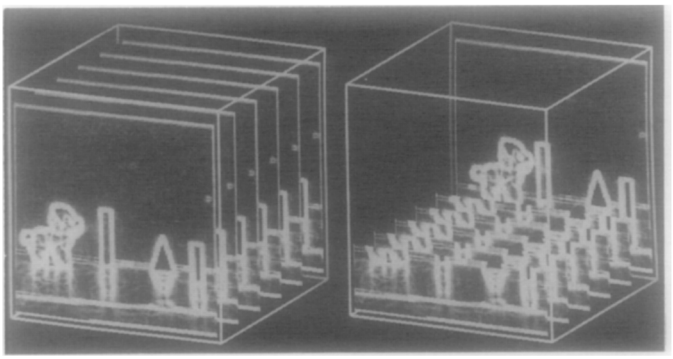

FIG. 20. Gradient magnitude spatio-temporal LAB intensity image: 3D solid (left) and 3D sliced solid (right).

Legendre [30] was the first to give a solution of this problem in terms of his *principle of least squares*. Of all possible sets of values of $\theta_1, \theta_2, \cdots, \theta_p$, the most satisfactory solution is that which makes the sum of squares of the errors a minimum; that is,

$$
\mathbf{Y} = \varepsilon_1^2 + \varepsilon_2^2 + \cdots + \varepsilon_n^2
$$

is desired to be a minimum. The resulting coefficients $\theta_1, \theta_2, \cdots, \theta_p$ are called the *regression coefficients*.

Assuming the principle of least squares, we now want to find the set of regression coefficients which can minimize the sum of squared errors given by

$$
\begin{aligned}
S(\boldsymbol{\Theta}) &= \sum_{v=1}^{n} \left[Y_v - \sum_{i=1}^{p} \theta_i X_{iv} \right]^2 \\
&= (\mathbf{Y} - \mathbf{X}\boldsymbol{\Theta})^{\mathrm{T}}(\mathbf{Y} - \mathbf{X}\boldsymbol{\Theta}) \\
&= \mathbf{Y}^{\mathrm{T}}\mathbf{Y} - 2\boldsymbol{\Theta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{Y} + \boldsymbol{\Theta}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\boldsymbol{\Theta}.
\end{aligned}
$$

By differentiating this expression with respect to $\boldsymbol{\Theta}$ and setting the result equal to $\mathbf{0}$, we obtain the normal equations

$$
\mathbf{X}^{\mathrm{T}}\mathbf{X}\hat{\boldsymbol{\Theta}} = \mathbf{X}^{\mathrm{T}}\mathbf{Y}.
$$

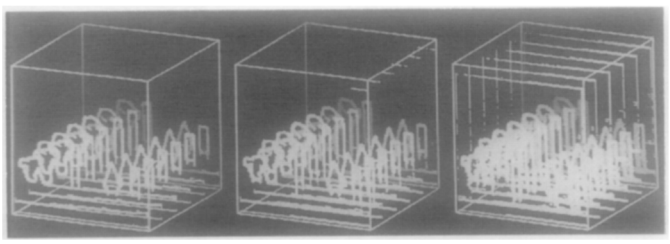

FIG. 21. Gradient magnitude spatio-temporal LAB intensity image: three 3D solids thresholded at different levels, where white pixels are those above the selected threshold.

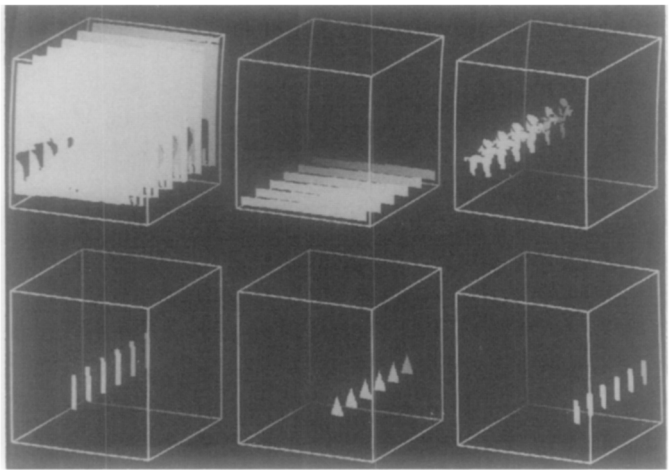

FIG. 22. Segmented spatio-temporal LAB intensity image solid, where different volumes are shown separately.

If $\mathbf{X}^T\mathbf{X}$ is nonsingular, the least squares regression coefficients can be obtained as

$$\hat{\Theta} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{11}$$

The sum of squares for residuals is then

$$S(\hat{\Theta}) = (\mathbf{Y} - \mathbf{X}\hat{\Theta})^T(\mathbf{Y} - \mathbf{X}\hat{\Theta}).$$

The method of least squares then gives the *optimal* estimates in the sense of being unbiased and having the smallest variance among all unbiased linear estimates. When the ε_v are independently and identically distributed as $N(0,\sigma^2)$, that is $\mathbf{Y} \sim N(\mathbf{0},\mathbf{I}\sigma^2)$, the estimates of the regression coefficients are the same as would be obtained from maximum likelihood method.

In summary, if the errors ε_v are independent and each follows the distribution $N(0,\sigma^2)$; that is, if $\mathbf{Y} \sim N(\mathbf{0},\mathbf{I}\sigma^2)$, then it can be shown that, if the model is correct, the following results are true [24]:

- $\hat{\Theta} \sim N[\Theta,(\mathbf{X}^T\mathbf{X})^{-1}\sigma^2]$
- $S(\hat{\Theta}) \sim \sigma^2\chi^2_{n-p}$
- $S(\Theta) - S(\hat{\Theta}) \sim \sigma^2\chi^2_{p,\lambda}$
- $S(\Theta) - S(\hat{\Theta})$ and $S(\hat{\Theta})$ are distributed independently

so that the ratio follows an F-distribution

$$\frac{[S(\Theta) - S(\hat{\Theta})]/p}{S(\hat{\Theta})/(n-p)} \sim F(p,n-p).$$

REFERENCES

1. E. Artzy, G. Frieder, and G. T. Herman, The theory, design, implementation and evaluation of a three-dimensional surface detection algorithm, *Comput. Graphics Image Process.* **15,** 1981, 1–24.
2. G. T. Herman and H. K. Liu, Dynamic boundary surface detection, *Comput. Graphics Image Process.* **7,** 1978, 130–138.
3. H. K. Liu, Two- and three-dimensional boundary detection, *Comput. Graphics Image Process.* **6,** 1977, 123–134.
4. D. G. Morgenthaler and A. Rosenfeld, Multidimensional edge detection by hypersurface fitting, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-3,** 1981, 482–486.
5. J. K. Udupa, S. N. Srihari, and G. T. Herman, Boundary detection in multidimensions, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-4,** 1982, 41–50.
6. E. H. Adelson and J. R. Bergen, Spatiotemporal energy models for the perception of motion, *J. Opt. Soc. Amer.* **2,** 1985, 284–299.
7. H. H. Baker, R. C. Bolles, and D. H. Marimont, "Generalizing epipolar-plane image analysis for non-orthogonal and varying view directions, *Proceedings of the Image Understanding Workshop, 1987,* pp. 843–848.
8. D. J. Fleet, A. D. Jepson, and P. E. Hallett, *A Spatio-Temporal Model for Early Visual Processing,* RCBV-TR-84-1, Research in Biological and Computational Vision, Univ. of Toronto, 1984.
9. D. J. Heeger, Optical flow from spatiotemporal filters, in *Proceedings of the First International Conference on Computer Vision, 1987,* pp. 181–190.
10. S.-P. Liou and R. Jain, Motion detection in spatio-temporal space, *Comput. Vision Graphics, Image Process.* **45,** 1989, 227–250.
11. A. B. Watson and A. J. Ahumada, Jr., Model of human visual-motion sensing, *J. Opt. Soc. Amer.,* **2,** 1985, 322–342.
12. R. Jain, *Environment Models and Information Assimilation,* Technical Report, IBM Almaden Research Laboratory, 1989.
13. S. W. Zucker and R. A. Hummel, A three-dimensional edge operator, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-3,** 1981, 324–331.
14. P. J. Besl and R. Jain, Segmentation through variable-order surface fitting, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-10,** 1988, 167–192.
15. R. M. Bolle and D. B. Cooper, Bayesian recognition of local 3-D shape by approximating image intensity functions with quadric polynomials, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-6,** 1984, 418–429.
16. R. C. Bolles and M. A. Fischler, "A RANSAC-based approach to model fitting and its application to finding cylinders in range data, *Proceedings of IJCAI, 1981,* pp. 637–643.
17. M. A. Fischler and R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Comm. ACM* **24,** 1981, 381–395.
18. T. Pavlidis, Segmentation of pictures and maps through functional approximation, *Comput. Graphics Image Process.* **1,** 1972, 360–372.
19. S.-P. Liou, Arnold H. Chiu, and R. Jain, A parallel technique for signal-level perceptual organization, *IEEE Trans. Pattern Anal. Mach. Intelligence,* **PAMI-13,** 1991.
20. N. I. Akhiezer, *The Classical Moment Problem and Some Related Questions in Analysis,* Olyver & Boyd, Edinburgh, 1965.
21. B. R. Hunt and T. M. Cannon, Non-stationary assumptions for Gaussian models of images", *IEEE Trans. Systems Man Cybernet.* **SMC-6,** 1976, 876–882.
22. H. C. Andrews and B. R. Hunt, *Digital Image Restoration,* Prentice-Hall, Englewood Cliffs, NJ, 1978.
23. B. R. Hunt, Non-stationary statistical image models (and their application to image data compression), *Comput. Graphics Image Process.* **12,** 1980, 173–186.

24. G. A. F. Seber, *Linear Regression Analysis,* Wiley, New York, 1977.

25. S. L. Horowitz and T. Pavlidis, Picture segmentation by a directed split–merge procedure, in *Proceedings, 2nd IJCPR, Copenhagen, Denmark, 1974,* pp. 424–433.

26. R. J. Brook and G. C. Arnold, *Applied Regression Analysis and Experimental Design,* Dekker, New York, 1985.

27. J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-8,** 1986, 679–698.

28. W. E. L. Grimson and T. Pavlidis, Discontinuity detection for visual surface reconstruction, *Comput. Vision Graphics Image Process.* **30,** 1985, 316–330.

29. R. M. Haralick, Digital step edges from zero crossing of second directional derivatives, *IEEE Trans. Pattern Anal. Mach. Intelligence* **PAMI-6,** 1984, 58–68.

30. Legendre, *Nouvelles Méthodes pour la Détermination des Orbites des Comètes,* Paris, 1806.