Methods for Numerical Integration of High-Dimensional Posterior Densities with Application to Statistical Image Models

Steven M. LaValle
Computer Science Department
Stanford University, Stanford, CA 94305
lavalle@cs.stanford.edu

Kenneth J. Moroney
Dept. of Electrical and Computer Engineering
University of Illinois, Urbana, IL 61801
moroney@cs.uiuc.edu

Seth A. Hutchinson
Dept. of Electrical and Computer Engineering
University of Illinois, Urbana, IL 61801
seth@uiuc.edu

Abstract

Numerical computation with Bayesian posterior densities has recently received much attention both in the applied statistics and image processing communities. This paper surveys previous literature and presents new, efficient methods for computing marginal density values for image models that have been widely considered in computer vision and image processing. The particular models chosen are a Markov random field formulation, implicit polynomial surface models, and parametric polynomial surface models. The computations can be used to make a variety of statistically-based decisions, such as assessing region homogeneity for segmentation, or performing model selection. Detailed descriptions of the methods are provided, along with demonstrative experiments on real imagery.

1 Introduction

Bayesian analysis has proven to be a powerful tool in many low-level computer vision and image processing applications; however, in many instances this tool is limited by computational requirements imposed by extracting information from high-dimensional probability spaces. In a standard application of Bayes' rule, an integral (or summation) is required to marginalize one set of the random variables with respect to another. This can be costly when the dimensions of the random variables are high, as is often the case with statistical image models (e.g., [12, 27, 48]).

High dimensionality of posteriors has led to the recent development of computation techniques that have increased the applicability of Bayesian analysis. For example, Gibbs sampling is a Markov chain-based technique that allows indirect sampling from (marginal) distributions, and has proven successful in image processing applications [23]. A recent discussion and comparison of Markov chain methods that use Monte-Carlo simulation, which includes the Gibbs sampler, appears in [5, 50]. Smith has provided a more general survey of Bayesian computation methods, including analytic approximations to the integrals, parametrizations and quadrature rules, and some adaptive sampling techniques [49].

In this paper, we present numerical methods for efficiently evaluating the marginalizing integrals for popular statistical image models, discuss applications, and present an empirical evaluation of the methods. We begin by introducing some notions that are common in a statistical image processing context (e.g., [23, 48, 54]). A vector $\mathbf{U}$ represents a continuous parameter space, and the vector $\mathbf{Y}$ represents the observations. These observations can be the image data, usually represented by $\mathbf{x}$, or some statistics of the image data. A noise (or degradation) model, $p(\mathbf{y}|\mathbf{u})$, represents the anticipated observation for a given parameter value. Finally, $p(\mathbf{u})$ represents a prior density on the parameter space.

Given these definitions, consider the marginalization of $\mathbf{y}$ with respect to $\mathbf{u}$:

$$\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathbf{du}. \tag{1}$$

It is assumed that both $p(\mathbf{y}|\mathbf{u})$ and $p(\mathbf{u})$ are easily identified such that the integrand of (1) is known. It is further assumed that $p(\mathbf{u}|\mathbf{y})$ and $p(\mathbf{y})$ are much more difficult to represent.

The need for efficient computation of (1) exists for many image processing applications. Con-

sider, for example, a Bayesian estimation context. Here, one is interested in selecting the $\mathbf{u}$ that maximizes the likelihood, $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$. This is done since by the application of Bayes' rule,

$$p(\mathbf{u}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{u})p(\mathbf{u})}{\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathbf{du}}, \tag{2}$$

$p(\mathbf{u}|\mathbf{y})$ is also maximized. By computing the denominator of (2), the equation can be directly used to obtain a normalized pdf value for a parameter, given the observations (i.e., $p(\mathbf{u}|\mathbf{y})$). Using this, comparisons can be made to the prior pdf values, $p(\mathbf{u})$.

Model order selection, a subject of interest in the computer vision community [6, 42], is another example. Of particular use for image segmentation, this subject addresses the problem of deciding which model, $\mathbf{U}'$ or $\mathbf{U}$, is appropriate for a given data set. The models are usually considered to be nested, $\mathbf{U}' \subseteq \mathbf{U}$. For example, $\mathbf{U}'$ could represent a linear model, and $\mathbf{U}$ a quadratic model. For two nested parameter spaces, the following ratio of marginals has been used extensively for Bayesian model selection [1, 12, 51]:

$$\frac{\int p(\mathbf{y}|\mathbf{u}')p(\mathbf{u}')\mathbf{du}'}{\int p(\mathbf{y}|\mathbf{u})p(\mathbf{u})\mathbf{du}}. \tag{3}$$

As (3) increases, confidence in $\mathbf{U}'$ also increases, favoring the simpler model.

A third application of the marginal computation (1) is found in the assessment of region homogeneity for image segmentation. For two subsets, R_1 and R_2, of an image, it has been shown that the ratio,

$$\frac{\left[\int p(\mathbf{y_1}|\mathbf{u_1})p(\mathbf{u_1})\mathbf{du_1}\right]\left[\int p(\mathbf{y_2}|\mathbf{u_2})p(\mathbf{u_2})\mathbf{du_2}\right]}{\int p(\mathbf{y_1}|\mathbf{u_{12}})p(\mathbf{y_2}|\mathbf{u_{12}})p(\mathbf{u_{12}})\mathbf{du_{12}}}, \tag{4}$$

can be used along with Bayes' rule to obtain the probability that the data in R_1 and R_2 were generated by the same parameter value (for some given parameter space) [36, 37]. The variable $\mathbf{u_{12}}$ refers to a combined parameter space that is associated with both R_1 and R_2.

The two ratios, (3) and (4), (and similar forms) have appeared recently in work from the statistics literature, and are termed *Bayes factors*. Smith and Speigelhalter used this ratio for model selection between nested linear parametric models [51]. Aitken has developed a Bayes factor for model comparison that conditions the prior model on the data [1]. Kass and Vaidyanathan present and discuss some asymptotic approximations and sensitivity to varying priors of the Bayes

factor [32]. The Bayes factor has also been carefully studied for evidence evaluation in a forensic science context [4, 15, 19, 20]. Other references to Bayes factors include [3, 24, 31].

In this paper, we introduce two new methods for efficient evaluation of (1). These integration techniques apply when the integrand of the marginalization (1) can be expressed in one of two forms: (a) as a function of a quadratic, or (b) as a function of a *ratio* of quadratics. In Section 2 we discuss existing, related integration methods, including certain limitations that make these methods insufficient for our needs. Section 3 discusses some popular statistical image models in which these two types of integrands appear.

Our integration methods that pertain to models in which $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$ is a function of a quadratic in $\mathbf{u}$ are discussed in Section 4. Section 4.1 discusses a technique which is based on the idea that the integrand is asymptotically Gaussian in $\mathbf{u}$. Since we are interested in techniques that can handle large amounts of uncertainty, this technique is shown to be most useful when the integrand is *directly* Gaussian in $\mathbf{u}$. Section 4.2 introduces a more general technique that creates large computational savings by efficiently mapping an N-variate integration space into a single dimension. The marginalization (1) can then be computed by traditional one-dimensional integration means, regardless of the original dimension of integration. The parametric polynomial model (Section 3.1) and a Markov random field model (Section 3.2) are examples of models to which these methods apply.

In Section 5, we present a new Monte Carlo-based integration method that applies to models in which $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$ is a function of a ratio of quadratics. This technique defines an importance sampling function for this class, which significantly reduces the number of samples needed. An example of a model to which this technique applies is the implicit surface model, discussed in Section 3.3.

In Section 6 we show some segmentation results that were obtained using the models in Section 3. These results depend heavily on the integration techniques presented in this paper. Also shown are graphical depictions of the computational savings yielded by the method in Section 5. Finally, some conclusions are presented in Section 7.

2 Related Integration Methods

Numerical methods for integration have been a topic of research for many years, and a number of methods have been developed. Of these, several methods may seem plausible for image processing applications; however, the complexity of a typical statistical image model can cause them to be inappropriate. In this section, we explore these methods and discuss the limitations of each that make them inappropriate for our needs. Basic integration techniques such as classical quadrature and basic Monte Carlo are discussed in sections 2.1 and 2.2. Integration approaches that have appeared in statistical contexts are discussed in Sections 2.3 and 2.4, which are asymptotic approximation and Gibbs sampling, respectively.

2.1 Quadrature

One straightforward approach to many integration problems is the use of classical quadrature. In general, a quadrature formula can be expressed as

$$\int f(x) \approx \sum_{k=1}^{n} A_k f(x_k), \tag{5}$$

in which the integral is approximated by a linear combination of values of the function. Three main concerns of this method are determining the weights A_k, the partitioning of the region of integration, and the number, n, of sample points.

It is well known that the number of sample points needed for a certain degree of accuracy increases rapidly as the dimension of integration increases (for example, see [34]). If the dimension of integration is low (e.g., 3 or less), this method can produce accurate approximations to an integral with reasonable cost; however, since statistical image models are often of high dimension, this rapid increase in the number of samples makes the quadrature approach computationally prohibitive.

2.2 Monte Carlo Integration and Importance Sampling

Monte Carlo integration, in general, is a technique that is often suitable for high dimensional integration. For a complete introduction to Monte-Carlo integration, see [29]. The basic Monte Carlo method iteratively approximates a definite integral by uniformly sampling from the domain of integration, and averaging the function values at the samples. The integrand is treated as a

random variable, and the sampling/averaging scheme yields a parameter estimate of the mean, or expected value of the random variable.

Although the number of samples required for a certain degree of accuracy does not depend on the dimension of integration, there are two limitations to the basic Monte Carlo approach: 1) the accuracy improves only linearly with the number of samples, and 2) more samples are needed if the integrand is peaked in a small region and approximately zero elsewhere [29]. More elaborate schemes with faster convergence rates are discussed in [57]; however, improvement in the convergence rate for these methods is possible only for low-dimensional cases (e.g., 3 or less). These approaches typically incorporate some form of quadrature, yielding greater computational cost.

The second limitation is of particular concern in a statistical context. As the amount of information contained in a posterior density increases, the integrand (1) becomes peaked. For example, suppose an image is presented in which all of the pixels are known to have some fixed, unknown real-valued intensity, corrupted by additive Gaussian iid noise with known variance. The random variable $\mathbf{U}$ can represent the fixed, underlying intensity value, and $\mathbf{Y}$ can represent a vector of observed image data. For a given observation $\mathbf{y}$, through Bayes' rule $p(\mathbf{u}|\mathbf{y})$ is proportional to $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$. As the number of data points increases, our ability to predict $\mathbf{u}$ increases since $p(\mathbf{u}|\mathbf{y})$ becomes peaked. By the proportionality, the integrand of (1) also becomes peaked. This same type of behavior occurs with the models discussed in this paper.

One common method for handling a peaked integrand is to introduce *importance sampling* into the Monte Carlo integration [29]. Rather than sampling uniformly from the domain of integration, the samples are concentrated in the region in which the integrand peaks. The samples are appropriately weighted in the resulting average to compensate for the nonuniform distribution of sample points. While this general technique is widely used in statistical computations, each type of integrand requires a unique importance function. This importance function defines the area in which the integral is peaked, and is crucial for importance sampling to succeed. In Section 5, we introduce one such importance function defined for models that can be expressed as a function of a ratio of quadratics.

2.3 Asymptotic Approximation

As mentioned in Section 2.2, in an image processing application, the integrand of (1) can become peaked. This observation has led to the use of asymptotic approximations when the number of image elements is large [9, 48]. If the models are expressed with smooth probability densities, then it can be shown that the integrand of (1) is approximately Gaussian as the number of samples increases (becoming a delta function as the number of samples reaches infinity). The integral is directly determined by integrating the approximating Gaussian.

In some applications this has led to useful results; however, in general we are interested in statistical methods that are capable of handling a greater degree of uncertainty. For instance, in a region-based segmentation scheme [38, 46], the number of points in a region can vary dramatically. Smaller regions will have a greater degree of uncertainty associated with them, which leads poorer accuracy in the asymptotic approximation.

2.4 Gibbs Sampling

The Gibbs sampler was introduced to the image processing community by Geman and Geman [23], and is described in detail in [13]. Here we describe a brief overview of the method. Assume that we have n random variables $(\mathbf{X}_1 \ldots \mathbf{X}_n)$. The availability of the n full conditionals of the form $p(\mathbf{x}_i|\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_n)$ is essential to the applicability of this method. Given these, this method allows us to create samples of the marginal densities by iteratively extracting random samples from the n full conditionals. The algorithm is initialized by selecting arbitrary values for the random variables, $(\mathbf{x}_1^0, \ldots, \mathbf{x}_n^0)$. Samples are then extracted as follows:

$$\begin{array}{rcl} \mathbf{x}_1^i & \sim & p(\mathbf{x}_1|\mathbf{x}_2^{i-1}, \ldots, \mathbf{x}_n^{i-1}) \\ \mathbf{x}_2^i & \sim & p(\mathbf{x}_2|\mathbf{x}_1^i, \mathbf{x}_3^{i-1}, \ldots, \mathbf{x}_n^{i-1}) \\ \cdot & & \\ \cdot & & \\ \cdot & & \\ \mathbf{x}_n^i & \sim & p(\mathbf{x}_n|\mathbf{x}_1^i, \ldots, \mathbf{x}_{n-1}^i) \end{array} \tag{6}$$

where $i = (1, \ldots k)$. At the end of k iterations, we have $(\mathbf{x}_1^k \ldots \mathbf{x}_n^k)$. Assuming that k is large enough, these represent single samples from the n random variables. Thus, this process can be repeated, say, m times to create m samples of the marginal distributions, $p(\mathbf{x}_1) \ldots p(\mathbf{x}_n)$. The Gibbs sampler can therefore be used to estimate any marginal distribution [43]. Given m samples,

the distribution of the jth variable can be estimated as

$$\hat{p}(\mathbf{x}_j) = \frac{1}{m}\sum_{l=1}^{m} p(\mathbf{x}_{jl}^k | \mathbf{x}_{rl}^k, r \neq j). \tag{7}$$

In the case at hand, the reader is reminded that the marginalization in (1) is the desired result. For the statistical models discussed in Section 3, densities are given to form the integrand, $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$. Hence, of the two random variables discussed in Section 1, $\mathbf{U}$ and $\mathbf{Y}$, only one conditional is available. Since the Gibbs sampler would also call for the availability of the conditional, $p(\mathbf{u}|\mathbf{y})$, it is inappropriate for this problem.

3 Image Model Applications

We will present our general methods of numerical integration in Sections 4 and 5. In this section we describe examples of image models to which these methods apply. For each application, sufficient information is given to form the integrand of (1), $p(\mathbf{y}|\mathbf{u})p(\mathbf{u})$. In each section we refer to a set of image elements as R, which could be a set of intensities or range coordinates, depending on the image type.

3.1 Parametric (Explicit) Polynomial Models

The general form of the parametric polynomial model is

$$\psi(\mathbf{u}; i, j) = u_1 + \sum_{m=2}^{N} u_m i^{a_m} j^{b_m}, \tag{8}$$

in which a_m and b_m are positive integers. The *degree* of the model is the maximum over m of $a_m + b_m$. The parameter space is thus spanned by the coefficients that are usually selected for surface estimation. This image model has been used for the segmentation of intensity images in [26], [39], and [48], and for range image segmentation in [6], [39], and [46]. A survey and discussion of parametric polynomial surface estimation that is based on calculus of variations is presented in [10], and a survey of early facet-model research and other intensity image segmentation techniques is presented in [25].

The observations, $\mathbf{Y}$, are represented by a vector of point-to-surface displacements of the image values (either range or intensity) in R, given a parameter value $\mathbf{u}$. We denote a single displacement as

$$\delta(x[i, j], \mathbf{u}) = x[i, j] - \psi(\mathbf{u}; i, j) \tag{9}$$

in which $x[i,j]$ is the image value at the i^{th} row and j^{th} column. The dimension of $\mathbf{Y}$ is equal to the number of pixels in R.

If we assume an additive Gaussian iid zero-mean noise model, the joint density is obtained by taking the product of the individual displacement densities:

$$p(\mathbf{y}|\mathbf{u}) = (2\pi\sigma^2)^{\frac{-|R|}{2}} exp\left\{\frac{-1}{2\sigma^2}\sum_{x[i,j]\in R}[\delta(x[i,j],\mathbf{u})]^2\right\}. \tag{10}$$

We define the prior model by assigning a uniform density to a bounded parameter space. For regions that we have considered, a rectangular portion of the parameter space can always be identified that encloses nearly all of the probability mass that contributes to the integrals in (4), and using the integration method of Section 4.2, we are actually not required to specify bounds to perform the integration (all of $\Re^N$ is used). The problem of selecting bounds for a uniform prior has been known to lead to difficulty in Bayesian analysis, and is referred to as Lindley's paradox [40]. As the volume over which the uniform density is defined increases, the ratio (4) decreases.

3.2 A Markov Random Field Model

We use the MRF formulation introduced in [30]. This model has been applied to texture segmentation of intensity images in [17, 48], and has been recently extended to texture modeling and segmentation of color images [44].

An image element represents a single intensity, $X[i,j]$, treated as a random variable. We have an N-dimensional parameter space, which represents the interaction of a pixel with a local set of neighboring pixels. The *order* of an MRF indicates the size of the local neighborhood that is considered. Figure 1 shows the neighbor set that is used for the MRF orders considered in our experiments.

We use μ and σ^2 to represent the mean and variance in R, respectively. For any general order of MRF interactions, the image element of the l^{th} parameter interaction is denoted by $T_l(x)$. Hence, in general at some point $X[i,j] = x$, the model is

$$x - \mu = \sum_{l=1}^{N} u_l(T_l(x) - \mu). \tag{11}$$

We could also consider μ as part of the parameter space. This would require the selection of appropriate prior density, $p(\mu)$, and require it to be integrated in (4).

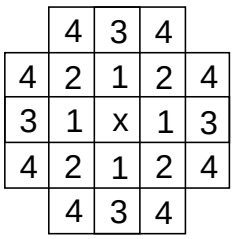


Figure 1: The MRF pixel neighborhood with $X[i,j]$ located in the center. For an n^{th}-order MRF, the pixels in boxes with numbers less than or equal to n comprise the neighborhood.

The observation space, $\mathbf{Y}$, is defined as a vector that corresponds to all of the intensity data, $x[i,j]$, in some region R. Hence, the dimension of $\mathbf{Y}$ is equal to the number of pixels in R.

We assume that the noise process that occurs in the linear prediction (11) is Gaussian. The joint density that we use over the points in R is not a proper pdf; however, it has been considered as a reasonable approximation and used in previous segmentation schemes [17, 44, 48].

We obtain the complete noise model by taking the product of the density expressions over each of the individual pixels:

$$p(\mathbf{y}|\mathbf{u}) = (2\pi\sigma^2)^{\frac{-|R|}{2}} exp\left\{\frac{-1}{2\sigma^2}\sum_{x\in R}\left[x-\mu-\sum_{l=1}^{N}u_l(T_l(x)-\mu)\right]^2\right\}. \tag{12}$$

For the texture model we also use a uniform prior density on a bounded parameter space.

3.3 Implicit Polynomial Models

For this model each image element represents a point in $\Re^3$, specified by $[x_1, x_2, x_3]$ coordinates, which we denote by $\mathbf{x}$. An implicit polynomial equation is represented as

$$\phi(\cdot, \mathbf{u}) \equiv \sum_{j=1}^{N} u_j x_1^{a_j} x_2^{b_j} x_3^{c_j} = 0 \tag{13}$$

with

$$a_N = b_N = c_N = 0. \tag{14}$$

The constants a_j, b_j, and c_j are positive integers, representing the exponents of each variable. The $\cdot$ used here indicates that we have an implicit function with $\mathbf{x}$ as the variables. We will later refer to $\phi(\mathbf{x}, \mathbf{u})$, which yields a nonzero value unless $\mathbf{x}$ is on the surface. The *degree* of the polynomial model

is the maximum over j of $a_j + b_j + c_j$. This model has been used for range image segmentation in [9], [21], [55], [56], and [53]. It has been used for object recognition applications in [16] and [33].

With the formulation given by (13), there are redundant representations of the solution sets (i.e., there are many parameter vectors that describe the same surface in $\Re^3$). It is profitable to choose some restriction of the parameter space that facilitates the integrations in (4), but maintains full expressive power. We use the constraints $\|\mathbf{u}\| = 1$ and $u_1 > 0$, to constrain the parameter space to a half-hypersphere.

The observation considered here is a function of the signed distances of the points $\mathbf{x} \in R$ from the surface determined by $\mathbf{u}$, termed *displacements*. Define $\delta(\mathbf{x}, \phi(\cdot, \mathbf{u}))$ to be the displacement of the point $\mathbf{x}$ to the surface described by the zero set $\{\mathbf{x} : \phi(\mathbf{x}, \mathbf{u}) = 0\}$. The function $\delta(\mathbf{x}, \phi(\cdot, \mathbf{u}))$ takes on negative values on one side of the surface and positive on the other.

We consider the following observation space definition, and others are mentioned in [36]:

$$y(R, \mathbf{u}) = \sum_{\mathbf{x} \in R} [\delta(\mathbf{x}, \phi(\cdot, \mathbf{u}))]^2. \tag{15}$$

Note that we use y instead of $\mathbf{y}$ when the observation is a scalar.

Although we have defined the observation space in terms of the displacements, a closed-form expression for the displacement of a point to a polynomial surface does not exist in general. We use a displacement estimate presented in [56]:

$$\hat{\delta}(\mathbf{x}, \phi(\cdot, \mathbf{u})) = \frac{\phi(\mathbf{x}, \mathbf{u})}{\|\nabla_x \phi(\cdot, \mathbf{u})\|}. \tag{16}$$

To define the noise model, we express the density corresponding to the displacement of an observed point from a given surface. We use a probability model for range-scanning error used and justified in [9]. The model asserts that the density, $p(\delta|\mathbf{u})$, of the displacement of an observed point from the surface, $\phi(\cdot, \mathbf{u})$, is a Gaussian random variable with zero mean and some known variance, σ^2.

Since taking the sum of squares of Gaussian densities yields the chi-square density, the density using (15) is

$$p(y|\mathbf{u}) = \chi^2_m(y) = \frac{1}{2^{m/2}\Gamma(m/2)} y^{m/2-1} e^{-y/2}. \tag{17}$$

Here y is the sum-of-squares for a given region, R, and parameter value $\mathbf{u}$, given by (15). Also, $\Gamma(\cdot)$ is the standard gamma function and $m = |R|$ (the number of elements in R).

We assign $p(\mathbf{u})$ to be a uniform prior on the constrained parameter space.

The method that we will discuss in Section 5 applies to integrals in which the integrand is a function of a quadratic ratio. We will now show that the model discussed above can be approximated by such a function.

Here we consider the case of evaluating the integral (1) for some region R. Shown explicitly, the computation of interest is

$$\int \chi^2_m(y(R,\mathbf{u}))p(\mathbf{u})\mathbf{du} \tag{18}$$

where $p(\mathbf{u})$ is assumed to be a uniform prior density, and hence does not affect the integration. Using the displacement estimate (16), the argument of the integrand is

$$y(R,\mathbf{u}) = \sum_{\mathbf{x}\in R}[\delta(\mathbf{x},\phi(\cdot,\mathbf{u}))]^2 \approx \sum_{\mathbf{x}\in R}\frac{[\phi(\mathbf{x},\mathbf{u})]^2}{\|\nabla_x\phi(\cdot,\mathbf{u})\|^2}. \tag{19}$$

Based on the need for computational efficiency, we borrow a simplification used by Taubin and Cooper [56]. In their work, the simplification was performed to facilitate optimization for the purpose of parameter estimation of implicit surfaces. This simplification makes the assumption that the magnitude of the gradient remains fairly constant over the set of points, for a given parameter value. Using this, (19) can be rewritten with a numerator summation and a denominator summation. Since their definition for the parameter space coincides with our parameter space, this simplification is equally valid for our work. This simplification thus yields

$$\sum_{\mathbf{x}\in R}\frac{[\phi(\mathbf{x},\mathbf{u})]^2}{\|\nabla_x\phi(\cdot,\mathbf{u})\|^2} \approx \frac{\displaystyle\sum_{\mathbf{x}\in R}[\phi(\mathbf{x},\mathbf{u})]^2}{\displaystyle\frac{1}{|R|}\sum_{\mathbf{x}\in R}\|\nabla_x\phi(\cdot,\mathbf{u})\|^2}. \tag{20}$$

Recall that $\phi(\cdot,\mathbf{u})$ is linear in the parameters $\mathbf{u}$. The numerator above must be a quadratic function in the parameter value, since it is a linear function squared. The denominator is also quadratic, since the gradient yields a linear function of $\mathbf{u}$, and the magnitude squared yields a quadratic. The numerator and denominator represent sums of quadratics, and hence are in turn quadratic. From this, they can each be expressed as an $(N+1)\times(N+1)$ quadratic form which gives

$$y(R,\mathbf{u}) \approx \frac{\mathbf{u}^T M_h \mathbf{u}}{\mathbf{u}^T Q_h \mathbf{u}}. \tag{21}$$

Thus, for this model, the integrand of (1) could be expressed as

$$f(\frac{\mathbf{u}^T M \mathbf{u}}{\mathbf{u}^T Q \mathbf{u}}), \tag{22}$$

in which M and Q are positive definite symmetric matrices, and f represents the chi-square density. This is the form that will be investigated in Section 5.

4 Integration of an N-variate Function of a Quadratic

In this section, we consider an integral of the form

$$\int f(g(\mathbf{u}))d\mathbf{u}, \tag{23}$$

in which $\mathbf{U} = \Re^N$ and g is a scalar, real-valued quadratic function

$$w \equiv g(\mathbf{u}) = \mathbf{u}^T M\mathbf{u} + b\mathbf{u} + c, \tag{24}$$

and f is a positive continuous function. Examples of models that provide the integrand of integrals of this type are the parametric polynomial model and Markov Random Field model, discussed respectively in Sections 3.1 and 3.2.

We discuss two methods that can be used to compute the integral, (23). The first approach can be utilized when the noise model is Gaussian, and the prior model is uniform. Also discussed here is an asymptotic approximation for use with large data sets, as done in [9, 48]. The second approach is efficient and more general, which numerically performs a Lebesque integration on an ellipsoidal decomposition of the parameter space.

4.1 Utilizing a Gaussian Assumption

Suppose that in particular, the noise model is Gaussian and the prior model is uniform, as is precisely the case for the models of Sections 3.1 and 3.2. The joint density, $p(\mathbf{y}|\mathbf{u})$, would then be Gaussian in $\mathbf{y}$ and the integral in question would be of the form

$$K_1 \int exp\left\{\mathbf{u}^T M\mathbf{u} + b\mathbf{u} + c\right\} d\mathbf{u} \tag{25}$$

in which

$$K_1 = (2\pi\sigma^2)^{\frac{-|R|}{2}} \tag{26}$$

and $|R|$ is the size of the data set over which the models are defined. By completing the square in the integrand, the integral becomes

$$K_1 \int exp\left\{-\frac{1}{2}(\mathbf{u}-\mu)^T M(\mathbf{u}-\mu) + K_2\right\} d\mathbf{u} \tag{27}$$

in which M is known to be the inverse of the covariance matrix, K_2 is some constant, and μ represents the mean vector.[1] This can then be factored further to produce

$$K_1\Big[(2\pi)^N|M^{-1}|\Big]^{1/2}exp\{K_2\}\int\Big[(2\pi)^N|M^{-1}|\Big]^{-1/2}exp\left\{-\frac{1}{2}(\mathbf{u}-\mu)^T M(\mathbf{u}-\mu)\right\}d\mathbf{u}. \tag{28}$$

From here we see that the integral in question, (25), is directly evaluated by

$$K_1\Big[(2\pi)^N|M^{-1}|\Big]^{1/2}exp\{K_2\} \tag{29}$$

since the integral in (28) is of the Gaussian and evaluates to 1. Further, by evaluating the integrand of (27) at $\mathbf{u}=\hat{\mathbf{u}}$, the maximum likelihood estimate of the true $\mathbf{u}$, $(\mathbf{u}-\mu)$ is minimized and the constant $exp\{K_2\}$ can be found. Thus it is found that

$$K_1\int exp\Big\{\mathbf{u}^T M\mathbf{u}+b\mathbf{u}+c\Big\}d\mathbf{u}=K_1\Big[(2\pi)^N|M^{-1}|\Big]^{1/2}\Big[exp\Big\{\mathbf{u}^T M\mathbf{u}+b\mathbf{u}+c\Big\}\Big]\Big|_{\mathbf{u}=\hat{\mathbf{u}}} \tag{30}$$

Now, suppose the noise model is an arbitrary pdf common to all elements of $\mathbf{Y}$, a vector of $|R|$ iid random variables. Thus the joint pdf over $\mathbf{Y}$, $p(\mathbf{y}|\mathbf{u})$, is found by

$$\prod_{m=1}^{|R|}p(\mathbf{y}_m|\mathbf{u}). \tag{31}$$

In a paper by Bolle and Cooper[9], it was shown that for a "reasonably smooth" *a priori* pdf $p(\mathbf{u})$, the integral

$$\int\left[\prod_{m=1}^{|R|}p(\mathbf{y}_m|\mathbf{u})\right]p(\mathbf{u})d\mathbf{u} \tag{32}$$

for large $|R|$ is approximately

$$(2\pi)^{N/2}|\Psi(\mathbf{Y},\hat{\mathbf{u}})|^{-1/2}\left[\prod_{m=1}^{|R|}p(\mathbf{y}_m|\hat{\mathbf{u}})\right]p(\hat{\mathbf{u}}) \tag{33}$$

where N is the dimension of $\mathbf{u}$, $|\cdot|$ denotes determinant and $\hat{\mathbf{u}}$ is the maximum likelihood estimate of the true $\mathbf{u}$. $\Psi(\mathbf{Y},\hat{\mathbf{u}})$ is a matrix with ijth element

$$-\frac{\partial^2}{\partial u_j\partial u_i}log\prod_{m=1}^{|R|}p(\mathbf{y}_m|\mathbf{u})|_{\mathbf{u}=\hat{\mathbf{u}}}. \tag{34}$$

This was based on an earlier result that stated that (31) is asymptotically Gaussian in $\mathbf{u}$.

[1] In some cases, a valid covariance matrix may not be positive definite. In practice, this usually corresponds to situations in which there are very few data points.

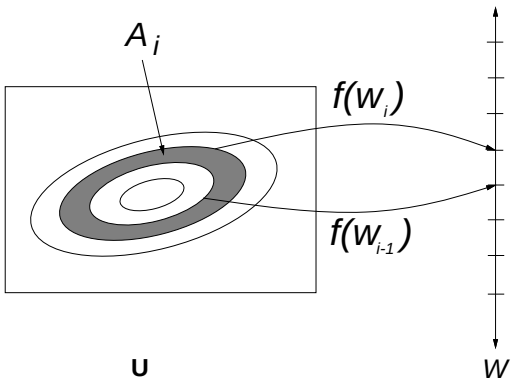


Figure 2: Decomposing the parameter space into concentric ellipsoids.

4.2 Using a Lebesgue Integration approach

The method discussed above provides an accurate and efficient solution to (23) even with high uncertainty. It does this with one restriction: that the integrand be Gaussian in the parameter space. In this section, we introduce a more general method that removes this restriction with no loss of accuracy or efficiency.

Here, we transform the N-variate integral in (23) into a single integral by decomposing $\mathbf{U}$ into subsets on which $f(g(\mathbf{u}))$ is approximately constant. This is accomplished by considering fixed values, w, for $g(\mathbf{u})$, and the quadratic surfaces in $\mathbf{U}$ that result from using (24). Hence we consider transforming the domain of integration from $\mathbf{u}$ to w, yielding

$$\int f(w)dG(w), \tag{35}$$

thus collapsing the N-dimensional integral into a one-dimensional integral over the measure space induced on the reals by g.

Now we consider the set of all points in the parameter space that map between w_{i-1} and w_i (see Figure 2):

$$A_i = \{\mathbf{u} : w_{i-1} < g(\mathbf{u}) \leq w_i\}. \tag{36}$$

In a summation, the differential $dG(w)$ is represented by the Lebesgue measure (or area) of A_i. Hence we can write

$$\int f(w)dG(w) = \lim_{[w_i - w_{i-1}] \to 0} \sum_i f(w_i)\mu(A_i), \tag{37}$$

in which $\mu(A_i)$ represents the measure of A_i.

Since g is quadratic, A_i is a bounded set iff $g(\mathbf{u}) = w_i$ is the equation of an ellipsoid. If A_i is unbounded, then it can be seen in (37) that the integral in (23) is infinite; therefore, we are only concerned with cases in which $g(\mathbf{u}) = w_i$ represents an ellipsoid.

The measure of A_i is found by taking the set difference of two concentric ellipsoids that are rotated and translated away from the origin, as depicted in Figure 2. Recall that the volume of an ellipse is proportional to its axis lengths. To compute $\mu(A_i)$, we center the ellipsoids at the origin with their axes aligned with the coordinate axes.

By using an affine transformation on $g(\mathbf{u}) = w_i$, described in [8], we obtain the quadratic form ${\mathbf{u}'}^T M' \mathbf{u}' = 1$, in which M' is diagonal. The resulting standardized ellipse equation is

$$\sum_j \left(\frac{u'_j}{a_j}\right)^2 = 1 \tag{38}$$

in which

$$a_j = \sqrt{\frac{k}{\lambda_j}} \tag{39}$$

and

$$k = \sum_j \frac{(b'_j)^2}{4\lambda_j} - c + w. \tag{40}$$

The vector $\mathbf{b}'$ is computed by the product, $\mathbf{R} \cdot \mathbf{b}$, in which $\mathbf{R}$ is the corresponding matrix of eigenvectors of the matrix M. Also, λ_i represents the i-th eigenvalue of the matrix M. The ellipse volume is

$$C_N \prod_{j=1}^{N} a_j \tag{41}$$

in which

$$C_N = \begin{cases} \dfrac{2\pi^{(N/2)}}{N(N/2-1)!} & \text{if } N \text{ is even} \\ 2^{(N+1)/2}\left[\dfrac{\pi^{(N-1)/2}}{N(N-2)(N-4)\cdots 1}\right] & \text{if } N \text{ is odd .} \end{cases} \tag{42}$$

and N is the dimension of $\mathbf{U}$.

In practice we compute the integral (23) by considering a finite approximation of the sum in (37):

$$\int f(g(\mathbf{u}))d\mathbf{u} = \int f(w)dG(w) \approx \sum_{i=0}^{k} f(w_i)\mu(A_i). \tag{43}$$

In general numerical quadrature formulas can also be applied; however, we have obtained satisfactory performance by directly using the sum.

We select starting and ending points, w_0 and w_k in (43) by making the assumption that

$$\int_{-\infty}^{w_0} f(w)dG(w) \approx 0 \tag{44}$$

and

$$\int_{w_k}^{\infty} f(w)dG(w) \approx 0. \tag{45}$$

Hence the performance of this method is affected by the rate at which f approaches the origin. To clarify, the rate at which this method converges is directly related to the width of f. If f is sharply distributed about the origin, i.e. small σ, then the number of discrete sample points, w_i, needed is also small. As f flows out from the origin, i.e. increasing σ, the number of required sample points also increases. This, however, is a small factor when compared to computational savings this method brings.

5 Integration of an N-variate function of a Quadratic ratio

In this section, we consider an integral of the form

$$\int f(h(\mathbf{u}))d(\mathbf{u}) \tag{46}$$

in which $\mathbf{U} = \Re^N$ and h is a ratio of quadratics of the form

$$h(\mathbf{u}) = \frac{\mathbf{u}^T M \mathbf{u}}{\mathbf{u}^T Q \mathbf{u}}. \tag{47}$$

Note that $h(\alpha\mathbf{u}) = h(\mathbf{u})$ for some scalar α. We consequently assume that the parameter space is constrained with the standard norm, $\|\mathbf{u}\| = 1$, and that $\mathbf{u} > \mathbf{0}$. The implicit polynomial model of Section 3.3 is a model family that is included by this form.

Although the integrand in the previous section permitted an efficient decomposition of the domain of integration, a similar approach does not seem possible for an integrand of the type in (46). Due to the quadratic expression appearing in the denominator, the level sets are not ellipses, but instead correspond to intersections of ellipses in the domain of integration. For this problem, however, Monte Carlo integration with importance sampling provides reasonable computation performance. In particular we identify a small, rectangular region in the domain of integration that

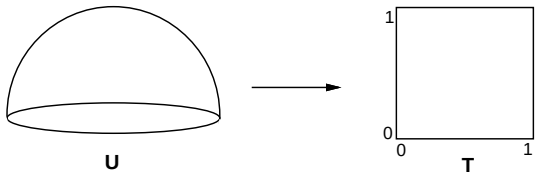


Figure 3: The parameter space is transformed into the unit hypercube for integration.

contains all of the points that significantly contribute to the integral. The random sampling is then only performed inside the rectangular region, and the number of samples required is significantly reduced (by a factor of thousands in many practical cases).

Before proceeding with a Monte Carlo analysis, we first transform the integral over the parameter space into a volume integral over the unit hypercube (see Figure 3). This transformation is a generalization of the spherical coordinate transformation [36, 52]:

$$\begin{array}{lcll} u_1 & = & g_1(\mathbf{t}) & \equiv \quad sin(\pi t_1)sin(\pi t_2)sin(\pi t_3)\ldots sin(\pi t_{N-1}) \\ u_2 & = & g_2(\mathbf{t}) & \equiv \quad sin(\pi t_1)sin(\pi t_2)sin(\pi t_3)\ldots cos(\pi t_{N-1}) \\ \vdots & & & \\ u_{N-2} & = & g_{N-2}(\mathbf{t}) & \equiv \quad sin(\pi t_1)sin(\pi t_2)cos(\pi t_3) \\ u_{N-1} & = & g_{N-1}(\mathbf{t}) & \equiv \quad sin(\pi t_1)cos(\pi t_2) \\ u_N & = & g_N(\mathbf{t}) & \equiv \quad cos(\pi t_1). \end{array} \tag{48}$$

The magnitude of the transformation Jacobian is

$$|J_N| = \pi^{N-1} sin^{N-2}(\pi t_1) sin^{N-3}(\pi t_2) \cdots sin^2(\pi t_{N-3}) sin(\pi t_{N-2}). \tag{49}$$

Finally, the transformed integral becomes

$$\int f(h(\mathbf{u}))d(\mathbf{u}) = \int_0^1 \cdots \int_0^1 f(h(g_1(\mathbf{t}), \ldots, g_N(\mathbf{t}))) \, |J_N| \, dt_1 dt_2 \ldots dt_{N-1}. \tag{50}$$

In the derivation that follows, we treat the new region of integration as a vector of random variables, denoted by $\mathbf{T}$, defined on a unit cube. Let $w \in L^2$ denote the integrand of (50).[2]

The integral (50) is represented as

$$I(w) = \int_T w(\mathbf{t}) d\mathbf{t}. \tag{51}$$

Take a set of n independent samples, $\mathbf{t_1}$, $\mathbf{t_2}$, $\cdots$, $\mathbf{t_n}$, drawn uniformly from the space $\mathbf{T}$. The n^{th} estimate of $I(w)$ is

$$I_n(w) = \frac{1}{n} \sum_{i=1}^{n} w(\mathbf{t_i}). \tag{52}$$

[2]By $h \in L^2$, we mean that $\int h^2 < \infty$.

By the strong law of large numbers, $\|I_n(w) - I(w)\| \to 0$ as $n \to \infty$, with probability one. Consider the variance of the estimate, $\sigma_n^2 = E[I_n(w) - I(w)]^2$. From (52), we observe by linearity that $E[I_n(w)] = E[I(w)] = I(w)$, since the region of integration is the unit cube. From this observation we obtain an expression for the variance of the estimate [41],

$$\sigma_n^2 = E[I_n^2(w)] - 2E[I_n(w)]E[I(w)] + E[I(w)]^2 = \frac{1}{n}\{I(w^2) - [I(w)]^2\}. \tag{53}$$

This indicates that the error variance is reduced at a rate of $1/n$.

Although the result does not depend on the dimension of integration, the convergence can be slow in practice. For each additional significant digit of accuracy, 100 times as many points must be used. When using the Monte Carlo approach for statistical quantities, an additional problem results. If a density becomes peaked around a small portion of the space, then most (or nearly all) of the random samples are drawn from the portion of the space in which the function is approximately zero. As mentioned in Section 2.2, one general approach to this difficulty is to perform importance sampling.

Consider a strictly-positive probability density function, $p(\mathbf{t})$, defined on T. We can compute the equivalent integral

$$I(h) = \int_T \left[\frac{w(\mathbf{t})}{p(\mathbf{t})}\right] p(\mathbf{t})\, d\mathbf{t}, \tag{54}$$

by drawing samples from the density $p(\mathbf{t})$.

We will next determine a rectangular region, $T' \subseteq T$ defined with boundaries $1/2 \pm c_i$, with $0 \leq c_i \leq 1/2$ for each $i \in \{1, \ldots, n-1\}$. We will choose a T' such that there are no points outside of T' that significantly contribute to the integration (due to peaking).

For a given T' and a positive $\epsilon \approx 0$ we can define a pdf for Monte Carlo sampling as

$$p(\mathbf{t}) = \begin{cases} \dfrac{1-\epsilon}{\mu(T')(1-\epsilon) + (1-\mu(T'))\epsilon} & \text{if } \mathbf{t} \in T' \\ \\ \dfrac{\epsilon}{\mu(T')(1-\epsilon) + (1-\mu(T'))\epsilon} & \text{otherwise} \end{cases}, \tag{55}$$

in which $\mu(T')$ represents the area (or measure) of T'. This pdf will concentrate $100(1-\epsilon)$ percent of the samples around the peak. In practice, we choose essentially all of the samples from T', and have found little sensitivity to the choice of ϵ. The pdf in (55) can alternatively be replaced by a pdf that varies within T'. For instance, the samples might be generated by a truncated Gaussian density,

with a mean at the center of T'. An alternative pdf could additionally improve performance, and this remains a topic of future investigation.

We next discuss the selection of the c_i's, and how the sampling is performed. Since the integrand of (46) is formed as a product of densities, we can take some maximum value such that sample points that yield a quadratic-ratio value greater than k contribute relatively little to the integration, since the density at least asymptotically approaches zero. For the model discussed in Section 3.3, the integrand is proportional to a chi-square density. For this case we use the Cornish-Fisher approximation [58] to the chi-square cumulative distribution function to obtain a value for k at the 99.9th percentile for some n. The left side of the equation below represents the set of all parameter values that yield sum-of-squares less than k. This set can be expressed as

$$\left\{ \mathbf{u} : \frac{\mathbf{u}^T M \mathbf{u}}{\mathbf{u}^T Q \mathbf{u}} < k \right\} = \left\{ \mathbf{u} : \mathbf{u}^T [M - kQ] \mathbf{u} < 0 \right\}. \tag{56}$$

Therefore, the right side above describes the interior of a cone, centered at $\mathbf{u} = \mathbf{0}$, which encloses all the points in the parameter space that significantly contribute to the integration. Note that in general, some axes of this cone may be unbounded; however, we additionally have $\|u\| = 1$.

Let $\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ denote the eigenvalues of $[M - kQ]$, in decreasing order. Also, let S denote the corresponding eigenvector matrix, which is a rotation matrix that aligns the cone with the coordinate axes (diagonalizing M). Take $\mathbf{u} = S\mathbf{v}$, and we obtain

$$\mathbf{u}^T [M - kQ] \mathbf{u} = (S\mathbf{v})^T [M - kQ](S\mathbf{v}) = \mathbf{v}^T S^T [M - kQ] S\mathbf{v} = \mathbf{v}^T \Lambda \mathbf{v} \tag{57}$$

in which $\Lambda = diag\{\lambda_1, \lambda_2, \ldots, \lambda_N\}$. Using $\mathbf{v}$, the equation becomes

$$\sum_i \lambda_i v_i^2 < 0, \tag{58}$$

in which some λ_i are negative.

We next determine maximum values for each v_i under the constraint that $\mathbf{v}$ lies in the interior of the cone.

Note that λ_N represents the minimum-valued eigenvector (by the imposed ordering). By inspection of (58), it can be observed that each v_i that corresponds to a positive eigenvalue can be bounded as

$$b_i = \sqrt{\frac{\lambda_i}{(\lambda_i + \lambda_N)}}. \tag{59}$$

The rectangular subset of $\Re^N$ that has corners located at coordinates $\pm b_i$ encloses the cone.

We can apply the inverse of the spherical coordinate transformation (48) to map the corners of the box into T. These form a rectangular subset, T', of T in which the corners have coordinates we denote by $1/2 \pm c_i$.[3]

Using these results, the integral (1) can be computed by sampling from T' and transforming the points into V to obtain

$$\frac{1}{nF} \sum_{\mathbf{v}} f\left(\frac{\sum_i \lambda_i v_i^2}{\mathbf{v}^T S^T Q S \mathbf{v}}\right), \tag{60}$$

in which F represents $p(\mathbf{t})$ when $\mathbf{t} \in T'$. Note that we have $F \approx 1/\mu(T')$, which represents the factor by which the number of required samples is reduced. In Section 6, we show plots of how this factor is affected by region size, region variance, and the degree of the polynomial model used to represent the data.

To compute an integral in the denominator of (4), the transformation must be applied to two different functions of a quadratic form. For this case we use the smallest rectangular region, T', (and corresponding rotation S) of the two regions R_1 and R_2. If that region is R_1 then the integral is computed by

$$\frac{1}{nF} \sum_{\mathbf{v}} f\left(\frac{\sum_i \lambda_i v_i^2}{\mathbf{v}^T S^T Q_1 S \mathbf{v}}\right) \; f\left(\frac{\mathbf{v}^T S^T M_2 S \mathbf{v}}{\mathbf{v}^T S^T Q_2 S \mathbf{v}}\right) \tag{61}$$

in which the λ_i are the eigenvalues of M_1 and S is its eigenvector matrix.

6 Computed examples

The integration techniques presented in this paper have been implemented, and the resulting computations have been used by our segmentation algorithms. In related research we have developed algorithms that: 1) determine a segmentation by iteratively merging regions that have a high probability of homogeneity [37], and 2) determine a set of the most plausible segmentation hypotheses while maintaining corresponding probabilities [36, 38]. Integrals of the form (1) are computed numerous times in these algorithms, thus requiring efficient integration algorithms. We have performed numerous experiments on range and intensity images with up to 20-dimensional parameter spaces, and several results are highlighted in this section to illustrate the utility of the

[3] Some of rectangular faces in the parameter space may lie outside the unit hypersphere. When the first axis is found that is outside, the remaining c_i are set to their maximum value, 1/2.

1. For each pair of adjacent regions R_i, $R_j \in \mathcal{R}$, compute $P(H(R_i \cup R_j)|\mathbf{y_i}, \mathbf{y_j})$, and store the result in a priority queue with elements sorted by probability.
2. Remove the first pair from the queue, R_{m_1},R_{m_2}, and update $\mathcal{R}$ by adding $R_m \equiv R_{m_1} \cup R_{m_2}$ and removing R_{m_1} and R_{m_2}.
3. For each R_i adjacent to R_m, compute $P(H(R_i \cup R_m)|\mathbf{y_i}, \mathbf{y_m})$, and insert the result into the priority queue.
4. If the probability of the first pair in the queue is less than P_c (or alternatively, the number of regions in $\mathcal{R}$ is c) then terminate
5. Go to 2

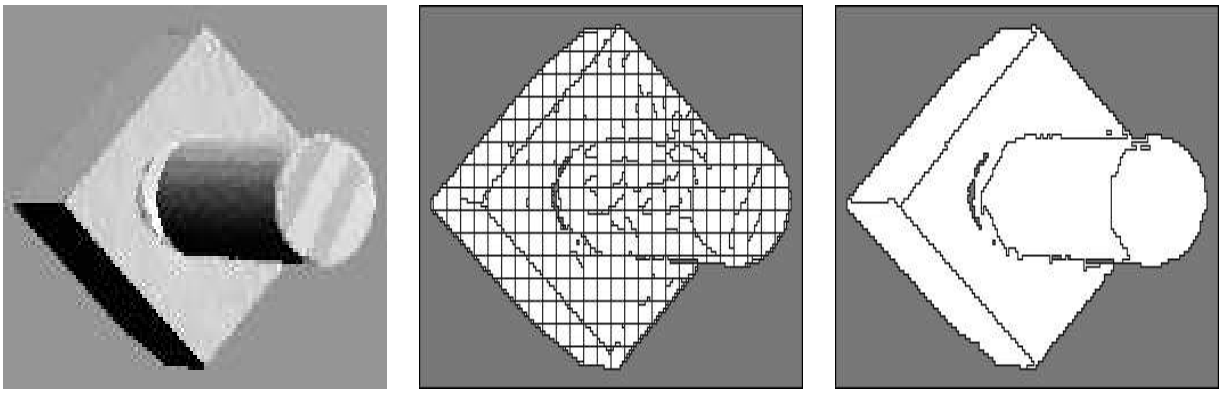
Figure 4: A highest-probability-first merging algorithm.

integration methods. This section also presents some experimental analysis of the performance improvement that is gained over crude Monte Carlo by using the method discussed in Section 5 to select samples.

Figures 5 through 8 show segmentation results that were obtained using the clustering algorithm given in Figure 4. The algorithm can be considered as agglomerative clustering ([18, 48]) with the metric-based merging criterion replaced by the probability of homogeneity, which is briefly discussed in Appendix A. The probabilities are computed for all adjacent region pairs, and the pair with the highest probability is merged. The new, merged region replaces the two individual regions in $\mathcal{R}$, and new merging probabilities are computed. This process iterates until the stopping criterion in Line 4 is met.

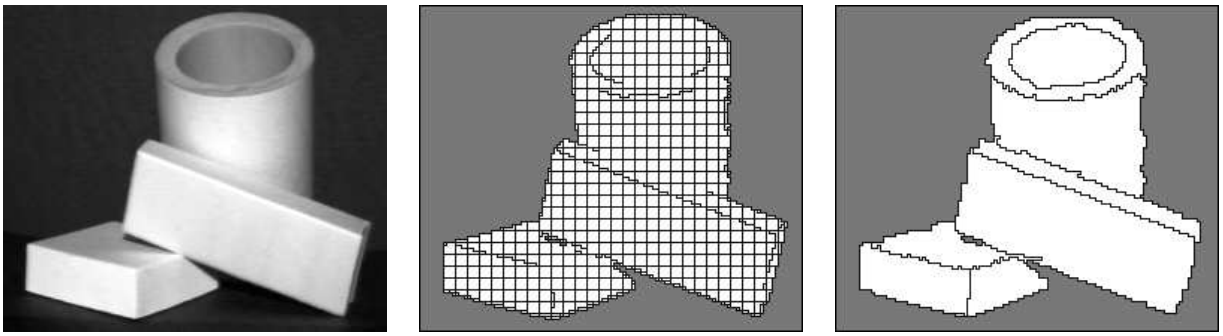
We allow two different stopping criteria in Line 4: either the number of final regions is specified, or merging is terminated after the highest-probability merge is below some value, reflecting a high risk merge. For the segmentations with range data the probability of homogeneity decreases abruptly once the major classes have been formed; consequently, we were able to use an insensitive terminating probability P_c to halt the merging. For the intensity images, the parametric models are not as accurate. Consequently, for many of the images, there is not an abrupt decrease in probabilities, and consequently we specified the class numbers for these experiments.

Figures 5 and 6 show two range image experiments. The three-dimensional range data sets shown in Figures 5.a and 6.a were modeled using the implicit quadric surface model of (Section



a. b. c.

Figure 5: Segmentation results of range data image using the implicit quadric surface model of Section 3.3 and integration technique of Section 5.



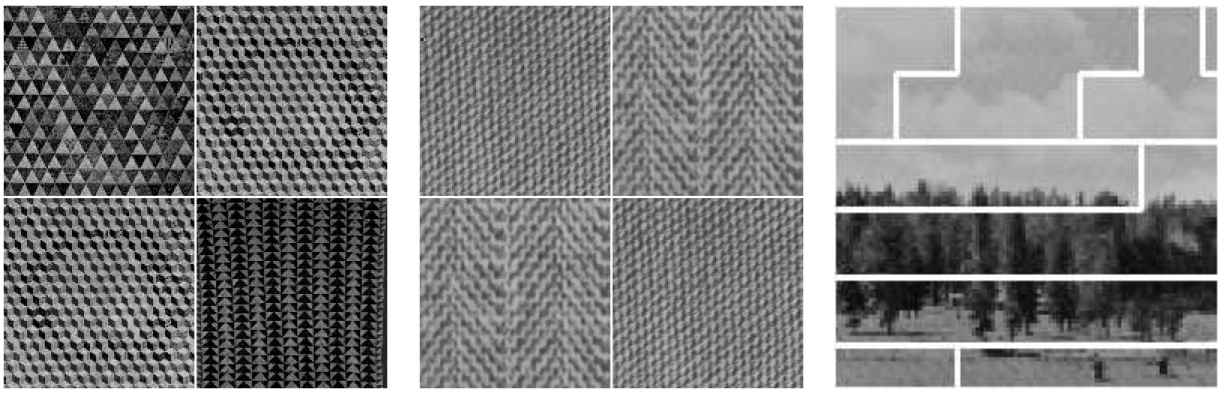
a. b. c.

Figure 6: Segmentation results of range data image using the implicit quadric surface model of Section 3.3 and integration technique of Section 5.

3.3). The use of this model results in 6-dimensional integrals within the probabilistic homogeneity equation (4). The integration technique described in Section 5 performed these integrations efficiently. Figures 5.b and 6.b show an (automated) initial partition of the image, on which the clustering is performed. The final segmentation results, shown in Figures 5.c and 6.c, are obtained after performing the clustering and a simple boundary localization operation [37].

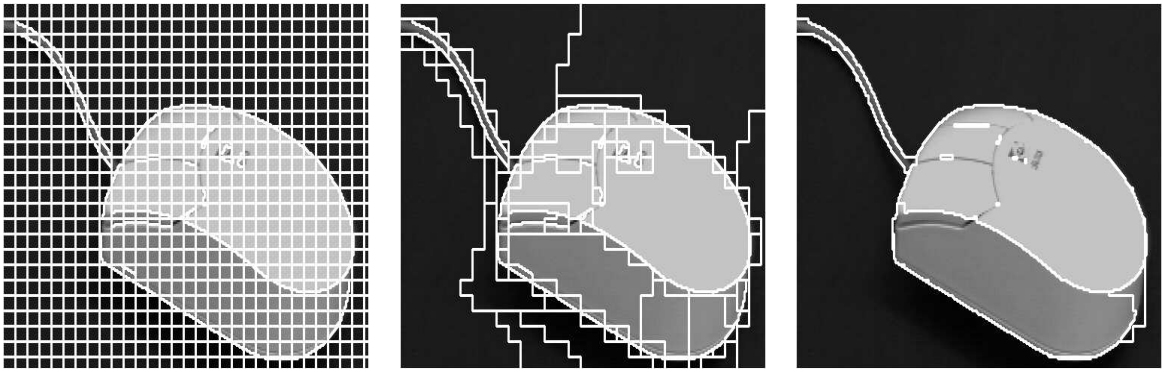
Figure 7 shows texture segmentation results on intensity images that were obtained by clustering on images that were initially partitioned with a square grid to make 64 square regions. These segmentations were computed using 2nd and 3rd order MRF models (described in Section 3.2). The parameter space dimension of these models are 8 and 12, respectively. The technique introduced in Section 4.2 was used to compute the integrals involved in these segmentations.

Figure 8 shows the segmentation of an intensity image by applying a quadric parametric poly-



a. b. c.

Figure 7: Final segmentation results on texture intensity images. The MRF model of Section 3.2 was used along with the integration technique of Section 4.2



a. b. c.

Figure 8: Intensity image segmenation results using the quadric parametric polynomial model of Section 3.1 and the integration technique of Section 4.2.

nomial model of Section 3.1 directly to the intensities. Again, the technique from Section 4.2 was used.

In each case, the appropriate integration method was used to compute the probability of homogeneity. The displayed results depend on the quality of the clustering algorithm, which is heavily dependent on the ability to compute the marginalization, (1), accurately and efficiently.

We next present some results that indicate the computational savings that are obtained by using the method in Section 5 in comparison to using basic Monte Carlo sampling. One of the key difficulties of using crude Monte Carlo in a statistical context is the generation of samples that are concentrated where the probability densities are peaked. The method in Section 5 overcomes this difficulty by identifying a small region that contains the peak. The sample reduction factor, F,

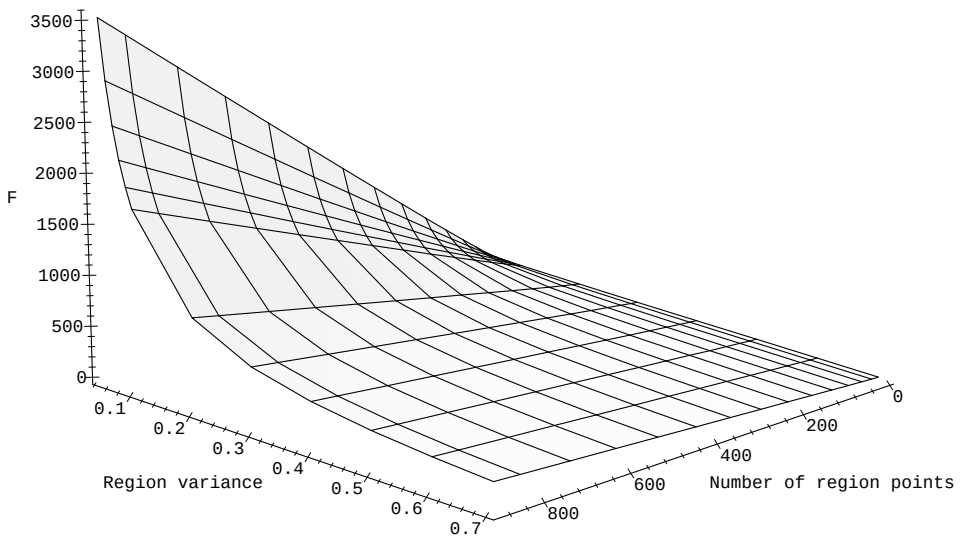


Figure 9: Sample reduction factor using implicit polynomial model of degree 1

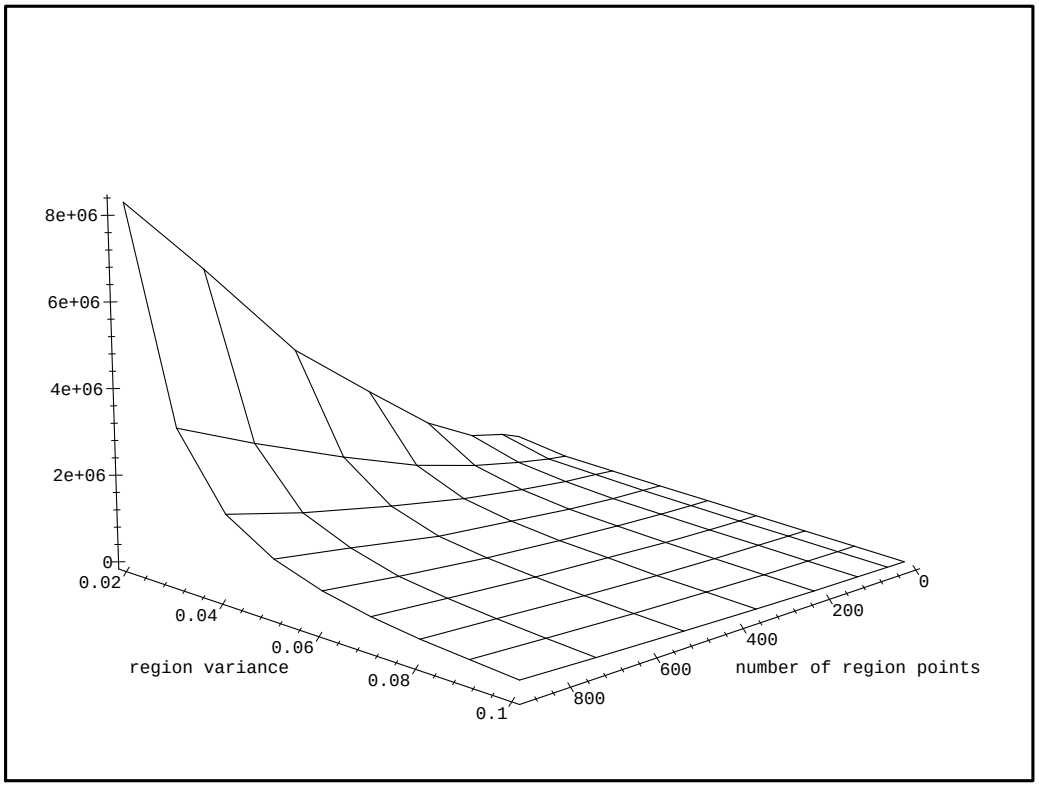


Figure 10: Sample reduction factor using implicit polynomial model of degree 2

used in equation (60), directly indicates the savings that are obtained over crude Monte Carlo, and is graphed for several cases.

The factor varies depending on several things: the region size, the locations of the data points, the region variance, and the degree of model used. Although the factor can vary tremendously from application to application, we provide some indication of its value by constructing synthetic planar regions of various sizes and variances. All of the regions are square, and the first two coordinates of the points lie at consequtive integer coordinates. Gaussian noise was simulated and applied to the data. The regions were modeled using the implicit polynomial model of Section 3.3. The sample reduction factor was then computed for each region. Figure 9 shows the factor as it relates to region size and variance while using the planar model (implicit polynomial model of degree 1). Figure 10 shows the factor while using the quadric model (implicit polynomial model of degree 2). Note that as variance increases, the integral rapidly becomes more peaked. This results in dramatic increases in the savings factor. Increasing the amount of information in the region will also cause the integral to become peaked, although, at a slower rate. This is shown in the figures as a slight increase in reduction factor as the region size is increased.

Suppose the integral is computed for a region that contains 900 points and has a variance of 0.01, which is quite typical in a range-image application. In the planar case, our method produced a savings of nearly 18,000. For the same region, using the quadric model, our method reduced the number of sample points by a factor of over 5.5×10^7. As another example, suppose a region contains 10,000 points and has a variance of 10, which is typical of an intensity-image application. Using the quadric model, the savings factor was over 6700. The largest factor that we have obtained in this experiment for the quadric model was 1.3×10^{18}, which corresponds to a region that has 10,000 points with a variance of 0.00001.

7 Conclusions

We have presented integration methods that compute a marginal density value for a wide class of statistical image models. In particular, these methods have been successfully applied to the implicit polynomial surface model family, the parametric (explicit) polynomial surface model family, and a Markov random field model family. These integration methods were crucial for the Bayesian

computation required in our related segmentation work [36], which use (4).

In general, we believe these computation methods will prove useful for additional image processing applications in which high-dimensional Bayesian modeling is employed. For example, since the models presented in Section 3 are nested families, an interesting area of future work remains to study the application of (3) for model selection.

Acknowledgements

This work was sponsored by NSF under grant #IRI-9110270. We are very grateful for the helpful comments and suggestions provided by the anonymous reviewers.

A Region Merging Probability

With every image element, $\mathbf{x}$, we associate a random vector $\mathbf{X}$, representing the image information, which may be 3D position, intensity, color, or other information. A *region*, R, is some connected subset of the image. In practice, most region-based segmentation algorithms begin by partitioning the image into an initial set of regions, $\mathcal{R}$ (e.g.,[25, 45, 46, 48]). This provides a computational advantage (since there are not as many potential groupings of data points to consider), and also allows statistical models to be effectively exploited [48].

For each $R_k \in \mathcal{R}$ we define the following four components:

- **Parameter space:** A random vector, $\mathbf{U_k}$, which could, for instance, represent a space of polynomial surfaces.
- **Observation space:** A random vector, $\mathbf{Y_k}$, obtained as a function of the data $\mathbf{x} \in R_k$.
- **Noise model:** A conditional density, $p(\mathbf{y_k}|\mathbf{u_k})$, which models noise and uncertainty.
- **Prior Model:** An initial parameter space density, $p(\mathbf{u_k})$.

We have shown that for two regions, R_1 and R_2, the posterior probability that $R_1 \cup R_2$ is homogeneous, given a prior probability, P_0, is determined through the following proposition [37]:

Proposition 1 *Given the observations* $\mathbf{y_1}$ *and* $\mathbf{y_2}$*, the posterior membership probability is*

$$P(H|\mathbf{y_1}, \mathbf{y_2}) = \frac{1}{1 + \lambda_0 \, \lambda_1(\mathbf{y_1}, \mathbf{y_2})}, \tag{62}$$

in which

$$\lambda_0 = \frac{1 - P_0}{P_0} \tag{63}$$

and $\lambda_1(\mathbf{y_1}, \mathbf{y_2}) =$

$$\frac{\left[\int p(\mathbf{y_1}|\mathbf{u_1})p(\mathbf{u_1})\mathbf{d}\mathbf{u_1}\right]\left[\int p(\mathbf{y_2}|\mathbf{u_2})p(\mathbf{u_2})\mathbf{d}\mathbf{u_2}\right]}{\int p(\mathbf{y_1}|\mathbf{u_{12}})p(\mathbf{y_2}|\mathbf{u_{12}})p(\mathbf{u_{12}})\mathbf{d}\mathbf{u_{12}}}. \tag{64}$$

The condition that $R_1 \cup R_2$ is homogeneous has been represented by H. The λ_0 and $\lambda_1(\mathbf{y_1}, \mathbf{y_2})$ ratios represent a decomposition into prior and posterior factors.

References

[1] M. Aitkin. Posterior Bayes factors. *J. Royal Statistical Society*, B53(1):111–142, 1991.

[2] Z. W. Bell. A Bayesian/Monte Carlo segmentation method for images dominated by Gaussian noise. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(9):985–989, September 1989.

[3] J. O. Berger and M. Delampady. Testing precise hypotheses (with discussion). *Statist. Sci.*, 2:317–335, 1989.

[4] D. A. Berry, I. W. Evitt, and R. Pinchin. Statistical inference in crime investigations using deoxyribonucleic acid profiling. *J. Royal Statistical Society*, C41:499–531, 1992.

[5] J. Besag and P. J. Green. Spatial statistics and Bayesian computation. *J. Royal Statistical Society*, B55(1):25–37, 1993.

[6] P. J. Besl and R. C. Jain. Segmentation through variable-order surface fitting. *IEEE Trans. Pattern Anal. Machine Intell.*, 10(2):167–191, March 1988.

[7] J. R. Beveridge, J. Griffith, R. R. Kohler, A. R. Hanson, and E. M. Riseman. Segmenting images using localized histograms and region merging. *Int. J. Comput. Vis.*, 2(3):311–347, January 1989.

[8] D. M. Bloom. *Linear Algebra and Geometry.* Cambridge University Press, 1978.

[9] R. M. Bolle and D. B. Cooper. On optimally combining pieces of information, with application to estimating 3-D complex-object position from range data. *IEEE Trans. Pattern Anal. Machine Intell.*, 8(5):619–638, September 1986.

[10] R. M. Bolle and B. C. Vemuri. On three-dimensional surface reconstruction methods. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(1):1–12, January 1991.

[11] C. Bouman and B. Liu. Multiple resolution segmentation of textured images. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(2):99–113, February 1991.

[12] B. P. Carlin, R. E. Kass, F. Javier Lerch, and B. R. Hugeunard. Predicting working memory failure: A subjective Bayesian approach to model selection. *J. American Statistical Association*, 87:319–327, 1989.

[13] G. Casella and E. I. George. Explaining the Gibbs sampler. *The American Statistician*, 46(3):167–174, August 1992.

[14] M. Celenk. A color clustering technique for image segmentation. *Comp. Vision, Graphics, and Image Process.*, 52:145–170, 1990.

[15] K. P.-S. Chan and C. G. G. Aitkin. Estimation of the bayes' factor in a forensic science problem. *J. Statist. Comput. Simul.*, 33:249–264, 1991.

[16] R. T. Chin and C. R. Dyer. Model-based recognition in robot vision. *Comput. Surv.*, 18(1):67–108, March 1986.

[17] F. S. Cohen and D. B. Cooper. Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian random fields. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(2):195–219, March 1987.

[18] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, NY, 1973.

[19] I. W. Evitt. A quantative theory for interpreting transfer evidence in criminal cases. *Applied Ststistics*, 33:25–32, 1984.

[20] I. W. Evitt, P. E. Cage, and C. G. G. Aitkin. Evaluation of the likelihood ratio for fibre transfer evidence in criminal cases. *Applied Ststistics*, 36:174–180, 1987.

[21] O. D. Faugeras and M. Hebert. The representation, recognition, and locating of 3-D objects. *Int. J. of Robot. Res.*, 5(3):27–52, Fall 1986.

[22] K. Fukunaka. *Introduction to Statistical Pattern Recognition.* Academic Press, New York, NY, 1972.

[23] D. Geman and S. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, 6(6):721–741, November 1984.

[24] I. J. Good. Weight of evidence: A brief survey. *Bayesian Statistics*, 2:249–270, 1985.

[25] R. M. Haralick and L. G. Shapiro. Image segmentation techniques. *Comp. Vision, Graphics, and Image Process.*, 29:100–132, January 1985.

[26] R. M. Haralick and L. Watson. A facet model for image data. *Comp. Vision, Graphics, and Image Process.*, 15:113–129, February 1981.

[27] R. Hoffman and A. K. Jain. Segmentation and classification of range images. *IEEE Trans. Pattern Anal. Machine Intell.*, 9(5):608–620, September 1987.

[28] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1988.

[29] M. H. Kalos and P. A. Whitlock. *Monte Carlo Methods.* Wiley, New York, NY, 1986.

[30] R. L. Kashyap and R. Chellappa. Estimation and choice of neighbors in spatial-interaction models of images. *IEEE Trans. Information Theory*, 29(1):60–72, January 1983.

[31] R. E. Kass, L. Tierney, and J. B. Kadane. Asymptotics in Bayesian computation. *Bayesian Statistics*, 3:263–278, 1988.

[32] R. E. Kass and S. K. Vaidyanathan. Approximate Bayes factors and orthogonal parameters, with application to testing equality of two binonmial proportions. *J. Royal Statistical Society*, B54:129–144, 1992.

[33] D. J. Kreigman and J. Ponce. On recognizing and positioning curved 3D objects from image contours. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(12):1127–1137, December 1990.

[34] V. I. Krylov. *Approximate Calculation of Integrals.* Macmillan, New York, NY, 1962.

[35] S. Lakshmanan and H. Derin. Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. *IEEE Trans. Pattern Anal. Machine Intell.*, 11(48):799–813, August 1989.

[36] S. M. LaValle. A Bayesian framework for considering probability distributions of image segments and segmentations. Master's thesis, Univ. of Illinois, Urbana/Champaign, December 1992.

[37] S. M. LaValle and S. A. Hutchinson. A Bayesian segmentation methodology for parametric image models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):211–218, February 1995.

[38] S. M. LaValle and S. A. Hutchinson. A framework for constructing probability distributions on the space of segmentations. *Computer Vision and Image Understanding*, 61(2):203–230, March 1995.

[39] A. Leonardis, A. Gupta, and R. Bajcsy. Segmentation as the search for the best description of the image in terms of primitives. In *Proc. Int. Conf. on Computer Vision*, pages 121–125, 1990.

[40] D. V. Lindley. A statistical paradox. *Biometrika*, 44:187–192, 1957.

[41] E. Masry and S. Cambanis. Trapezoidal Monte Carlo integration. *SIAM J. Numer. Anal.*, 27(1):225–246, February 1990.

[42] M. J. Mirza and K. L. Boyer. An information theoretic robust sequential procedure for surface model order selection in noisy range data. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 366–371, 1992.

[43] A. Noble and J. Mundy. Toward template-based tolerancing from a Bayesian viewpoint. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 246–252, New York, June 1993.

[44] D. K. Panjwani and G. Healey. Unsupervised segmentation of textured color images using markov random fields. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 776–777, New York, June 1993.

[45] T. Pavlidis. *Structural Pattern Recognition.* Springer-Verlag, New York, NY, 1977.

[46] B. Sabata, F. Arman, and J. K. Aggarwal. Segmentation of 3D range images using pyramidal data structures. *Comp. Vision, Graphics, and Image Process.*, 57:373–387, May 1993.

[47] P. K. Sahoo, S. Soltani, A. K. Wong, and Y. C. Chen. A survey of thresholding techniques. *Comp. Vision, Graphics, and Image Process.*, 41:233–260, February 1988.

[48] J. F. Silverman and D. B. Cooper. Bayesian clustering for unsupervised estimation of surface and texture models. *IEEE Trans. Pattern Anal. Machine Intell.*, 10(4):482–496, July 1988.

[49] A. F. M. Smith. Bayesian computational methods. *Phil. Trans. R. Soc. Lond. A*, 337:369–386, 1991.

[50] A. F. M. Smith and G. O. Roberts. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *J. Royal Statistical Society*, B55(1):3–23, 1991.

[51] A. F. M. Smith and D. J. Spiegelhalter. Bayes factors and choice criteria for linear models. *J. Royal Statistical Society*, B42:213–220, 1980.

[52] A. H. Stroud. *Approximate Calculation of Multiple Integrals.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971.

[53] S. Sullivan, J. Ponce, and L. Sandford. On using geometric distance fits to estimate 3D object shape, pose, and deformation from range, CT, and video images. In *Proc. IEEE Conf. on Comp. Vision and Patt. Recog.*, pages 110–115, 1993.

[54] R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *Int. J. Comput. Vis.*, 5(3):271–301, December 1990.

[55] G. Taubin. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 13(11):1115–1137, November 1991.

[56] G. Taubin and D.B. Cooper. Recognition and positioning of 3D piecewise algebraic objects using Euclidean invariants. In *Proc. Workshop on the Integration of Numerical and Symbolic Computing Methods*, Saratoga Springs, NY, July 1990.

[57] S. Yakowitz, J. E. Krimmel, and F. Szidarovsky. Weighted Monte Carlo integration. *SIAM J. Numer. Anal.*, 15(6):1289–1300, December 1978.

[58] J. H. Zar. Approximations for the percentage points of the chi-squared distribution. *J. Appl. Statist.*, 27(3):280–290, 1978.

[59] J. Zhang and J. W. Modestino. A model-fitting approach to cluster validation with applications to stochastic model-based image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(10):1009–1017, October 1990.