Energy Minimization by Effective Jump-Diffusion Method for Range Segmentation

Feng Han[1], Zhuowen Tu[2] and Song-Chun Zhu[1,2]

1. Department of Computer Science,
2. Department of Statistics,
University of California at Los Angeles[1],
Los Angeles, CA 90095

Abstract

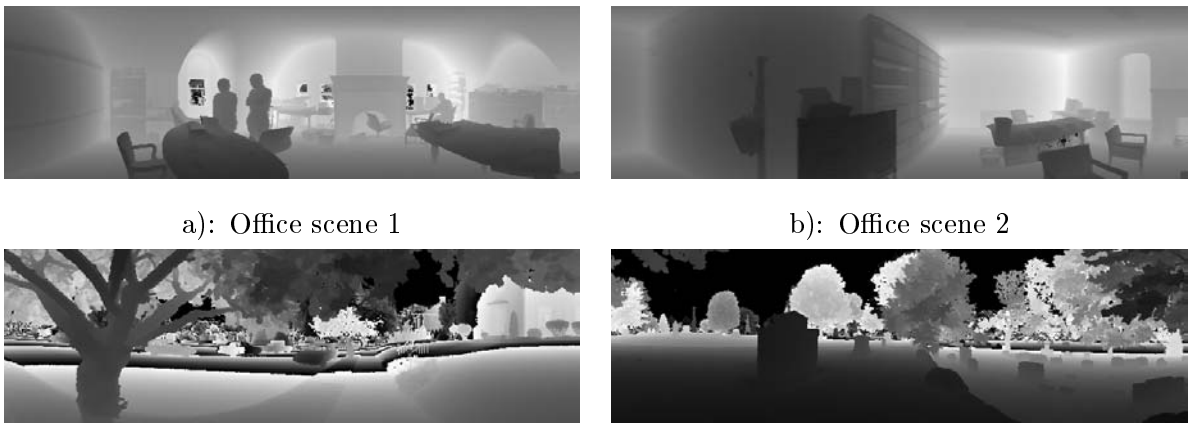
This paper presents a stochastic jump-diffusion method for optimizing a Bayesian posterior probability in segmenting range data and their associated reflectance images. The algorithm works well on complex real world scenes (indoor and outdoor), which consist of an unknown number of objects (or surfaces) of various sizes and types, such as planes, conics, smooth surfaces, and cluttered objects (like trees and bushes). Formulated in the Bayesian framework, the posterior probability is distributed over a countable number of subspaces of varying dimensions. To search for globally optimal solution, the paper adopts a stochastic jump-diffusion process[16] to simulate a Markov chain random walk for exploring this complex solution space. A number of reversible jump[15] dynamics realize the moves between different subspaces, such as switching surface models and changing the number of objects. The stochastic Langevin equation realizes diffusions, such as region competition[39] in each subspace. To achieve effective computation, the algorithm pre-computes some importance proposal probabilities through Hough transforms, edge detection, and data clustering. The latter is used by the Markov chains for fast mixing. For the varying sizes (scales) of objects in natural scenes, the algorithm computes in a multi-scale fashion. The algorithm is first tested against an ensemble of 1D simulated data for performance analysis. Then the algorithm is applied to three datasets of range images under the same parameter setting. The results are satisfactory in comparison with manual segmentation.

Keywords: Energy Minimization, Jump-Diffusion, Range Segmentation, Markov Chain Monte Carlo, Data Clustering, Edge Detection, Hough Transform, Changing Point Detection.

---

[1]The work was done when the authors were at the Computer Science Department, Ohio State University

1 Introduction



a): Office scene 1 b): Office scene 2

c): A street scene d): A cemetery scene

Figure 1: Four examples of indoor and outdoor scenes from the Brown range dataset. The laser scanner scans the scene in cylindric coordinates and produces panoramic views of the scenes.

This paper is concerned with the segmentation and surface reconstruction of real world scenes from laser range images. Some typical examples of the range images, both indoor and outdoor, are shown in Fig. 1. Our research interest is motivated by some new developments in sensor technology and demands in applications.

1. Recently, high precision laser range scanners have become accessible to many users to acquire complex real world scenes like those displayed in Fig. 1. In aerospace imaging, 3D Lidar images have accuracy up to 1 centimeter for terrain maps and city scenes. These images provide much more accurate depth information than conventional vision cues, such as motion, shading, and binocular stereo. Thus it is increasingly important to have effective algorithms for parsing these range images.

2. There are new applications in graphics, visualization and spatial information management, for example, image based rendering, augmented reality, and spatio-temporal databases of 3D urban and suburban maps and developments. These demand the reconstruction of complex 3D scenes from range data.

3. The study of 3D range data is also motivated by the need of prior knowledge (prob-

abilistic models) in solving the ill-posed problems in vision. Currently, most vision algorithms assume low level smoothness priors. Though some work has been done for studying natural image statistics[21] and learning priors from optic images[40], to the best of our knowledge, there is no work for learning prior models, for example the layout of 3D objects, from real world range scenes as Fig. 1 shows. Such prior models of 3D scenes are badly needed for many 3D vision tasks.

In contrast to the new developments and applications, existing range image segmentation algorithms are mostly motivated by traditional applications in recognizing industry parts in assembly lines, and thus they are focused on block worlds with mostly polyhedra objects. In the literature, methods on general image segmentation have been introduced and extended to rang image segmentation, for example, edge detection[23], region based methods[3, 5], and surface fitting[17], clustering[18, 12], and generalized Hough transform[4] for detecting parametric surfaces of low dimensions. We refer to (Hoover et al, 1996) for a survey of range segmentation algorithms and a good empirical comparison done jointly by a few groups[19]. One latest interesting work dealing with range data is directed to [44]. Generally speaking, algorithms for range segmentation are not as advanced as those for intensity image segmentation. For example, there is no algorithm, to our knowledge, which can satisfactorily segment complex scenes as those displayed in Fig. 1.

The difficulties for segmenting real world scenes lie in several aspects.

Firstly, natural scenes contain many types of objects, for example, man-made objects (buildings, desks), animate objects (human and animals), and free form objects (trees and terrain). These objects should be represented by various families of surface models which have different dimensions of parameterization. For example, Shade et al.[32] in graphics argued for a spectrum of representation, from polygon to sprites and planar texture maps, for various precision requirements of photorealism. Thus an algorithm must engage multiple surface models in representation and be capable of switching between these models in computation. In the formulation of Bayesian inference, the posterior probability (or energy functional) is distributed over a countable number subspaces of varying dimensions. Each subspace is for a certain number of surface models combined. Thus conventional greedy algorithms are not applicable.

Secondly, objects (or surfaces) in natural scene come with multiple scales. For example, the office scenes in Fig. 1 contain large surfaces such as walls, ceilings, and floors, middle size objects such as people, chairs and tables, and small objects such as books and cups

on the desk top. This is in contrast with the block world (see Figures 13 and 14) where objects are of similar sizes. In computation, the algorithm must engage large and small moves and extract information at multiple scales. In representation, this broad range of scales seems to disable the conventional model complexity criteria, such as MDL (minimum description length)[31], AIC (Akaike Information Criterion) [1], BIC (Bayesian information criterion)[2], which are derived from the concern of information coding. Thus other prior models should be sought to ensure that surfaces of various sizes appear in a scene and thus in a segmentation.

Thirdly, though range data are very accurate on depth, they are very noisy in comparison with optical images around object boundaries. It gets worse in objects like trees and bushes. Furthermore depth data are missing at infinity objects, such as the sky, or at metal, glass and ceramic objects where the laser rays never return to the scanner.

Motivated by these problems, this paper presents a stochastic jump-diffusion algorithm for segmenting and reconstructing 3D scenes from range images. In comparison with previous work on range segmentation, the paper makes the following contributions.

1. To deal with the variety of objects in real world scenes, this paper incorporates five types of surface models, such as planes and conics for man-made objects, splines for free-form flexible objects, and a non-parametric (3D histogram) model for cluttered objects. These surfaces models compete to explain the range data under the constraints of a statistical prior for model complexity. The paper also introduces various prior models on surfaces, boundaries, and vertices (corners) to ensure regularities.

2. To handle missing range data, the algorithm integrates the range data with their associated reflectance map under the Bayes framework. The reflectance measures the proportion of laser energy returned from surface in $[0, 1]$ and therefore carries material properties. It is especially useful for glass, metal, ceramics, and the sky.

3. To achieve globally optimal solutions, the algorithm simulates ergodic Markov chains to sample the posterior probability over a complex solution space with countable subspaces of varying dimensions. The Markov chain consists of reversible jumps and stochastic diffusions. The jumps realize split and merge, model switching, while the diffusions realize boundary evolution and competition and model adaptation.

4. To improve the convergence speed and use information at multiple scales, the algorithm pre-computes some bottom-up information in a coarse-to-fine manner: edge

detection and surface clustering at multiple scales. The computed information is expressed as *importance proposal probabilities*[34] on the surface and boundaries for narrowing the search spaces in a probabilistic fashion, and drives the Markov chain for fast mixing. This follows a data driven Markov chain Monte Carlo method which has been successfully applied in parsing optical images[34, 35].

The algorithm is first tested against an ensemble of one hundred 1D simulated range data for performance analysis. Then the algorithm is applied to three datasets of range images. The first two are the standard USF polyhedra data and curved-surface data for comparison, and the third is from Brown university which contains real world scenes. The experiments demonstrate robust and satisfactory results under the same parameter setting.

In the following of the paper, we first discuss the jump-diffusion process and evaluate the performance in 1D simulated data. Then we present a Bayesian formulation of the problem and the design of algorithm. Finally, we show experimental results and conclude with some critical discussions.

2 Jump-diffusion for energy minimization: a toy example

In this section, we discuss the jump-diffusion and bottom-up approaches using an ensemble of simulated 1D range data, thus the fundamental ideas of the algorithm will not be entangled in the details of the 2D range segmentation problem. Furthermore since we know the ground truth for the simulated data, we evaluate how well the algorithm approaches globally optimal solutions and we also compare the convergence speeds of the jump-diffusion algorithms with different designs.

2.1 Segmenting 1D range data: Problem formulation

Figure 2.a displays a simulated 1D range image $\mathbf{I}(x),\ x \in [0,1]$. It is generated by adding Gaussian noise $N(0,\sigma^2)$ to the original surfaces $\mathbf{I}_o$ in figure 2.b. $\mathbf{I}_{\text{th}}$ consists of an unknown number of k surfaces which could be either straight lines or circular arcs, separated by *changing points*,

$$0 = x_0 < x_1 < x_2 < \cdots < x_{k-1} < x_k = 1.$$

Let $\ell_i \in \{line, circle\}$ indexes the surface types for interval $[x_{i-1}, x_i)$ with parameters θ_i, $i = 1, 2, ..., k$. For a straight line $\theta = (s, \rho)$ represents the slope s and interception ρ. For a

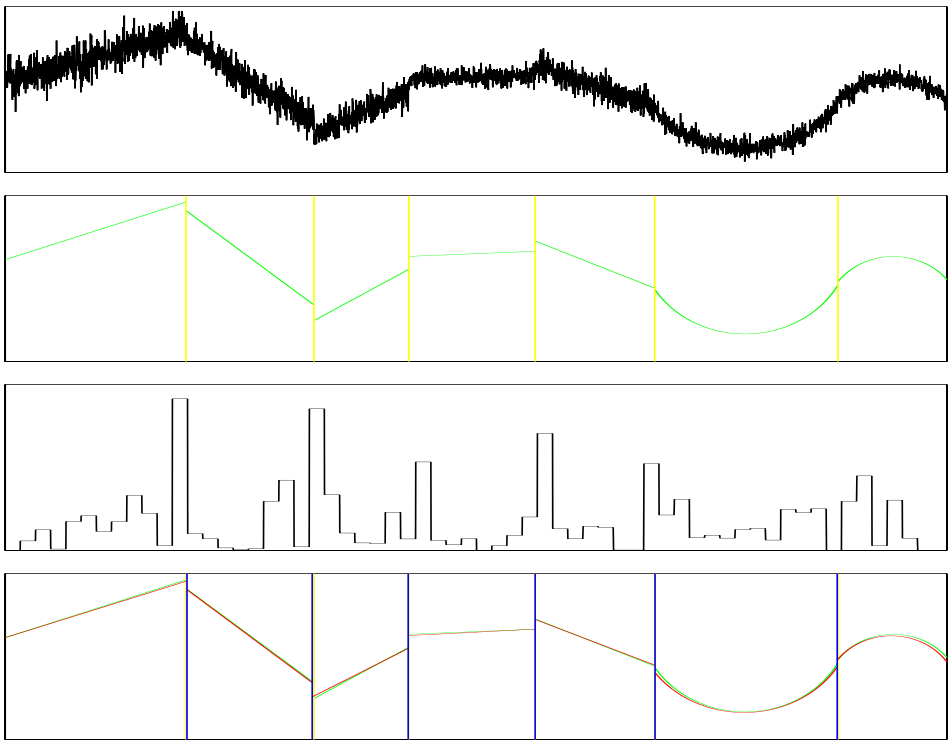


Figure 2: a). A 1D range image $\mathbf{I}(x),\ x \in [0,1)$, b). the true segmentation W_{th}, c). edgeness measure $b(x)\ x \in [0,1)$ for changing point detection on $\mathbf{I}$. d). The best solution W^* found by the algorithm plotted against W_{th}.

circular arc, $\theta = (\xi, \eta, \gamma)$ represents the center (ξ, η) and radius γ_i. Thus the "world scene" is represented by a vector of random variables,

$$W = (k, \{x_i : i = 1, 2, ..., k-1\},\ \{(\ell_i, \theta_i); i = 1, 2, ..., k\}).$$

The surface $\mathbf{I}_o$ is fully determined by W with

$$\mathbf{I}_o(x) = \mathbf{I}_o(x; \ell_i, \theta_i), x \in [x_{i-1}, x_i), i = 1, 2, ..., k.$$

By standard Bayesian formulation, we have the posterior probability

$$p(W|\mathbf{I}) \propto \exp\{-\frac{1}{2\sigma^2}\sum_{i=1}^{k}\int_{x_{i-1}}^{x_i}(\mathbf{I}(x) - I_o(x; \ell_i, \theta_i))^2 dx\} \cdot p(k)\prod_{i=1}^{k} p(\theta_i|\ell_i) \tag{1}$$

The first factor above is the likelihood and the rest are prior probabilities $p(k) \propto \exp^{-\lambda_o k}$ and $p(\theta_i|\ell_i) \propto \exp^{\lambda \#\theta_i}$ the parameters which penalize the parameter numbers $\#\theta_i$. Other

variables are assumed to be uniformly distributed for simplicity. Thus an energy function is defined,

$$E(W) = \frac{1}{2\sigma^2}\sum_{i=1}^{k}\int_{x_{i-1}}^{x_i}(\mathbf{I}(x) - \mathbf{I}_o(x;\ell_i,\theta_i))^2 dx + \lambda_o k + \sum_{i=1}^{k}\#\theta_i. \quad (2)$$

Now, the problem is that W does not have a fixed dimension. For example, if there are 8 objects ($k = 8$) with 5 lines and 3 circular arcs, then W has $7+2\times5+3\times3 = 26$ dimensions. But if $k = 10$ with 4 lines and 6 circular arcs,then it has 35 dimensions. The probability $p(W|\mathbf{I})$ (or the energy $E(W)$) is thus distributed over a countable number of subspaces of varying dimensions. Thus to achieve globally optimal solutions, we should adopt more advanced energy minimization approach – the jump-diffusion method via Markov chain Monte Carlo.

2.2 Background of jump-diffusion

In statistics literature, there were some designs of *hybrid sampler* (Tierney, 1994) which traverses parameter spaces of varying dimensions by random choices of different Markov chain moves. Grenander and Miller (1994) first introduced the jump-diffusion process which mixed the Metropolis-Hastings method[26] and Langevin equations[14]. Other notable work includes (Green, 1995) for reversible-jumps and (Phillips and Smith, 1995) for model comparison with reversible jumps. In this subsection, we briefly present the basic ideas and discuss some problems with convergence speed.

In the 1D range segmentation problem above, let Ω denote the solution space which is a union of a countable number of spaces

$$\Omega = \cup_{n=1}^{\infty}\Omega_n,$$

where $n = (k, \ell_1, ..., \ell_k)$ indexes the various combinations and subspace. The algorithm simulates ergodic Markov chain traversing the solution space by coordinating two types of moves: reversible jumps between different subspaces and stochastic diffusion within each subspace.

1. Reversible jumps

Let $W = (n, \psi)$ be the state of a Markov chain at time t with $\psi \in \Omega_n$. In an infinitesimal time interval dt, the Markov chain jumps to a new space Ω_m ($m \neq n$) at state $W' = (m, \phi)$. There are three types of moves: 1). switching a line to a circular arc, or vice versa, 2).

merging two adjacent regions to a line or a circle, 3). split a region into two regions (lines or circles). Thus a subspace Ω_n with k regions is connected to $k + 2 \times (k-1) + 4 \times k = 7k - 2$ other subspaces by the three types of moves. We denote by $\mathcal{C}(n)$ the set of indexes to the $7K - 2$ subspaces, and denote

$$\mathcal{J}(n, \psi) = \cup_{m \in \mathcal{C}(n)} \Omega_m, \quad \text{and} \quad (m, \phi) \in \mathcal{J}(n, \psi) \text{ iff } (n, \psi) \in \mathcal{J}(m, \phi).$$

the spaces connected to point (n, ψ) by the three jumps.

The jump is realized by a Metropolis move[26] which proposes to move from (n, ψ) to (m, ϕ) $(m \neq n)$ by a proposal probability $q(n \to m)q(\phi|m)d\phi$ and accepts the proposal with probability

$$\alpha((n, \psi) \to (m, \phi)) = \min(1, \frac{q(m \to n)q(\psi|n)d\psi \cdot p(m, \phi|\mathbf{I})d\phi}{q(n \to m)q(\phi|m)d\phi \cdot p(n, \psi|\mathbf{I})d\psi}). \tag{3}$$

In all designs, we have

$$\sum_{m \in \mathcal{C}(n)} q(n \to m) = 1 \quad \forall n.$$

The Markov transition probability is

$$P((n, \psi) \to (m, \phi))d\phi = q(n \to m)q(\phi|m)d\phi \; \alpha((n, \psi) \to (m, \phi)).$$

Then for any two Borel sets $A \subset \Omega_n$ and $B \subset \Omega_m$, the *detailed balance equation* holds

$$\int_A p(n, \psi|\mathbf{I})d\psi \int_B P((n, \psi), (m, \phi))d\phi = \int_B p(m, \phi|\mathbf{I})d\phi \int_A P((m, \phi), (n, d\psi))d\psi. \tag{4}$$

We can view the subspaces Ω_n, Ω_m as discrete points with probabilities,

$$\pi(n) = \int_{\Omega_n} p(n, \psi|\mathbf{I})d\psi, \quad \pi(m) = \int_{\Omega_m} p(m, \phi|\mathbf{I})d\phi.$$

the discrete transition matrix is P with each element

$$P_{n,m} = P(n \to m) = \begin{cases} \int_{\Omega_n} p(n, \psi|\mathbf{I})d\psi \int_{\omega_m} P((n, \psi) \to (m, \phi))d\phi, & \text{if } m \in \mathcal{C}(n), \\ 0, & \text{else} \end{cases} \tag{5}$$

Then we have an irreducible and aperiodic Markov chain with detailed balance

$$\pi(m)P(m \to n) = \pi(n)P(n \to m), \quad \forall n, m.$$

Thus we have the following conclusion from the Perron-Frobenius theorem (see [7]).

Theorem 1 *Suppose the total number k of object is finite in a scene, the three types of jumps (model switching, split, and merge) realize an irreducible and aperiodic Markov chains with a finite stochastic matrix P. Then starting with an arbitrary initial distribution π_o, after M jumps, the Markov chain state follows a probability that approaching π as the unique invariant probability,*

$$\pi_o P^M = \pi + O(|\lambda_2|^M).$$

$0 \leq |\lambda_2| < 1$ *is the second largest eigenvector modulus (SLEM) of P.*

Thus the Markov chain visits each subspace Ω_n at probability $\pi(n)$ after some burning period $M > M_o$.

2. Stochastic diffusions

But not every two points $\psi \in \Omega_n$ and $\phi \in \Omega_m$ are connected directly by the three jumps. Thus stochastic diffusion (or Langevin) equations are used to sample (or minimize) in each subspace Ω_n. As $n = (k, \ell_1, ..., \ell_k)$ is fixed, the energy functional $E(W)$ becomes

$$E[\psi] = E(x_1, ..., x_{k-1},\ \theta_1, ..., \theta_k) = \frac{1}{2\sigma^2} \sum_{i=1}^{k} \int_{x_{i-1}}^{x_i} (\mathbf{I}(x) - \mathbf{I}_o(x; \ell_i, \theta_i))^2 dx + \text{const.}$$

The Langevin equation is a steepest descent driven by Gaussian random force dw_t (Brownian motion) with temperature T,

$$d\psi_t = -\frac{dE[\psi]}{d\psi} dt + \sqrt{2T(t)} dw_t, \quad dw_t \sim N(0, (dt)^2).$$

The following conclusion is well known in the literature (see [14] and refs therein).

Theorem 2 *The continuous Langevin equation above simulates a Markov chain with stationary density*

$$\pi(\psi) \propto e^{-E(\psi)/T}.$$

For example, the movement of changing point is driven by

$$\frac{dx_i(t)}{dt} = \frac{1}{2\sigma^2}((\mathbf{I}(x) - \mathbf{I}_o(x; \ell_{i-1}, \theta_{i-1}))^2 - (\mathbf{I}(x) - \mathbf{I}_o(x; \ell_i, \theta_i))^2) + \sqrt{2T(t)} N(0, 1).$$

This is the 1D case of the region competition equation[39]. In practice, the Brownian motion is found to be useful to avoid local pitfalls. For $\Theta_i, i = 1, 2..., k$, it may appear that we can fit the best θ_i for each interval $[x_{i-1}, x_i)$ instead of running the diffusion. But This usually is an "over-commitment" because the current interval may contain more than one object. Thus a question rises for how long we should run the diffusion between the jumps.

3. The coordination of jumps and diffusions

The continuous diffusion is interrupted by some jumps at time instances $t_1 < t_2 < \cdots < t_M$... as Poisson events. In practice, the diffusion always runs in discrete time steps with Δt. Thus the discrete waiting time τ_j between two consecutive jumps is

$$w = \frac{t_{j+1} - t_j}{\Delta t} \sim p(w) = e^{-\tau} \frac{\tau^w}{w!},$$

with the expected waiting time $E[w] = \tau$ which controls the frequency of jumps. Then the two processes realize ergodic Markov chain sampling the posterior probability $p(W|\mathbf{I})$ over the solution space Ω[16]. Due to the strong structures in real world signals/scenes, the posterior probability is often very "cold", thus one may have to raise the temperature slightly ($T = 5\ 10$) and reduce it gradually to ($T = 0.1\ 1$) to find a nearly global optimum.

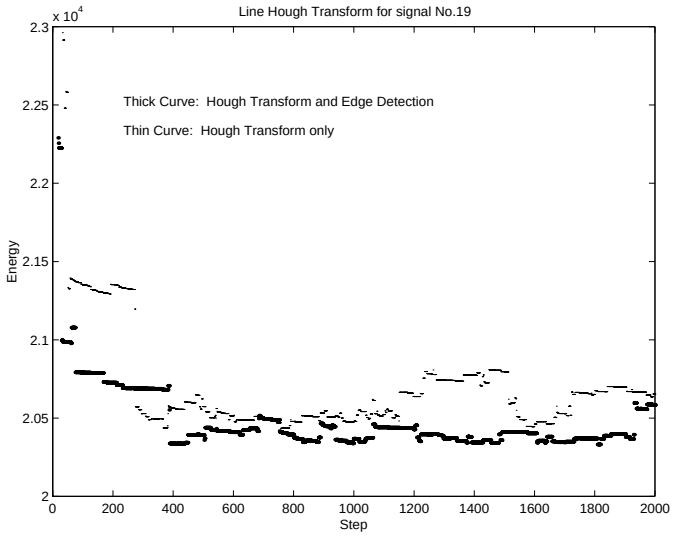


Figure 3: The plots of the energy $E(W)$ against running time t of two Markov chain trials on the signal in fig.2.

Figure 3 shows two trials (thin and thick curves respectively) of the jump-diffusion process running on the input 1D range data in Figure 2. The energy plots go up and down (i.e. not greedy) and the continuous energy curves are interrupted by jumps.

To summarize the jump-diffusion process, we draw figure 4 to illustrate the Markov chain dynamics. In figure 4, three two dimensional subspaces $\Omega_m, \Omega_n, \omega_p$ are illustrated with some probabilities represented by the landscapes. At each subspace, some trials of the diffusion equations are simulated from a point (see several pathes). Then the Markov chain can jump between these subspaces as the large arrows show.

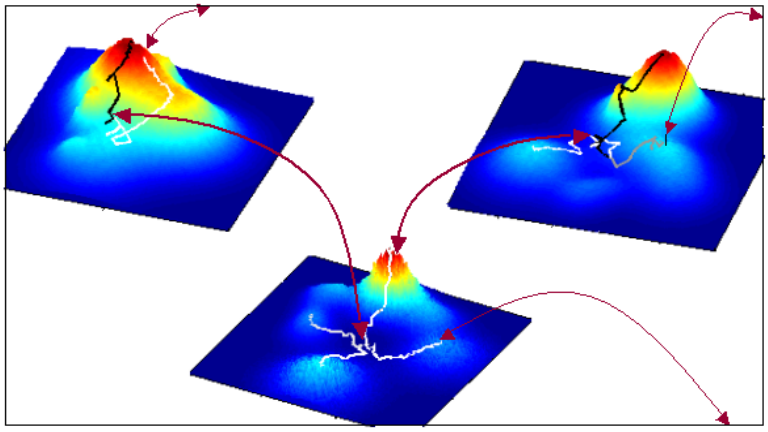


Figure 4: An illustration of the jump-diffusion process.

2.3 Data-driven techniques and convergence evaluation

Though the jump-diffusion is a general tool for energy minimization. Its applications to computer vision have been very limited (indeed prohibited) by its computing speed. There are some theorems for bounding the second largest eigenvalue λ_2 by the conductance of the transition graph given by the stochastic matrix $P_{n,m}$[7]. Such analysis provides us with some intuition of improving the speed.

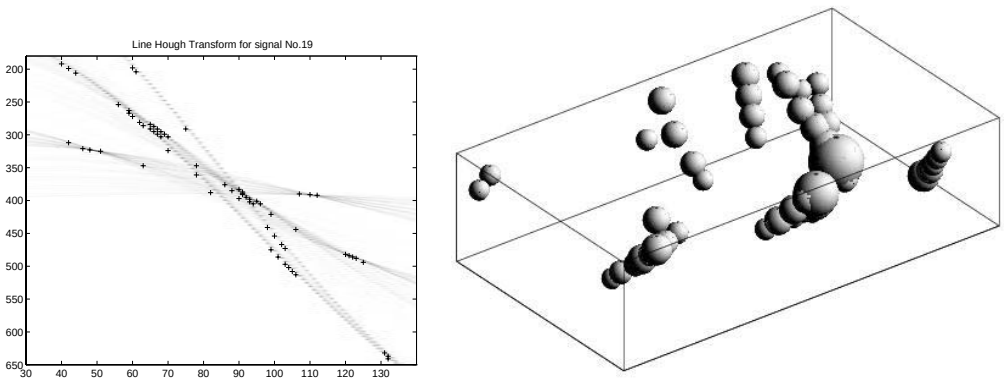
We observed that the bottlenecks are in the jumps affected by the design of proposal probabilities, $q(\phi|m)$ and $q(\psi|n)$ in equation (3). More specifically,

$$q(\phi\,|m) = \begin{cases} q(\theta_i|\ell_i,[x_{i-1},x_i)) & \text{switch } [x_{i-1},x_i) \text{ to model } (\ell_i,\theta_i); \\ q(\theta|\ell,[x_{i-2},x_i)) & \text{merge to a model } (\ell,\theta); \\ q(x|[x_{i-1},x_i))q(\theta_a|\ell_a,[x_{i-1},x))q(\theta_b|\ell_b,[x,x_i)) & \text{split } [x_{i-1},x_i) \text{ into } (\ell_a,\theta_a) \text{ and } (\ell_b,\theta_b) \text{ at } x. \end{cases} \tag{6}$$

In statistical literature(Grenander and Miller 94 and Green 95), the proposal probabilities were taken mostly as uniform distributions. That is to jump to randomly selected lines or/and circles for new models. Such proposals are almost always rejected because the ratio $p(m,\phi|\mathbf{I})/p(n,\psi|\mathbf{I}) = e^{-\Delta E}$ is close to zero.

To have smart jumps, the Markov chain must be equipped with some domain knowledge (or heuristics). Recently the authors introduced a data-driven Markov chain Monte Carlo scheme[34] which computes the proposal probabilities by bottom-up method in each of the parameter space for x, $(line, s, \rho)$ and (arc, ξ, η, γ).

1. Hough transform in the model spaces. For example, Figure 5.a is the Hough transform[20,



a b

Figure 5: Results of Hough transforms on the signal in Fig. 2).a a). on the line model space (s, ρ), b). in the circle model space (ξ, η, γ).

4] in the line space (i.e. plane $\theta = (s, \rho)$). The crosses are detected as candidates $\theta^{(1)}_{\text{line}}, \theta^{(2)}_{\text{line}}, ..., \theta^{(N_{\text{line}})}_{\text{line}}$. Figure 5.b is the Hough transform results on the circular arc space $\theta = (\xi, \eta, \gamma)$ with bounds. The balls are candidate circles $\theta^{(1)}_{\text{arc}}, \theta^{(2)}_{\text{arc}}, ..., \theta^{(N_{\text{arc}})}_{\text{arc}}$ with the sizes representing the weights (total number of votes received). Thus, when we propose a new model for an interval $[a, b)$, we compute the *importance proposal probability* by Parzen windows centered at the candidates.

$$q(\theta \,|\, \ell, [a, b)) = \sum_{i=1}^{N_\ell} \omega_i G(\theta - \theta^{(i)}_\ell), \quad \ell \in \{\text{line, arc}\}.$$

ω_i is the accumulated weights voted from the data in $[a, b)$.

2. Edge detection in the x space. For example, Figure 2.b shows the result of an edge strength $f(x|\nabla G * \mathbf{I}, \nabla^2 G * \mathbf{I})$ based on two filters: the 1st and 2nd derivatives of Gaussians. Instead of making a hard decision which is bound to be unreliable, we treat the strength measure as a probability. Thus the proposal for changing point is

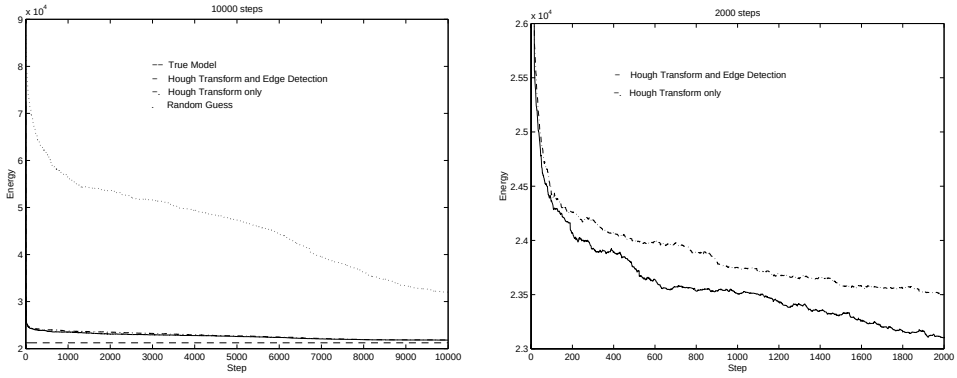
$$q(x \,|\, [a, b)) = \frac{f(x|\nabla G * \mathbf{I}, \nabla^2 G * \mathbf{I})}{\int_a^b f(x|\nabla G * \mathbf{I}, \nabla^2 G * \mathbf{I}) dx}$$

At present we are not able to link the design of $q()$'s to the convergence rate analytically. Thus we seek empirical comparison.[2] An ensemble of 100 1D range data (like Fig. 2) are

[2]The simulation on this pilot example was implemented by a MS student Qiming Luo, and the results were first reported in a unpublished technical report (Zhu, Luo and Zhang, 1999).

simulated randomly with the truth segmentation (global minimum) known. Three Markov chain designs are compared over the 100 1D range data.

- MCMC I: use uniform distributions for $q()$'s, no data-driven heuristics.
- MCMC II: use Hough transform results for $q(\theta \mid \ell, [a, b))$ and uniform distribution for $q(x|[a, b))$.
- MCMC III: use both Hough transform and edge detections for proposals.



a b

Figure 6: The energy curves of MCMC II (thin) and MCMC III (thick) averaged over 100 randomly generated signals in a). 10,000 steps, and b). 2,000 steps.

Figure 2.d displays the optimal solution W^* found by MCMC III. Figure 3 shows the energy $E(W)$ against running time for the input in Figure 2.a by the thin curve (MCMC II) and thick curves (MCMC III). Figure 6 plots the energy changes averaged over 100 signals for $10,000$ steps, the energy jumps disappear because of averaging. The dotted curve is for MCMC I, the dash-dotted curve is for MCMC II, and the solid curve is for MCMC III. The bottom line is the average "true" global optimal energy. Figure 6.b is a zoom-in view of the first 2,000 steps of MCMC II and MCMC III.

To summarize, the importance proposal probabilities exponentially improve the convergence speed. In this experiment, the improvement is mostly from the Hough transform as the edge detection heuristics has rather high entropy (see Fig. 2.b). The so designed jump-diffusion process is capable of finding nearly global minima regardless of initial states.

3 Bayesian Formulation: integrating cues, models and prior

In this section, we formulate the problem of 2D range segmentation and surface reconstruction under the Bayesian framework by integrating two cues, five families of surface models, and various prior models.

3.1 Problem formulation

We denote an image lattice by $\Lambda = \{(i,j): \ 0 \le i \le L_1, 0 \le j \le L_2\}$. A ranger scanner captures two images. One is the 3D range data which is a mapping from lattice Λ to a 3D point,

$$\mathbf{D}: \Lambda \to \mathcal{R}^3, \qquad \mathbf{D}(i,j) = (x(i,j), y(i,j), z(i,j)).$$

(i,j) indexes a laser ray that hits a surface point (x,y,z) and returns. The other is a reflectance map

$$\mathbf{I}: \Lambda \to \{0, 1, ..., G\}$$

$\mathbf{I}(i,j)$ is the portion of laser energy returned from point $\mathbf{D}(i,j)$. $\mathbf{I}(i,j)$ measures some material properties. For example, surfaces of high specularities, such as glass, ceramics, metals, appear dark in $\mathbf{I}$. $\mathbf{I}(i,j) = 0$ for mirrors and surfaces at infinity, such as the sky. $D(i,j)$ is generally very noisy and thus unreliable when $\mathbf{I}(i,j)$ is low, and is considered a missing point if $\mathbf{I}(i,j) = 0$.

The objective is to partition the image lattice into an unknown number of K disjoint regions,

$$\Lambda = \cup_{n=1}^{K} R_n, \quad R_n \cap R_m = \emptyset \quad \forall m \neq n.$$

As natural scenes contain objects (or surfaces) of different types, like the 1D example, at each region R, the range data fit to a surface model of type ℓ^D with parameter Θ^D and the reflectance fit to a reflectance model of type ℓ^I with parameter Θ^I. Thus a *solution* is denoted by

$$W = (K, \ \{R_i: \ i = 1, 2, ..., K\}, \ \{(\ell_i^D, \Theta_i^D), (\ell_i^I, \Theta_i^I): \ i = 1, 2..., K\}).$$

The objective is to maximize a posterior probability over a solution space $\Omega \ni W$,

$$W^* = \arg\max_{\Omega \ni W} p(W|\mathbf{D}, \mathbf{I}) = \arg\max_{\Omega \ni W} p(\mathbf{D}, \mathbf{I}|W)p(W).$$

In practice, two regions R_i, R_j may share the same surface model but with different reflectance, that is, $(\ell_i^D, \Theta_i^D) = (\ell_j^D, \Theta_j^D)$ but $(\ell_i^I, \Theta_i^I) \neq (\ell_j^I, \Theta_j^I)$. For example, a painting

or a piece of cloth hung on a wall, a thin book or paper on a desk, may fit to the same surfaces as the wall or desk respectively, but they have different reflectance. It is also possible that $(\ell_i^D, \Theta_i^D) \neq (\ell_j^D, \Theta_j^D)$ but $(\ell_i^I, \Theta_i^I) = (\ell_j^I, \Theta_j^I)$. To minimize the coding length [45] and to pool information from pixels over large areas, we shall allow adjacent regions to share either depth or reflectance models. Thus a boundary between two regions could be labelled as a *reflectance boundary*, a *depth boundary*, or both.

In the following, we briefly describe the likelihood model $p((\mathbf{D}, \mathbf{I})|W)$ and the prior probability $p(W)$.

3.2 Likelihood coupling a mixture of surface and reflectance models

In the literature, there are many ways for representing a surface, such as implicit polynomials [5, 17], superquadrics [28], and other deformable models. In this paper, five types of surface models are chosen to account for various shapes in natural scenes. New models can be added under the same formulation and algorithm.

1. Family D_1: planar surfaces with unit normal (a, b, c) and interception d,

$$ax + by + cz = d; \quad a^2 + b^2 + c^2 = 1.$$

   Thus it is specified by three parameters $\Theta = (a, b, d)$. We denote by $\Omega_1^D \ni \Theta$ as the space of all planes.

2. Family D_2: conic surfaces – spheres, ellipsoids, cylinders, cones, and tori for many man-made objects and parts. We adopt the representation in (Marshal et al. 2001). These surfaces are specified by 7 parameters $\Theta = (\varrho, \varphi, \vartheta, k, s, \sigma, \tau)$. We refer to [25] for detailed discussions and fitting methods. We denote by $\Omega_2^D \ni \Theta$ the space of family D_2.

3. Family D_3: B-spline surfaces with 4 control points. As surfaces in a natural scene have a broad range of sizes and orientation, we choose a reference plane $\rho : ax+by+cz = d$ which approximately fits to the surface normal. Then a rectangular domain $[0, \delta] \times [0, \phi]$ is adaptively defined on ρ to just cover the surface indexed by two parameters (u, v). In practice, a domain much larger than the surface will be hard to control. Then a grid of $h \times w$ control points are chosen on this rectangular domain, and a

B-spline surface is

$$s(u,v) = \sum_{s=1}^{h} \sum_{t=1}^{w} p_{s,t} B_s(u) B_t(v),$$

where $p_{s,t} = (\eta_{s,t}, \zeta_{s,t}, \xi_{s,t})$ is a control point with $(\eta_{s,t}, \zeta_{s,t})$ being coordinates on ρ and $\xi_{s,t}$ is the degree of freedom at a point. By choosing $h = w = 2$, a surface in D_3 is specified by 9 parameters $\Theta = (a, b, d, \delta, \phi, \xi_{0,0}, \xi_{0,1}, \xi_{1,0}, \xi_{1,1})$. We denote by $\Omega_3^D \ni \Theta$ the space of family D_3.

4. Family D_4: B-spline surfaces with 9 control points. Like D_3, it is a reference plane ρ and a 3×3 grid. It is specified by 14 parameters $\Theta = (a, b, d, \delta, \phi, \xi_{0,0}, ..., \xi_{2,2})$.

5. Family D_5: cluttered surfaces. Some objects in natural scenes, such as trees and bushes have very noisy range depth. To the best of our knowledge, there is no effective models in the literature for such surfaces. Motivated by the success of non-parametric intensity histogram in intensity and texture modeling [34], we adopt a non-parametric 3D histogram for this kind of surfaces. It is specified by $\Theta = (h_1^u, h_2^u, ..., h_{L_u}^u, h_1^v, h_2^v, ..., h_{L_v}^v, h_1^w, h_2^w, ..., h_{L_w}^w)$, where L_u, L_v *and* L_w are the number of bins on u, v, w directions respectively. We denote by $\Omega_5^D \ni \Theta$ the space of family D_5.

Fig 7 displays some typical surfaces for the five families.

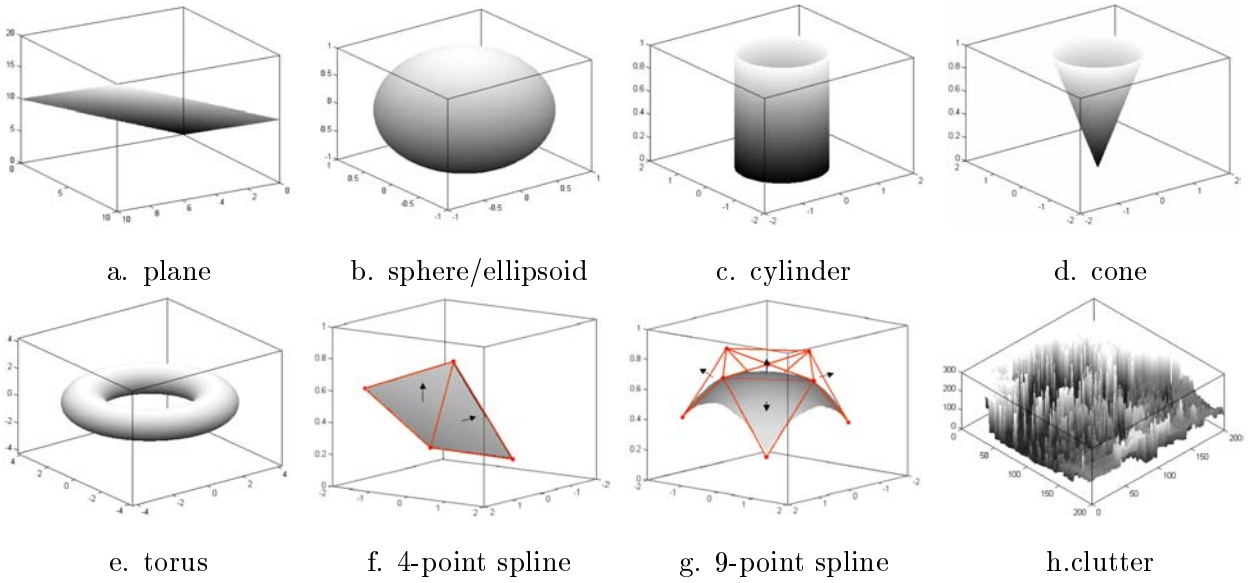
For the reflectance image $\mathbf{I}$, we use three families of models, denoted by $\Omega_i^I, i = 1, 2, 3$ respectively.

1. Family $\mathbf{I}_1$: regions with constant reflectance $\Theta = \mu \in \Omega_i^I$. They represent most of the surfaces with uniform material properties, or surfaces where range data is missing and $\mathbf{I}$ is close to zero.

2. Family $\mathbf{I}_2$: regions with smooth variation of reflectance, modelled by a B-spline model as in family D_3.

3. Family $\mathbf{I}_3$: This is a cluttered region with a non-parametrichistogram $\Theta = (h_1, h_2, ..., h_L)$ for its intensity with L being the number of bins.

For the surface and reflectance models above (except the histogram models), the likelihood model for a solution W assumes the fitting residues to be Gaussian noise subject to



a. plane b. sphere/ellipsoid c. cylinder d. cone

e. torus f. 4-point spline g. 9-point spline h.clutter

Figure 7: Some typical surfaces for the five families of surfaces.

some robust statistics treatment[6],

$$p(\mathbf{D}, \mathbf{I} \,|\, W) \propto \prod_{n=1}^{K} p(D_{R_n}, \mathbf{I}_{R_n} \,;\, (\ell_n^D, \Theta_n^D), (\ell_n^I, \Theta_n^I)) \tag{7}$$

$$\propto \prod_{n=1}^{K} \exp\{- \sum_{(i,j)\in R_n} E(D(i,j), \mathbf{I}(i,j);\ (\ell_n^D, \Theta_n^D), (\ell_n^I, \Theta_n^I))\}. \tag{8}$$

At each pixel (i, j) in R_n, the data energy $E_{i,j} = E(D(i,j), \mathbf{I}(i,j);\ (\ell_n^D, \Theta_n^D), (\ell_n^I, \Theta_n^I))$ is the squared distance from the 3D point $\mathbf{D}(i,j) = (x(i,j), y(i,j), z(i,j))$ to the fitting surface $S(\ell_n^D, \Theta_n^D)$ plus the fitness distance of reflectance $\mathbf{I}(i,j)$ to the reflectance model $J(\ell_n^I, \Theta_n^I)$.

$$E_{i,j} = d^2(\mathbf{D}(i,j), S(\ell_n^D, \Theta_n^D)) \cdot \delta_{\mathbf{I}(i,j)\geq\tau)} - d^2(\mathbf{I}(i,j), J(\ell_n^I, \Theta_n^I))$$

The depth data $D(i,j)$ is considered missing if the reflectance $\mathbf{I}(i,j)$ is lower than a threshold τ, i.e $\delta(\mathbf{I}(i,j) \geq \tau) = 0$.

In practice, we use a robust statistics method to handle outliers[6]. We adopt a two-step procedure. Firstly, we truncate points that are less than 25% of the maximum error; Secondly, truncate points at trough or plateau. Furthermore, the least median of squares method based on orthogonal distance in [38] has been adopted.

Of course, there are alterative likelihood models for laser radar range data that have been developed by Shapiro, Green, and their colleagues [42, 43] which could also be used

here.

3.3 Priors on surfaces, boundaries and corners

Generally speaking, the prior model $p(W)$ should penalize model complexity, enforce stiffness of surfaces, and enhance smoothness of the boundaries, and form canonical corners.

In this paper, the prior model for W is

$$p(W) = p(K)p(\pi_K)\prod_{n=1}^{K} p(\ell_n^D)p(\Theta_n^D|\ell_n^D)p(\ell_n^I)p(\Theta_n^I|\ell_n^I).$$

$\pi_K = (R_1, ..., R_K)$ denotes a K-partition of the lattice Λ. which forms a planar graph with K faces for the regions, a number of M edges for boundaries, and N vertices for corners,

$$\pi_K = (R_k, k = 1, ..., K; \quad \Gamma_m, m = 1, ..., M; \quad V_n, n = 1, ..., N)$$

Thus $p(\pi_K) = \prod_{k=1}^{K} p(R_K) \prod_{m=1}^{M} p(\Gamma_m) \prod_{n=1}^{N} p(V_n)$ since R_k, Γ_m, and V_n are being treated as independent here. We find that some previous prior models used by Leclerc and Fischler [24] for computing 3D wireframe from line drawings quite relevant to our prior models. Leclerc and Fischler used planarity and symmetry of 3D angles at each vertex to recover 3D wireframes from 2D drawings. Of course, natural scenes are much more complex than the "wireframe world". Our prior probability $p(W)$ consists of four parts.

1. *Prior on surface number and sizes for surface model complexity*

It is well known that a higher order model always fits a surface better than a lower order model, but the former could be less stable in the presence of noise. Some conventional model complexity criteria in model selection and merging techniques include MDL (minimum description length)[31], AIC (Akaike Information Criterion) [1], BIC (Bayesian information criterion)[2]. A survey study for range surface fitting is reported in (Bubna, 2000)[8]. According to such criteria, model complexity is regularized by three factors, which penalizes the number of surface K, the number of parameters in each surface model $\#|\Theta|$ respectively.

$$p(K) \propto e^{-\lambda_o K}, \quad p(\Theta_n^D \,|\, \ell_n^D) \propto e^{-\lambda^D \#|\Theta_n^D|}, \quad \text{and} \;\; p(\Theta_n^I \,|\, \ell_n^I) \propto e^{-\lambda^I \#|\Theta_n^I|}, \quad \forall n.$$

However, in our experiments as well as in our previous experiments on segmenting intensity images[34], we observed that such criteria are not appropriate in comparison with human segmentation results. Conventional model complexity criteria, like MDL, are motivated by shortest image coding. But the task of segmentation and image understanding

is very different from coding. The extent to which an object is segmented depends on the importance and familiarity of the object in the scene and the task. In particular, a natural scene contains objects of very broad range of sizes measured by their areas. Unfortunately, it is impractical to define the importance of each types of objects in a general purpose segmentation algorithm. We adopt a statistical model on the surface areas $|R_n|$

$$p(R_n) \propto e^{-\alpha |R_n|^c}, \quad \forall n = 1, 2, ..., K. \tag{9}$$

c being a constant and α being the a scale factor to control the scale of the segmentation. In our experiments, α is the only parameter that is left to be set. All other parameters are set to a value for all experiments.

2. *Prior on B-spline control points for surface stiffness*

For all the B-spline models, a prior is imposed on the control points $\{\xi_{s,t} : \ 0 \leq s, t \leq 2 \text{ or } 3\}$ such that the surface is close to be planar. We triangulate the spline grid on the ρ-plane, and every adjacent three control points form a plane. The prior energy terms enforce the normals of adjacent planes to be parallel to each other. A similar prior was used in the wireframe reconstruction[24].

3. *Prior for surface boundary smoothness*

Due to the heavy noise of the range data along surface boundaries, thus a boundary smoothness prior is adopted, like in the SNAKE [48] or region competition model[39]. Let $\Gamma(s) = (x(s), y(s)), \ s \in [a, b]$ be a boundary between two surfaces,

$$p(\Gamma(s)) \propto \exp\{-\int \phi(\dot{\Gamma}(s)) + \phi(\ddot{\Gamma}(s))ds\}, \ \text{ or } \ p(\Gamma(s)) \propto \exp\{-\int \sqrt{\dot{x}^2(s) + \dot{y}^2(s)}ds.$$

$\phi()$ is a quadratic function with flat tails to account for sharp L-shaped turns in boundaries.

4. *Prior for canonical corners*

A prior is imposed on each vertex V_n by $p(V_n), n = 1, 2, \cdots, N$. Since the natural scene is regular and symmetric in most cases, the angles at a corner should be more or less equalas in [24].

To summarize, the Bayesian framework provides a convenient way for integrating multiple generative models, for coupling two cues, and for introducing prior models. This enables us to deal with complex natural scenes.

4 Computing globally optimal solutions by Jump-diffusion

Obviously the posterior probability is again distributed over a countable number of subspaces of varying dimensions. In the literature of range segmentation, methods, such as edge detection [23], region growing [17, 5], clustering [18, 12], and some energy minimization methods, generalized Hough transforms, can produce useful information, but none of these methods are capable of exploring such complex spaces thoroughly, let alone finding a global optimum.

Our algorithm is a straight-forward extension from the 1D range examples in section (2). It engages six Markov chain jump and diffusion processes. To speed up the MCMC search, we use data clustering in each model space and an edge detection/partition on the lattice Λ. These are discussed in the following three subsections.

4.1 Ergodic Markov chain search by six dynamics

In this subsection, we briefly present the six types of moves/dynamics which form an ergodic Markov chain in exploring the solution space.

Dynamics 1: diffusion of region boundary – stochastic region competition.

Within a subspace of fixed dimension (i.e. the number of surfaces and their models are given), the boundaries evolve according to a region competition equation[39] as a group of stochastic partial differential equations (sPDE). Let $\Gamma_{ij}(s) = (x(s), y(s))$, $s \in (a, b)$ denote the boundary between two regions R_i and R_j, and let $(\ell_i^D, \Theta_i^D, \ell_i^I, \Theta_i^I)$ and $(\ell_j^D, \Theta_j^D, \ell_j^I, \Theta_j^I)$ be the models of the two regions respectively. The motion of curve $\Gamma_{ij}(s)$ follows the following equation[39].

$$\frac{d\Gamma_{ij}(s)}{dt} = -\frac{\delta \log p(W|\mathbf{D}, \mathbf{I})}{\delta \Gamma_{ij}(s)} + \sqrt{2T(t)} dw_t \vec{\mathbf{n}}(s), \quad dw_t \sim N(0, (dt)^2). \tag{10}$$

The Brownian motion is always along the curve normal direction $\vec{\mathbf{n}}(s) = (-\dot{y}(s), \dot{x}(s))/\sqrt{\dot{x}^2(s) + \dot{y}^2(s)})$ at each point s.

To couple the continuous representation of curves Γ_{ij}, we assume the lattice Λ to be a continuous 2D plane. The curve $\Gamma_{ij}(s)$ is involved in three terms in the posterior $p(W|\mathbf{D}, \mathbf{I})$: the smoothness prior and the likelihood on two regions R_i and R_j.

$$\begin{aligned} -\frac{\delta \log p(W|\mathbf{D}, \mathbf{I})}{\delta \Gamma_{ij}(s)} &= -\frac{\delta p(\Gamma_{ij}(s))}{\delta \Gamma_{ij}} - \frac{\delta}{\delta \Gamma_{ij}} \{\log p(\mathbf{D}_{R_i}, \mathbf{I}_{R_i} \,;\, \ell_i^D, \Theta_i^D, \ell_i^I, \Theta_i^I)\} \\ &\quad - \frac{\delta}{\delta \Gamma_{ij}} \{\log p(\mathbf{D}_{R_j}, \mathbf{I}_{R_j} \,;\, \ell_j^D, \Theta_j^D, \ell_j^I, \Theta_j^I)\} \end{aligned}$$

$$
\begin{aligned}
= \quad & \frac{\delta}{\delta\Gamma_{ij}}\{\mu\int_a^b \sqrt{\dot{x}^2(s)+\dot{y}^2(s)}ds\} \\
& + \frac{\delta}{\delta\Gamma_{ij}}\{\int\int_{R_i} -\log p(\mathbf{D}_{R_i}(x,y), \mathbf{I}_{R_i}(x,y)\ ;\ \ell_i^D, \Theta_i^D)\ dxdy\} \\
& + \frac{\delta}{\delta\Gamma_{ij}}\{\int\int_{R_j} -\log p(\mathbf{D}_{R_j}(x,y), \mathbf{I}_{R_j}(x,y)\ ;\ \ell_j^D, \Theta_j^D)\ dxdy\}.
\end{aligned}
$$

By a Green's theorem and an Euler-Lagrange equation, the gradient is

$$
\begin{aligned}
\frac{d\Gamma_{ij}(s)}{dt} \quad = \quad & \{\ -2\mu\kappa(s) + \log\frac{p(\mathbf{D}(x(s),y(s)); (l_i^D, \Theta_i^D))}{p(\mathbf{D}(x(s),y(s)); (l_j^D, \Theta_j^D))} \cdot \delta(\mathbf{I}(x(s),y(s)) \geq \tau) \\
& + \log\frac{p(\mathbf{I}(x(s),y(s)); l_i^I, \Theta_i^I)}{p(\mathbf{I}(x(s),y(s)); l_j^I, \Theta_j^I)} + \sqrt{2T(t)}\frac{dw_t}{dt}\ \}\vec{\mathbf{n}}(s).
\end{aligned}
$$

In the above equations $\kappa(s)$ is its curvature. At each point $(x(s), y(s))$ along the curve, two local log-likelihood ratio tests are done to compare the fitness of the two region models: one for the surface model and the other for the reflectance model. When the range data is less reliable, i.e. $\delta(\mathbf{I}(x(s), y(s)) \geq \tau) = 0$, its log-likelihood ratio test is not used. Thus the two cues are tightly coupled which is desirable in Bayesian cue integration[37].

Dynamics 2: diffusion of vertices.

A vertex $V = (x, y)$ refers to an intersection of more than two regions. It involves some prior model $p(V)$ for canonical corners in previous section, and the curvature is ill-defined at such point. Its diffusion is implemented by the Gibbs sampler[13]. That is, we consider a local lattice, say 3×3 pixels, and randomly select a position subject to the posterior probability. We also implement a corner detection method which may provides bottom-up heuristics for the right position of the vertices. But somehow in experiments we did not observe significant improvement using such heuristics.

Dynamics 3: diffusion of surface and reflectance models.

This is the diffusion of the parameters $((\Theta_n^D, \Theta_n^I))$ for a region R_n, $n = 1, 2, ..., K$ with other variables in W fixed.

$$
\frac{d(\Theta_n^D, \Theta_n^I)}{dt} = \frac{d\log p(\mathbf{D}_{R_n}, \mathbf{I}_{R_n}\ ;\ \ell_n^D, \Theta_n^D, \ell_n^I, \Theta_n^I)}{d(\Theta_n^D, \Theta_n^I)}
$$

Some robust statistics method is used in calculating the gradient and some range pixels do not contribute to the surface fitting if the reflectance is low. We found the Brownian motion may not be necessary in such spaces.

dynamic 4: switching a surface or reflectance model ℓ_n^D or ℓ_n^I.

This is similar to the 1D example but we have more families of model to choose. Suppose at a time instance, a region R_n is selected to switch to a model ℓ_n^D. Then we need some heuristic information for the new model Θ_n^D. The importance proposal probability is calculated, like $q(\phi \mid m)$ in equation (3), based on a number of candidate surfaces pre-computed by a data clustering approach. As we shall discuss below, data clustering is a better method than Hough transform in high dimensional spaces.

Dynamics 5 and 6: split and merge of regions.

Split and merge are a pair of reversible moves to realize the jump process between subspaces. Assume a region R_k with model (Θ_k^D, Θ_k^I) will be split into two regions R_i and R_j with models (Θ_i^D, Θ_i^I) and (Θ_j^D, Θ_j^I). Then the present state of the Markov chain W and the new state W' are

$$\begin{aligned} W &= (K, R_k, (l_k^D, \Theta_k^D), (l_k^I, \Theta_k^I)),\ W_-), \\ W' &= (K+1, R_i, R_j, (l_i^D, \Theta_i^D), (l_i^I, \Theta_i^I), ((l_j^D, \Theta_j^D), (l_j^I, \Theta_j^I),\ W_-). \end{aligned}$$

W_- is the other variables in W that remain unchanged during this jump. The split and merge are proposed with probability $G(W \to W')dW'$ and $G(W' \to W)dW$, while the split move is accepted with probability

$$\alpha(W \to dW') = \min(1, \frac{G(W' \to W)dW p(W'|\mathbf{I})dW'}{G(W \to W')dW' p(W|\mathbf{I})dW}).$$

The merge proposal probability is,

$$G(W' \to W) = q(6)q(R_i, R_j)q(\ell_k^D, \Theta_k^D \mid R_k)q(\ell_k^I, \Theta_k^I \mid R_k).$$

$q(6)$ is the probability for choosing merge move, and $q(R_i, R_j)$ is the probability for choosing R_i, R_j. $q(\ell_k^D, \Theta_k^D \mid R_k)$ is the probability for a new surface model which are selected from a set of bottom-up candidates according to a probability which is summed votes from pixels in the new region R_k.

Similarly, the split proposal $G(W \to W')$ is,

$$q(5)q(R_k)q(\Gamma_{ij}|R_k)q(\ell_i^D, \Theta_i^D \mid R_i)q(\ell_i^I, \Theta_i^I \mid R_i)q(\ell_j^D, \Theta_j^D \mid R_j)q(\ell_j^I, \Theta_j^I \mid R_j).$$

Once R_k is chosen to split, Γ_{ij} is a candidate splitting boundary. In 1D example, this is randomly chosen by an edge strength function. In 2D this is selected from a set of candidate partition pre-computed by edge detection.

In the following, we focus on the computation of two importance proposal probabilities used above: 1). $q(\Gamma \,|\, R)$ – splitting boundary of a region R 2). $q(\ell, \Theta \,|\, R)$ – the new model of a region (surface or reflectance). As we noted before, natural scenes contain objects of broad range of sizes, the bottom-up computation shall be done in multiple scales.

4.2 Coarse-to-fine Edge Detection and Partition

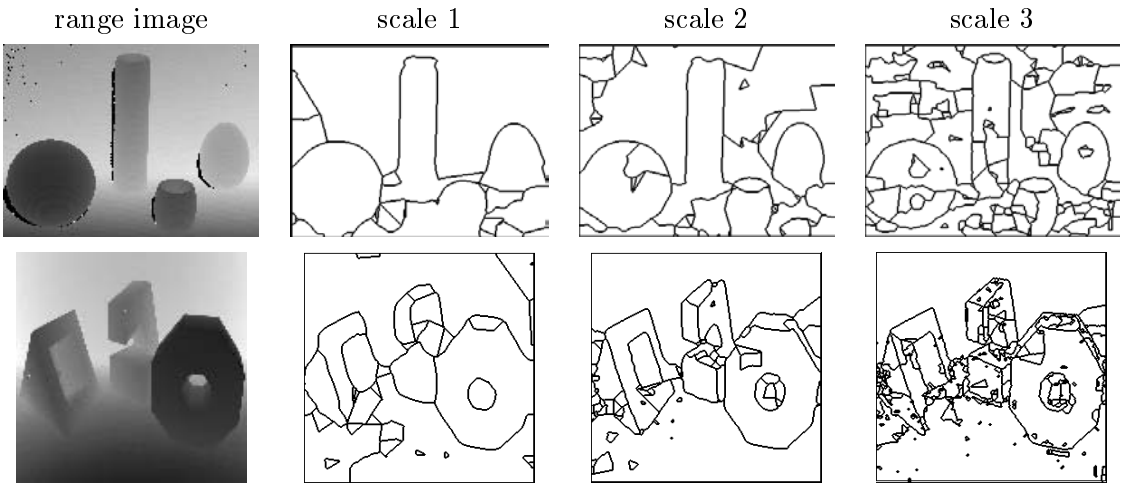
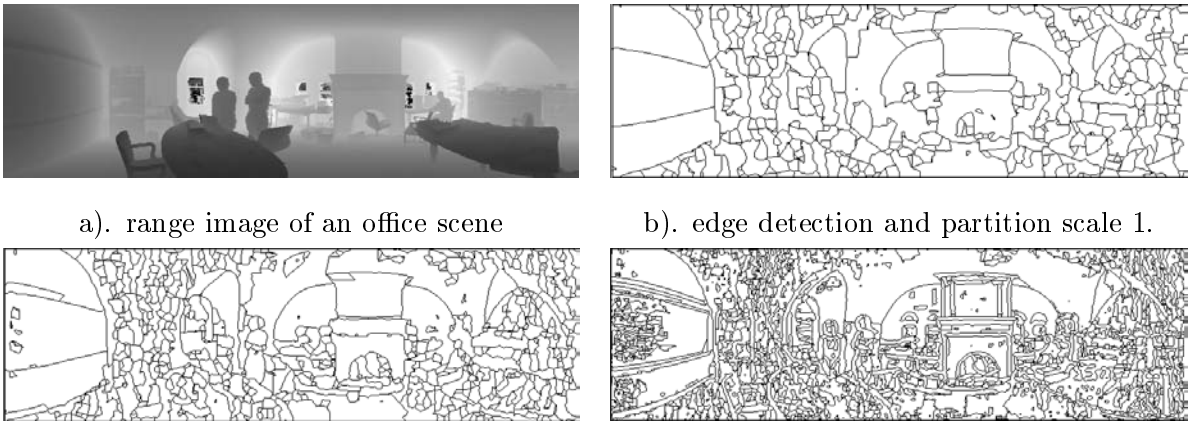


Figure 8: Computed edge maps based on the range cue at three scales for one curved scene and one polyhedral scene in Florida dataset respectively.

In this section, we detect potential edges based on local edge cues, and trace the edges to form a partition of the lattice which will be used as candidate boundaries in splitting regions. We organize the edge maps in three scales according to some edge strength measure. For example, figures 8 and 9 display one example for each of the three database: polyhedra, conics, and real scenes. The edges in figures 8 are based on range data only, while edges in figures 9 combine both range and reflectance measures. We observe that edge detection does provide useful information about the boundaries, especially on occlusion (step edges). However, such local detection is not reliable enough to be the final result, and there are fundamental upper bounds[22] on the errors which one cannot go beyond without involving the global models.

Edges in the reflectance image indicate abrupt changes of surface materials, and are often step edges. In range depth images, edges could be surface discontinuities or surface normal



a). range image of an office scene

b). edge detection and partition scale 1.

c). edge detection and partition scale 2,

d). edge detection and partition scale 3

Figure 9: Computed edge maps at three scales for an office scene.

discontinuities. Because of the noisy nature, the surface normal at each point is estimated over a small window, say a 5×5 patch Δ, by principle component analysis method (see [12]).

Let $\{p_i = (x_i, y_i, z_i) : (m, n) \in \Delta(x, y), i = 1, 2, ..., |\Delta|\}$ be a set of 3D points in a local patch Δ centered at (x, y), and $\bar{p}$ their mass center. One can estimate the local surface normal by minimizing the quadratic error function

$$\mathbf{n}^* = \arg\min_{\mathbf{n}} \mathbf{n}' S \mathbf{n}, \quad \text{with } S = \sum_i (p_i - \bar{p})(p_i - \bar{p})'.$$

$\mathbf{n}^*$ is equal to the eigen-vector of the scatter matrix S which corresponds to the smallest eigen-value $\lambda_{\min}$. With the normal, a local plane $ax + by + cz = d$ ($c = \sqrt{1 - a^2 - b^2}$) is fitted to the patch.

An edge strength is computed on vector space $\mathbf{s} = (a, b, d)$'s of adjacent pixels using a technique of (Nitzberg et al. 1993). Firstly we compute a 2×2 matrix at each point (x, y),

$$\Sigma(x, y) = \int \int_{\Delta(x,y)} \begin{pmatrix} \nabla_x \mathbf{s}^2, & \nabla_x \mathbf{s} \nabla_y \mathbf{s} \\ \nabla_x \mathbf{s} \nabla_y \mathbf{s}, & \nabla_x \mathbf{s}^2 \end{pmatrix} \rho(u - x, v - y)\, dudv, \tag{11}$$

where $\rho(u - x, v - y)$ is a Parzen window centered at (x, y). Let λ_1 and λ_2 ($\lambda_2 \leq \lambda_1$) be the two eigenvalues of the matrix, and v_1 and v_2 the corresponding eigenvectors. Then the edge strength, orientation and cornerness are measured by $e()$, $\theta()$, and c respectively,

$$e(x, y) = \sqrt{\lambda_1 + \lambda_2}, \quad \theta(x, y) = arg(v_1), \quad c(x, y) = \lambda_2.$$

In addition to computing the edge maps from range images, we also apply standard edge detection to the reflectance image and obtain edge maps on three scales. We threshold the edge strength $e()$ at three levels to generate the edge maps shown in figures 8 and 9 after tracing them with heuristic local information to form closed partitions[10].

Given a region R to split, we superimpose R with one of the three edge maps depending on the size of R (large region will use coarse edge partition in general). Then the edge partition within R are candidate sub-regions. Thus the splitting boundaries Γ is chosen at random from a set of candidates. We refer to our previous work on intensity segmentation for detailed formulation[34].

4.3 Coarse-to-fine surface clustering

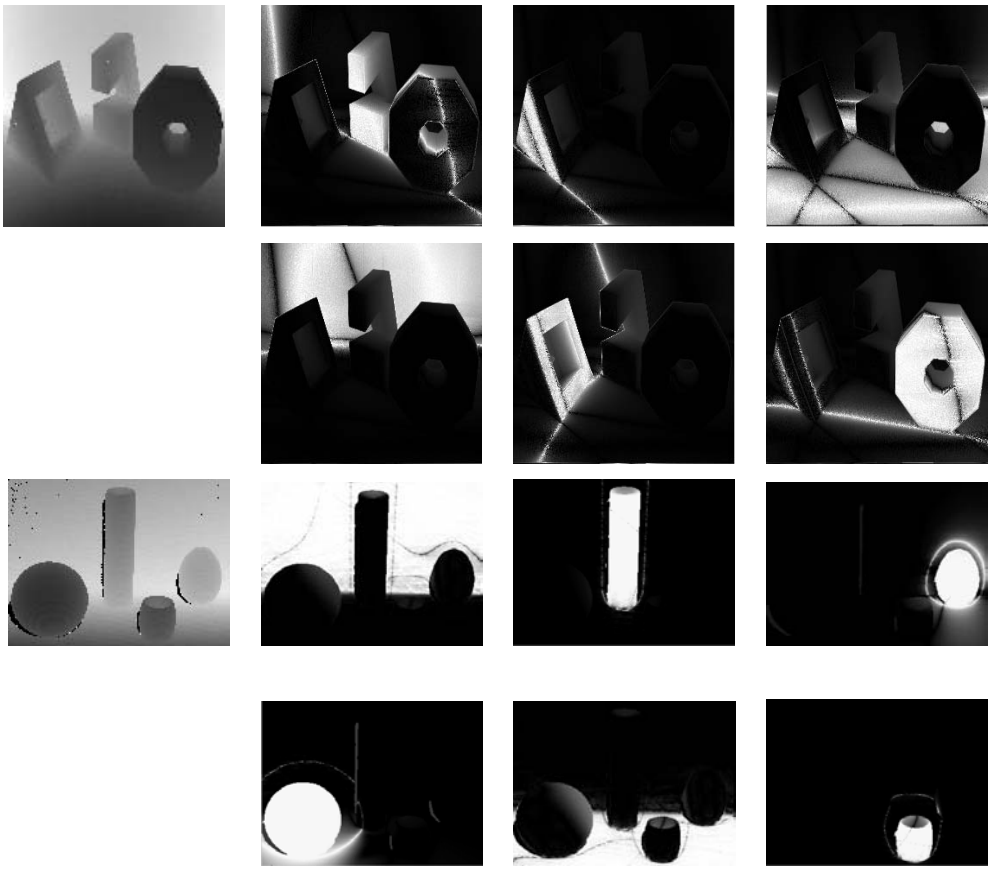


Figure 10: A polyhedra and a conics range images each with six saliency maps for six clustered surfaces.

We compute importance proposal probabilities on the parameter spaces Ω_1^D, Ω_2^D, Ω_3^D, Ω_4^D and Ω_5^D respectively. These probabilities are expressed by a set of candidate surface models in non-parametric forms. But unlike the 1D example represented in section (2), we shall use data clustering instead of Hough transforms for two reasons: 1). Hough transforms become impractical in high dimensional space (say more than three dimensions), 2). Hough transform assume a 2-category detections and thus the peaks (candidates) in the space can be contaminated by each other. In contrast, data clustering is thus more general.

From edge detection in the previous subsection, each small patch Δ is fitted to a local plane (a, b, d) with mass center $\bar{p}$ and the smallest eigen-value $\lambda_{\min}$) of the scatter matrix. Therefore, we collect a set of patches,

$$Q = \{(\Delta_j, a_j, b_j, d_j, \bar{p}_j, \lambda_{\min,j} : j = 1, 2, ..., J = |\Lambda|/\delta^2\},$$

after subsampling the lattice Λ by a factor of δ. In practice, we can discard patches which have relatively large λ_{min}, i.e. patches that are likely on the boundary. We can also use adaptive patch sizes.

The patches in set Q are clustered into a set of C candidate surfaces in all five model spaces

$$\mathcal{C} = \{\Theta_i : \Theta_i \in \Omega_1^D \cup \Omega_2^D \cup \Omega_3^D \cup \Omega_4^D \cup \Omega_5^D, i = 1, ..., C.\}$$

by either the EM-clustering algorithm [49] or the mean-shift clustering algorithm [46, 47]. The EM algorithm is used here and the number of hypothetic clusters in each space is chosen to be excessive.

For example, figure 10 shows six chosen clusters (among many) for a polyhedra scene and a conics scene. Each cluster is associated with a "saliency map" where the brightness at a patch displays the probability that it fits to the cluster (or candidate model). Such probability comes automatically from the EM-clustering. It is very informative in such simple scenes where the models are sufficient to describe the surfaces, and objects have similar sizes.

In natural scene, the results are less satisfactory. Very often small objects, like a book on the top of a desk can be easily assigned to a nearby large objects. To resolve this problem, we compute the clusters in a coarse-to-fine strategy. For example, Fig. 11 shows eight chosen saliency maps for the most prominent clusters in the office scene, which correspond to the floor, desktop, furnace, windows, walls, and ceiling respectively. The total sum of the probability over the lattice is a measure of how prominent a cluster is. Then for patch in Q

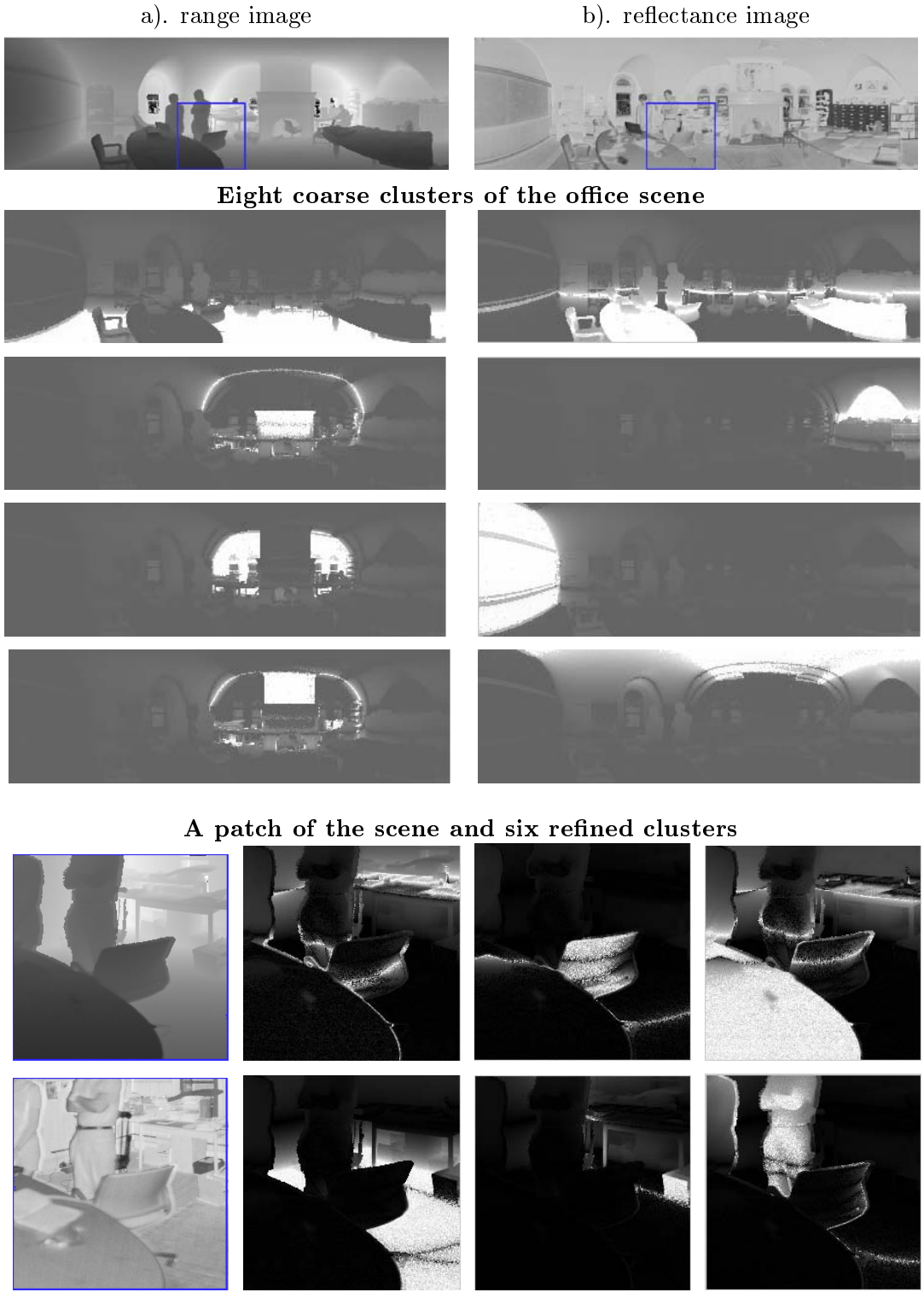


Figure 11: Saliency maps for office scene at two scales. See text for explanation.

which do not fit very well to these prominent clusters, we refine the range data by sub-pixel interpolation, and conduct the clustering on such areas. For example, figure 11 lower panel shows six of the clusters for a sub-area (indicated by a window), such as, people, chairbacks, small box etc.

These candidate models are used to form the importance proposal probabilities as it is in the 1D example. Give a region R, each pixel inside R votes for the candidate models by a continuous probability. Then the proposed model is selected from the candidates proportional to their votes and some random perturbations.

5 Experiments

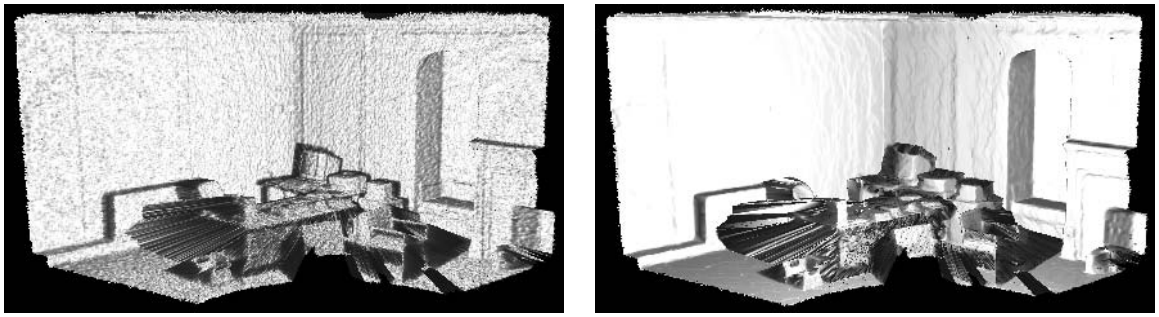
5.1 The datasets and preprocessing

We test the algorithm on three datasets. The first two are the standard Perceptron LADAR camera images and K2T structured light camera images in the USF dataset. The third one is a dataset from Brown University, where images are collected with a long range scanner LMS-Z210 by Riegl. The field of view is 80^0 vertically and 259^0 horizontally. Each image contains 444×1440 measurements with an angular separation of 0.18 degree as Figure 1 shows.

In general, range data are contaminated by heavy noise. Effective preprocessing must be used to deal with all types of errors presented in the data acquisition, while preserving the true discontinuities. In our experiments, we adopt the least median of squares (LMedS) and anisotropic diffusion [36] to pre-process the range data. LMedS is related to the median filter used in image processing to remove impulsive noise from images and can be used to remove strong outliers in range data. After that, the anisotropic diffusion is adopted to handle the general noises while avoiding the side effects caused by simple Gaussian or diffusion smoothing, like the decreasing of the absolute value of curvature and smoothing of orientation discontinuities into spurious curved patch. Fig. 12 shows a surface rendered before and after the preprocessing.

5.2 Results and evaluation

We run the algorithm on three dataset under one parameter setting with only one free parameter c in equation (9) which controls the extent of the segmentation. The algorithm



a) b)

Figure 12: A range scene rendered by OpenGL. a) before and b) after preprocessing.

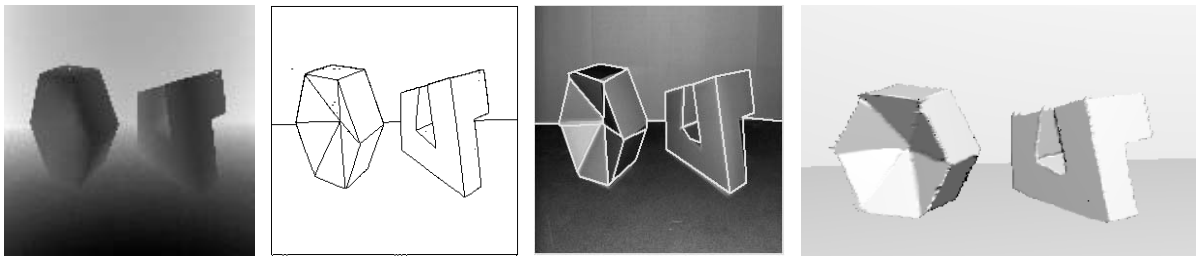
starts with arbitrary initializations.

Figure 13 displays the segmentation results on four images in dataset 1. Figure 14 shows two examples in dataset 2. For the two datasets, we only use range data and the segments are superimposed on the reflectance images. For comparison, we also show in these figures a manual segmentation used in [19] and [30]. In these two figures, we also show the 3D reconstructed scenes based on our segmentation results and the fitted surface models in OpenGL from a novel viewing angle. This is a good way to examine the sufficiency of the models used. In these reconstructed 3D scenes, the background and floor behind the occluding objects are completed using the method discussed next. It is of no surprise that the algorithm can parse such scene very well, because the image models are sufficient to account for the surfaces in these two datasets.

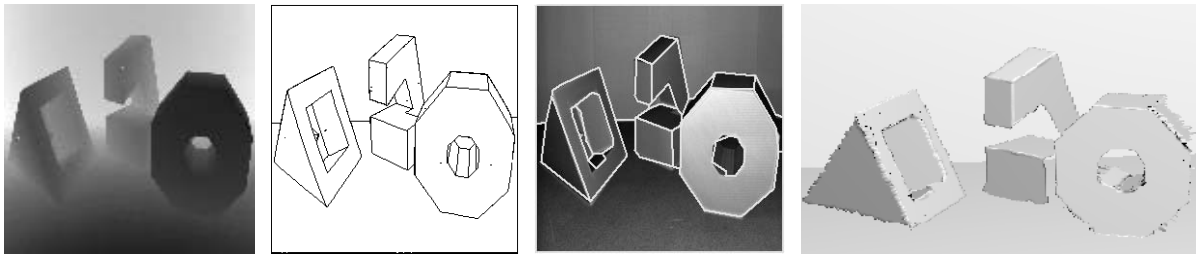
The six examples on the Brown dataset are shown in figures 15 and 16. While the trees are correctly segmented out as shown in Figure 15, we further show the 3D clutter histogram model by one image with very complex tree structures in figure 16. All these results show that the clutter model does well for such cluttered regions.

Range image is often incomplete due to partial occlusion or poor surface reflectance. This can be clearly seen from Fig. 12, in which the floor and the two walls have a lot of missing points. Analysis and reconstruction of range images usually focuses on complex objects completely contained in the field of view; little attention has been devoted so far to the reconstruction of simply-shaped wide areas like parts of wall hidden behind furniture and facility pieces in the indoor scene shown in Fig. 12 [11]. In the reconstructing process, how to fill the missing data points of surfaces behind occlusions is a challenging question.

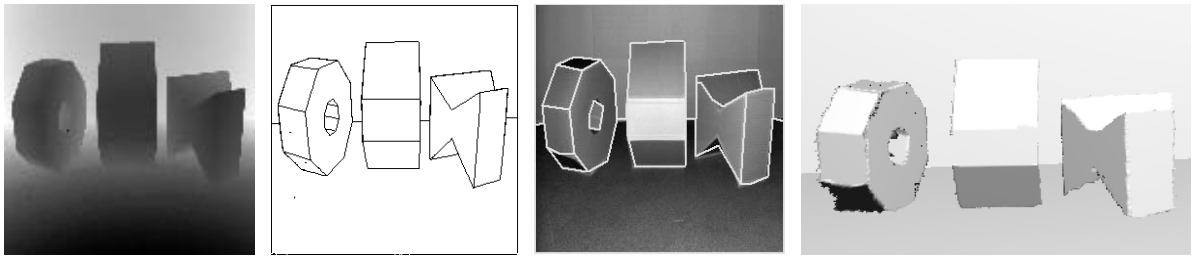
Dataset 1, example 1.



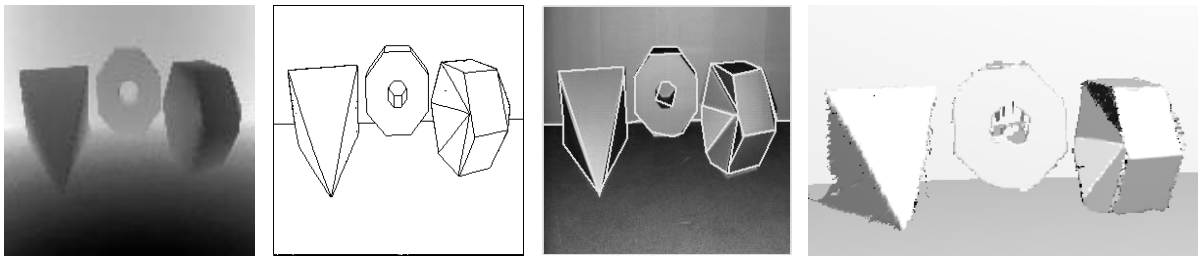
Dataset 1, example 2.



Dataset 1, example 3



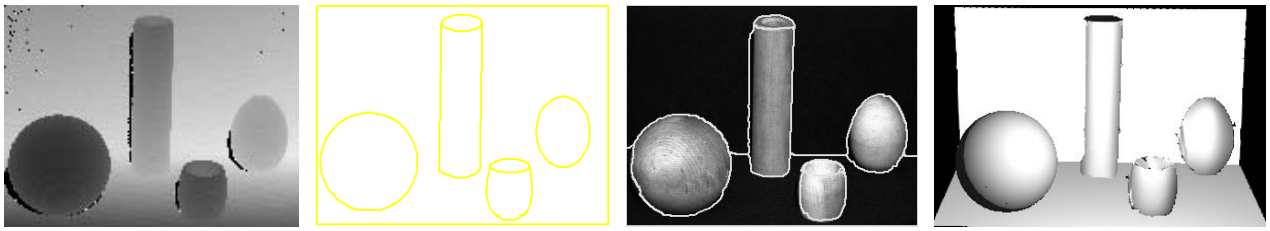
Dataset 1, example 4.



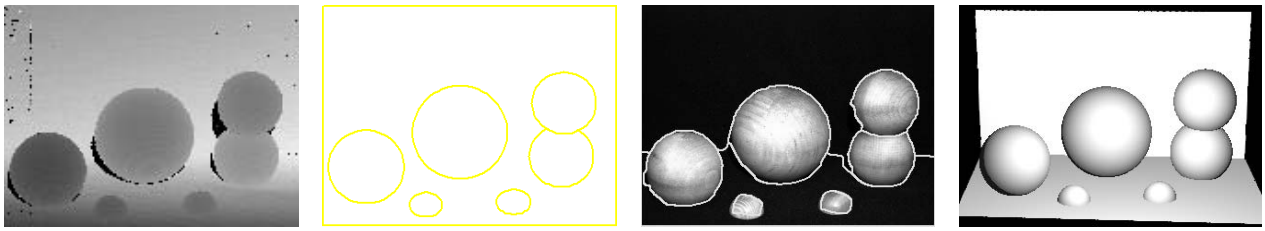
a. range data *b*. manual segment *c*. our result *d*). reconstruction

Figure 13: Segmentation results compared with the manual segments provided in [19]. We only use range data and the segments are superimposed on the reflectance images in c. The reconstructions are shown in slightly different views.

Dataset 2, example 1



Dataset 2, example 2



a. range *b.* manual segment *c.* our result *d.* reconstruction

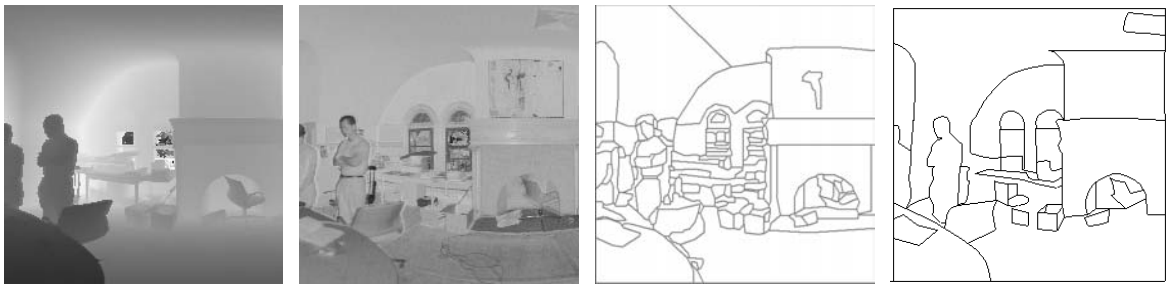
Figure 14: Segmentation on the second dataset compared with manual segment provided in[30]. We only use range data and the segments are superimposed on the reflectance images in c. The reconstruction are generated from novel views.

The completion of these depth information needs higher level understanding of the 3D models. To solve this problem, an algorithm also shall make inference about two things: 1). The types of boundaries as crease, occluding, and so on. 2). The ownership of the boundary to a surface. In our reconstruction procedure, we only use a simple prior model to recover the missing parts of the backgrounds (like the walls and the floor) by assuming they are rectangles. Since we can obtain the needed parameters to represent these rectangles from the segmentation result, it is not difficult to fill the missing points.

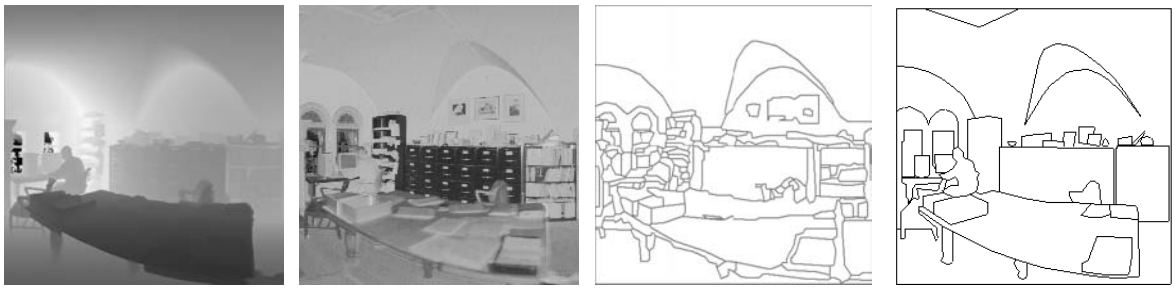
6 Discussion

Our work reassure the representative power of the statistical Bayesian formulation which can couple visual cues, engage many prior models, and incorporate many families of generative models for natural scenes. It also shows that the jump-diffusion process is a general tool for energy minimization in complex solution spaces. Furthermore, the convergence can be accelerated exponentially by bottom-up heuristic information.

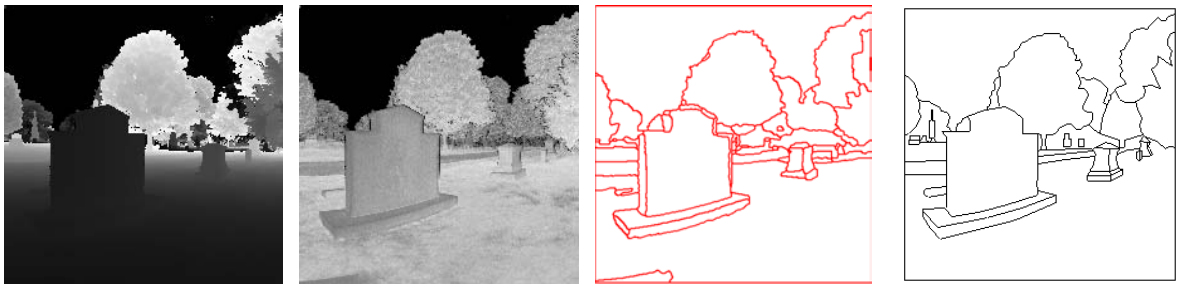
Dataset 3, example 1



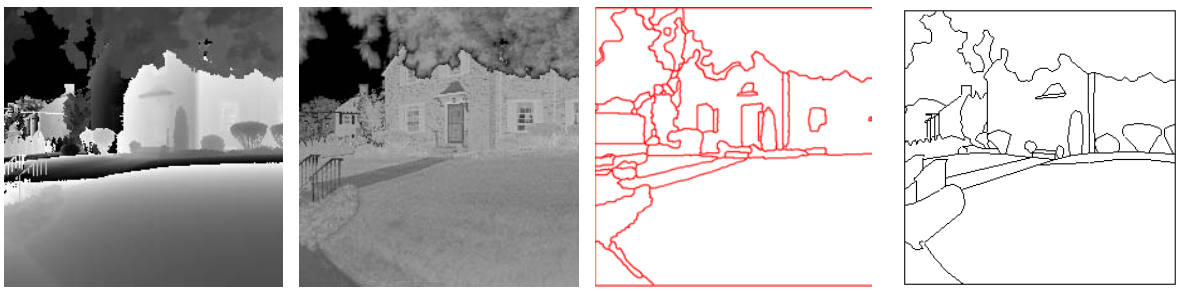
Dataset 3, example 2



Dataset 3, example 3



Dataset 3, example 4



a. range *b.* reflectance *c.* our result *d.* manual segment

Figure 15: Segmentation results for parts of the four scenes in Fig. 1.

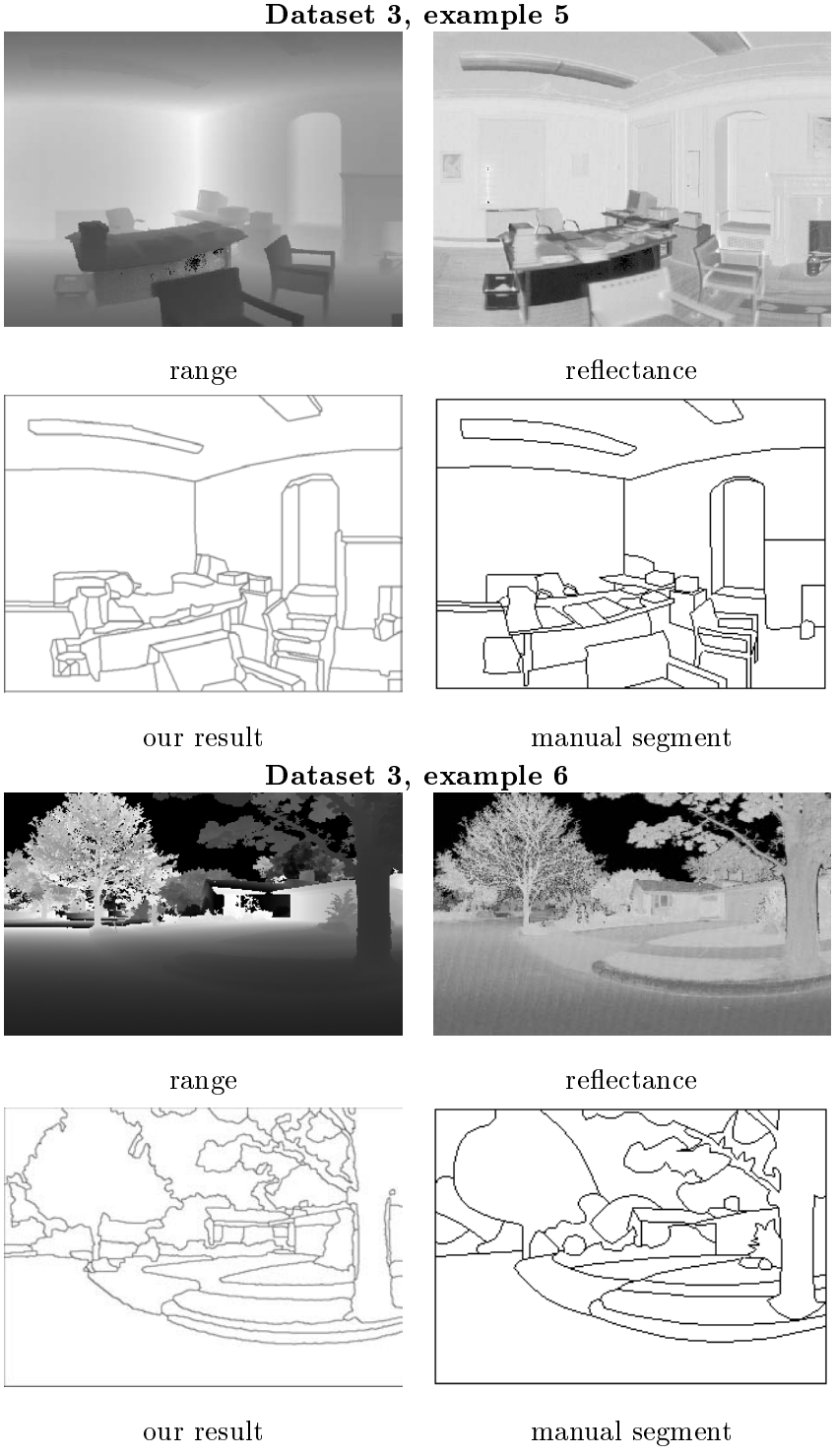


Figure 16: Segmenting the most cluttered part of office B in Fig.1 and a scene with trees.

Some remaining problems that need to be resolved in future research.

1). The algorithm is still time consuming. It currently takes about 1 hour on a pentium IV PC to segment a scene with arbitrary initial conditions. However, we feel there are many engineering methods that can largely reduce the computational time.

2). The experiments reveal that when the models are not sufficient, then the segmentation is not good. For example, the cables in the air and railing on the lane to the door in Fig. 15) are missing. Because they are 1D structures not regions.

3). Better prior model for 3D objects are needed to group surfaces into objects, and therefore to complete surfaces behind the occluding objects.

Acknowledgment

We acknowledge the use of the range datasets from the vision group at University of Southern Florida. We thank Ann Lee and David Mumford for their generous assistance with the Brown dataset. We appreciate a student Qiming Luo for implementing the 1D examples. This work is supported by a ONR grant N-000140-110-535 and a NSF grant IIS-00-92664.

References

[1] H. Akaike, "On entropy maximization principle", in *Applications of Statistics*, P.R. Krishnaiah eds. Amsterdam: North-Holland, 1977.

[2] H. Akaike, "A Bayesian analysis of the minimum AIC procedure", *Annals of the Institute of Statistical Mathematics*, 30A, pp 9-14, 1978.

[3] F. Arman, B. Sabata, and J.K. aggarwal, "Segmentation of 3D range images using pyramidal data structures", *CVGIP: Image Understanding*, 57(3): 373-387, 1993.

[4] D. H. Ballard, "Generalized Hough transform to detect arbitrary shapes", *Pattern Recognition*, 13(2):111-122, 1981.

[5] P.J. Besl and R.C. Jain, "Segmentation through variable order surface fitting", *IEEE Trans. on PAMI*, vol. 10, no.2, pp167-192, 1988.

[6] M. J. Black and A. Rangarajan, "On the unification of line process, outlier rejection, and robust statistics with applications in early vision", *Int'l J. of Comp. Vis.*, Vol. 19, No. 1 pp 57-91. 1996.

[7] P. Bremaud, *Markov Chains – Gibbs fields, Monte Carlo Simulation, and Queues*, (Chapter 6), Springer-Verlag, New York, Inc. 1999.

[8] K. Bubna, "Model selection techniques and merging rules for range data segmentation algorithms", *Computer Vision and Image Understanding*, **80**, pp 215-245, 2000.

[9] F. S. Cohen, W. Ibrahim and C. Pintavirooj "Ordering and parameterizing scattered 3D data for B-spline surface approximation", *IEEE Trans. on PAMI*, Vol. 22, No. 6 pp 642-648. 2000.

[10] I. J. Cox, J. M. Rehg, and S. Hingorani, "A Bayesian multiple-hypothesis approach to edge grouping and contour segmentation", *Int'l J. of Computer Vision*, 11:1, 5-24, 1993.

[11] F. Dell'Acqua and R. Fisher, "Reconstruction of planar surfaces behind occlusions in range images", *IEEE Trans. on PAMI*, 2002 (to appear).

[12] P. J. Flynn and A.K. Jain, "Surface classification: hypothesis testing and parameter estimation", *Proc. of CVPR*, 1988.

[13] S. Geman and D. Geman. "Stochastic relaxation, Gibbs distributions, and Bayesian Restoration of Images", *IEEE Trans. PAMI 6*, pp. 721-741, 1984.

[14] S. Geman and C.R. Huang, "Diffusion for global optimization", *SIAM J. on Control and Optimization*, vol. 24, No.5, pp 1031-1043, 1986.

[15] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination", *Biometrika*, vol.82, 711-732, 1995.

[16] U. Grenander and M.I. Millter, "Representations of Knowledge in Complex Systems", *Journal of the Royal Stat. Soc. Series B*, vol. 56, isuue 4, 1994.

[17] A. Gupta, A. Leonardis, and R. Bajcsy, "Segmentation of range images as the search for geometric parametric models", *Int'l J. of Computer Vision*, 14(3), pp 253-277, 1995.

[18] R. L. Hoffman and A. K. Jain, "Segmentation and classification of range images", *IEEE Trans. on PAMI*, vol. 9, no.5, pp608-620, 1987.

[19] A. Hoover, G. Jean-Baptise, X. Jiang, P.J. Flynn, H. Bunke, D.B. Goldgof, K. Bowyer, D.W. Eggert, A. Fitzgibbon, and R.B. Fisher. "An experimental comparison of range image segmentation algorithms", *IEEE Trans. on PAMI*, vol.18, no.7, pp673-689, 1996.

[20] P. V. C. Hough, "A method and means for recognizing complex patterns", *US Patent* 3,069,654, 1992.

[21] J.G. Huang, A. B. Lee, and D.B. Mumford, "Statistics of range images", *Proc. of CVPR*, Hilton Head, South Carolina, 2000.

[22] S. Kinishi, J. M. Coughlan, A. L. Yuille, and S. C. Zhu, "Fundamental bounds on edge detection: an information theoretic evaluation of different edge cues", *IEEE. Trans. on PAMI*, To appear in 2002.

[23] R. Krishnapuram and S. Gupta, "Morphological methods for detection and classification of edges in range images", *Mathematical Imaging ans Vision*, vol.2 pp351-375, 1992.

[24] Y. G. Leclerc and M.A. Fischler, "An optimization-based approach to the interpretation of single line drawings as 3D wire frame", *Int'l J. of Comp. Vis.*, 9:2, 113-136, 1992.

[25] D. Marshal, G. Lukacs and R. Martin, "Robust segmentation of primitives from range data in the presentce of geometric degeneracy", *IEEE Trans. on PAMI*, Vol. 23, No. 3 pp 304-314. 2001.

[26] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Equations of state calculations by fast computing machines. *J. Chem. Phys.* 21, 1087-92, 1953.

[27] M. Nitzberg, T. Shiota, and D. Mumford. "Filtering, segmentation and depth". *Springer Lecture Notes in Computer Science*, 662, 1993.

[28] A. P. Pentland, "Perceptual organization and the representation of natural form", *Artificial Intelligence*, 28, 293-331, 1986.

[29] D. B. Phillips and A.F.M. Smith, "Bayesian model comparison via jump diffusion", in *Markov Chain Monte Carlo in Practice*, Ed. W.R. Gilks, S.T. Richardson and D.J. Spiegelhalter, Ch.13, Capman and Hall, 1995.

[30] M. W. Powell, K.W. Bowyer, X. Jiang and H. Bunke "Comparing curved-surface range image segmentation", *Proc. of ICCV*, Bombay, India, 286-291, 1998.

[31] J. Rissanen. *Stochastic Complexity in Statistical Inquiry*, World Scientific, Singapore, 1989.

[32] J. Shade, S. Gortler, L.W. He, and R. Szeliski, "Layered depth images", *Proc. of SIGGRAPH*, 231-242, 1998.

[33] L. Tierney, "Markov chains for exploring posterior distributions", *Annal of Statistics*, 22, 1701-28, 1994.

[34] Z.W. Tu, S.C. Zhu and H.Y. Shum, "Image segmentation by data driven Markov chain Monte Carlo." *Proc. of ICCV*, Vancouver, 2001.

[35] Z.W. Tu and S.C. Zhu, "Parsing images into region and curve processes", *Proc. of ECCV*, Copenhagen, DK, May, 2002.

[36] M. Umasuthan and A.M. Wallace, "Outlier removal and discontinuity preserving smoothing of range data", *IEE Proc.-Vis. Image Signal Process*, Vol. 143, No. 3, 1996.

[37] A.L. Yuille and J. J. Clark, "Bayesian models, deformable templates and comptitive priors", in *Spatial Vision in Humans and Robots*, L. Harris and M. Jenkin (eds.), Cambridge Univ. Press, 1993.

[38] Z.Y. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting", *Technical Report of INRIA*, http://www-sop.inria.fr/robotvis/personnel/zzhang/Publis/Tutorial-Estim/node1.html, 1995.

[39] S.C. Zhu and A. L. Yuille. "Region competition: unifying snakes, region growing, and Bayes/MDL for multiband Image Segmentation". *IEEE Trans. PAMI.* vol. 18, No. 9. pp 884-900. 1996.

[40] S.C. Zhu and D.B. Mumford, "Prior learning and Gibbs reaction-diffusion", *IEEE Trans. on PAMI*, vol.19, no.11, pp 1236-1250, Nov. 1997.

[41] S.C. Zhu, Q.M. Luo, and R. Zhang, "Effective statistical inference by data-driven Markov chain Monte Carlo", *Unpublished technical report*, OSU Vision and Learning laboratory, September, 1999.

[42] T.J. Green and J.H. Shapiro, "Maximum-Likelihood Laser Radar Rane Profiling with the Expectation-Maximization Algorithm", *Optical Enginerring*, vol. 31, pp 2343-2354, 1992.

[43] T.J. Green and J.H. Shapiro, "Detecting Objects in 3D Laser Radar Ranege Images", *Optical Enginerring*, vol. 33, pp 865-873, 1994.

[44] A.D. Lanterman, "Jump-Diffusion Algorithm for Multiple Target using Laser Dadar Range Data", *Optical Enginerring*, vol. 40, pp 1724-1728, 2001.

[45] Y.G. Leclerc, "Constructing Simple Stable Descriptions for Image Partitioning", *International Journal of Computer Vision*, vol. 3, no. 1, pp 73-102, 1989.

[46] D. Comaniciu and P. Meer, "Mean shift analysis and applications", *Proc. ICCV*, pp 1197-1203, 1999.

[47] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering", *IEEE Trans. on PAMI*, vol. 17, no. 8, pp 790-799, Aug. 1995.

[48] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: Active Contour Models", *Int'l J. of Computer Vision*, vol. 1, no. 4, pp 321-332, 1988.

[49] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and Texture-Based Image Segmentation Using EM and Its Application to Content-Based Image Retrieval", *Proc. Int'l Conf. Computer Vision*, pp 675-682, 1998.