AUTOMATIC RECOGNITION OF CIVIL INFRASTRUCTURE OBJECTS IN MOBILE MAPPING IMAGERY USING A MARKOV RANDOM FIELD MODEL

Zhuowen Tu[1] and Ron Li[2]
[1]Department of Computer and Information Science
[2]Department of Civil and Environment Engineering and Geodetic Science
The Ohio State University, USA

Working Group II/1

KEY WORDS: Markov Chain Monte Carlo, Gibbs distribution, Object Recognition, Color Image

ABSTRACT

Information technology is increasingly used to support civil infrastructure systems that are large, complex heterogeneous, and distributed. These dynamic systems include communication systems, roads, bridges, traffic control facilities, and facilities for the distribution of water, gas and electricity. Mobile mapping is a new technology to capture georeferenced data. It is, however, still not practical to extract spatial and attribute information of infrastructure objects fully automatically.

In this article, a framework for 3D-object recognition is proposed according to a *viewpoint dependent* theory. A novel system that generates hot-spot maps using color indexing and edge gradient indexing and recognizes traffic lights using MCMC (Markov Chain Monte Carlo) method is proposed. The hot-spot map generation method we developed is much faster than general color image segmentation and thus is practical to be applied in a recognition system. In this approach, both top-down and bottom-up methods are combined by the MCMC engine, which not only recognizes traffic lights but also tells us their poses. This system is robust for different degrees of illumination and rotation.

1 INTRODUCTION

To automatically recognize 3D objects from color images is a challenging problem and has not yet been solved. The recognition of infrastructure objects in outdoor scenes, outside of the controlled laboratory environment, is even more difficult. Different methods of acquiring data and different models (e.g., active vs. passive sensor), may lead to different ways in which the ORS (object recognition system) is formed. We will discuss mainly the recognition of objects in a color image sequence is a color image sequence with georeferencing information captured by the MMS (mobile mapping system) (Li et al. 1999).

We will try to simulate the way in which human beings interpret the scene. The stereo system that human beings use runs very fast and accurately, enabling them to survive in the environment in which they live. “How are 3D objects represented in the human visual system?” (Bulthoff et al. 1994), then becomes the major question we should ask if we want to produce a similar visual system. Different answers to this question will yield different model representations and thus lead to different approaches. There are two common answers to this question: *viewpoint invariant* and *viewpoint dependent*, which yield object-centered and view-centered approaches respectively. The viewpoint invariant answer says that people actually “store” viewpoint-invariant properties of objects in their brains that could be used to match with invariant properties extracted from a 2D image. In this approach, a list of invariant properties, either photometric or geometric, are extracted to match those rooted in 3D objects. The viewpoint dependent answer instead says that multiple views of 3D objects are “stored” to match 2D projections of the 3D objects. Template matching is an old and well-known technology that could be used in a view-centered approach. But it’s impossible to compare a 2D image with an infinite number of views of object using simple template matching. Dickinson et al. (1991) gave a smart framework of how to recognize objects through multiple views. In Bulthoff et al. (1994), the authors made a very good point that if an object-centered reference frame can recover objects independently of their poses, then neither recognition time nor accuracy should be related to the viewpoint of the observer with respect to the objects. If instead the model is viewpoint dependent, then both recognition time and accuracy should be systematically related to the viewpoint of the sensor with respect to the objects. The authors also made a conclusion from psychophysical and computational studies that human beings encode 3D objects as multiple 2D viewpoint representations and achieve subordinate-level recognition by employing a time-consuming normalization process to match objects seen in unfamiliar viewpoints to familiar stored viewpoints. Because matching 3D invariant properties between a 3D model and

a 2D scene is faster than between a number of 2D images of a 3D model viewed at different poses and a 2D scene, we argue here that, although the view-centered approach may be the one humans use, 3D invariant properties are still needed to guide visual systems in searching for the best interpretation.

2 A FRAMEWORK FOR THE VIEW INDEPENDENT APPROACH

Dickinson et al. (1991) proposed a model representation hierarchy that separates 3D models into a finite number of primitives, which are further decomposed into aspects, faces, etc. Here, we expand this hierarchy into a more general framework that will be consistent with most existing ORS's.

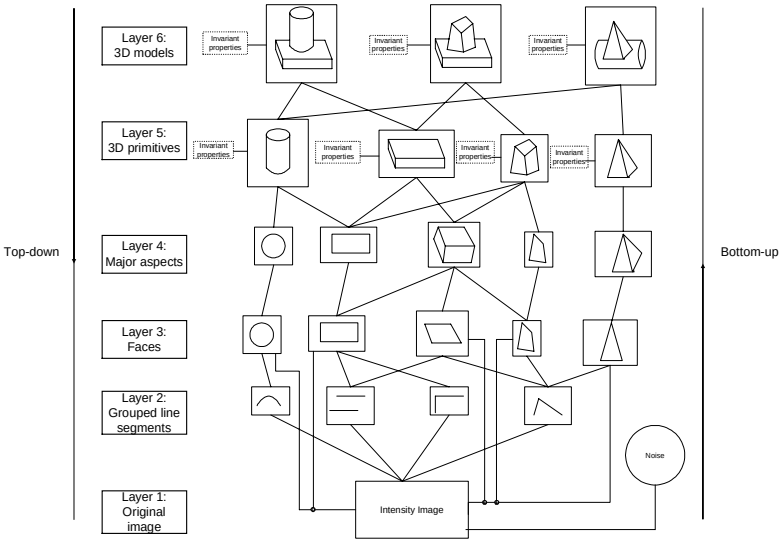


Figure 1. The framework of a model representation. The process starting from layer 1, the original intensity image, followed by edge detection, segmentation, perceptual organization and matching is called the bottom-up approach. The process that works the other way around, starting from layer 6, the 3D model, followed by decompositions and verifications is called top-down approach.

Figure 1 gives a general framework of model representation into which many existing 3D object recognition systems can be fitted. Different systems may have different connections between these various layers, leading to different degrees of complexity and flexibility. Dickinson et al. gives a detailed comparison of primitive complexity, model complexity, search complexity, etc., among different systems. It shows that the 3D volumetric primitive representation method had the best overall performance.

3 INTEGRATING TOP-DOWN AND BOTTOM-UP METHODS FOR TRAFFIC LIGHT RECOGNITION

In this section, we propose a system that integrates top-down and bottom-up processes by MCMC (Markov Chain Monte Carlo) to recognize infrastructure objects, specifically, traffic lights. The framework we discussed in section 2 is used in this system leading to a fast and efficient way of automatic 3D ORS.

3.1 Interpretation of a scene

Computer vision tries to understand the back-projection of 3D scene to a 2D image. Recognition of 3D objects appearing in 2D images requires proper models to represent 2D images and thus to match to a 3D scene. Miller et al. (1995, 1997) gave a basic random model to represent 3D scenes for the recognition of objects by jump-diffusion. Suppose we have detailed 3D models $\{O_i, i = 1 \cdots n\}$ that describe every possible existing object in a 3D scene and each of these models is parametrized by 3D coordinates, pose, etc. Any possible scene x can be denoted as $x \in \chi \subset \cup_{i=1}^{n} \cup_{m=0}^{\infty} O_i^n$ where m is the number of occurances of each type of objects and n is the overall number of objects that appear in the scene. The image data could be denoted as $y \in \mathrm{Y}$ where Y is the observation space. We then have the likelihood function $L(\bullet \mid \bullet) : Y \times X \to \Re$. The likelihood of y given 3D scene x, $L(y \mid x)$, is a conditional

probability. We can further IOP (Interior Orientation Parameter) as $e \in E$. In Bayesian inference problems, the posterior probability density is needed to estimate x given y. The posterior probability is

$$p(x \mid y; e) = \frac{1}{Z(y)} \pi(x) L(y \mid x; e) \quad (1)$$

where $Z(y)$ is the probability of y. To recoginze 3D objects in 2D images, we choose the MAP (Maximize A Posteriori Probability) estimator which finds the x that makes $p(x \mid y; e)$ the global maximum. Since each observed image is just the 2D projection of the 3D scene, we have $Y = \chi \times \Re^3 \times \Re^2_{e_2} + N$, where $\Re^3$ is the 3D transformation, $\Re^2_{e_2}$ is the 2D transformation in which e_2 is the IOP, and N is the imposed noise. Many existing bottom-up methods try to find x, either implicitly or explicitly, with given data y. Among them the indexing of 3D invariants is a straightforward method. The Generalized Hough Transform (GHT) is another way, which tries to find the most significant evidence by voting in χ space according to a given y. Direct indexing (Funt and Finlayson 1995) is straightforward and easy to compute. However, 3D invariants may not always exist. The Hough transform space is actually a rough approximation of $p(x \mid y; e)$ and works only in well defined situations. The method we propose tries to exploit the advantages of indexing and the Hough transform for a fast approximation and then estimate x more accurately using a Markov random process.

3.2 Top-down and bottom-up method

3.2.1 Description of models—traffic lights

To focus on our task, the recognition of traffic lights in outdoor images, we must describe the parameters to be estimated in detail: (1) the *type t*; (2) the *Illuminance* of the shell, red light, yellow light and green light which will be denoted as $c_s(R,G,B)$, $c_r(R,G,B)$, $c_y(R,G,B)$ and $c_g(R,G,B)$ respectively; (3) the *Size* of the primitive (w,h). For each model, we will assume that each type of traffic light is made by several primitives that have identical shape and size; (4) the *Spatial position* (x,y,z); and (5) the *Rotation angles* (ν,κ,φ) in terms of X, Y and Z respectively.

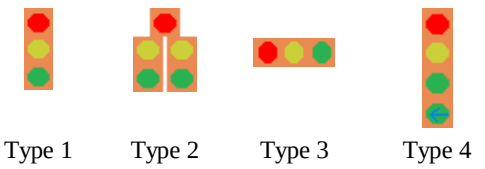


Figure 3. Four typical types of traffic lights that appear the most. Generally speaking, different types of objects should have different parameter spaces to describe them. In the case of traffic lights, however, we have the same number and items of parameters for each type.

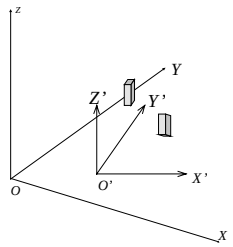


Figure 4. View centered coordinates system (X,Y,Z), local coordinates (X',Y',Z') and occurances of traffic lights.

Figure 4 shows the basic coordinates systems where (X,Y,Z) is the view centered coordinate system and (X',Y',Z') is the local coordinate system. The reason we define the local coordinates is because the occurance of trafic lights shows nice properties that meet our aspect framework in Figure1. Let (ν',κ',φ') be the rotation angles of the traffic lights in terms of the local coordinate system (X',Y',Z'). We may assume that $\nu' = 0$ and $\kappa' = 0$, since traffic lights are always hung perpendicular to ground, and that the rotation angle φ' is close to one of the four major aspects, $0, \frac{1}{2}\pi, \pi \text{ and } \frac{3}{2}\pi$. Suppose the probability distribution of φ' is the summation of four Gaussian distributions. We thus obtain the prior probability of φ' as $p(\varphi') = \frac{f(\varphi')}{Z}$ where

$$f(\varphi') = \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2\sigma^2}(\varphi'-0)^2\} + \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2\sigma^2}(\varphi'-\frac{1}{2}\pi)^2\} + \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2\sigma^2}(\varphi'-\pi)^2\} + \frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2\sigma^2}(\varphi'-\frac{3}{2}\pi)^2\}\} \quad (2)$$

and $Z = \int_0^{2\pi} f(\varphi')d\varphi'$. Let $(\nu_1, \kappa_1, \varphi_1)$ be the rotation angles of the local coordinate system in terms of the view centered coordinate system. We can compute (ν, κ, φ) by $\nu = \nu_1 + \nu'$, $\kappa = \kappa_1 + \kappa'$, and $\varphi = \varphi_1 + \varphi'$ where ν' and κ' could be approximated as 0. However, $(\nu_1, \kappa_1, \varphi_1)$ can be solved by the vanishing points detection method, which will be discussed in the next section.

3.2.2 Vanishing point detection

As we stated above, it's important to know $(\nu_1, \kappa_1, \varphi_1)$ to compute (ν, κ, φ). It is well known that a set of parallel lines in a 3D scene generates a set of lines in a 2D image that converge to a single point, called the *vanishing point*. Although an infinite number of parallel line sets exist in a real scene, in mobile mapping imageries, the dominant directions are along $(\nu_1, \kappa_1, \varphi_1)$. Due to this fact we may derive $(\nu_1, \kappa_1, \varphi_1)$ by extracting the vanishing points in a single image.

As stated in Lutton et al. (1994), let $\vec{U}(\theta_u, \phi_u)$ be the direction of a vanishing point direction in the Gaussian sphere and $\vec{N}_i(\theta_i, \phi_i)$ be the norm of a surface that passes through the origin of the Gaussian sphere and two extremes of straight line segments. Since we know that $\vec{N}_i \bullet \vec{U} = 0$, we have $\cos(\theta_i - \theta_u)\sin\phi_i \sin\phi_u + \cos\phi_i \cos\phi_u = 0$. We omit the detailed algorithm due to space limitations.

(a) (b) (c)

Figure 5. Vanishing points detection algorithm applied in both the color image and gray image, (a) Vanishing point geometry and the corresponding Gaussian sphere, (b) A color image of size 720X400, (c) Voting space of (θ_u, ϕ_u) generated from the straight lines extracted in (b).

We tested this algorithm on many images (both color and gray value) and we found it to be robust under different circumstances. The directions of $(\nu_1, \kappa_1, \varphi_1)$ can be easily computed from a single image.

3.2.3 Overall method

The framework we discussed in section 2 is applied in the recognition of traffic lights. A traffic light consists of several primitives with four major aspects, which can be determined by the vanishing points detection method discussed in 3.2.2. We tested several color image segmentation methods. They were found to be time-consuming with results that were not good enough for further processing. To make our algorithm practical, we developed a new method that integrates bottom-up and top-down methods. The basic strategy is stated below.

Bottom-up approach

- Edges are detected from the color image.
- $(\nu_1, \kappa_1, \varphi_1)$ are computed using the vanishing points detection algorithm.
- Different aspect images of primitives are used as templates in histogram filtering to compute the hot spot map at different scales.
- The Minimal Risk Signal detection method is used to derive the hot spot map as a typical signal detection problem with one image used as a training set.
- Image pieces that contain hot spots are extracted.

Top-down approach

- Markov Chain Monte Carlo method is used to recognize traffic lights and to get the best estimations of parameters that describe each traffic light.

3.2.4 Histogram filtering

For many years, researchers have been trying to recognize objects in color images using color and geometric invariants. Swain and Ballard (1991) initiated a new method called "color indexing" that actually compares histograms of a given

image with those of an object stored in a database in black-white, red-green and blue-yellow spaces. To capture more invariant information, Funt and Finlayson (1995) used the Laplacian and the four directional first derivatives to convolve with the color image and compute the histogram again. Slater and Healey (1996) used local color pixel distributions instead of the whole image to recognize objects. The local color invariants are important to us because we only want to extract those hot spots that are most likely to be traffic lights. To avoid the use of any segmentation method, we developed a new algorithm that captures both photometric and geometric invariants to get hot spot maps using histogram filtering. The algorithm is as follows:

(1) The original color image in (R, G, B) space is transformed into L^*, u^*, v^* space (Wyszecki and Stiles, 1982) to achieve the equal distance propety.

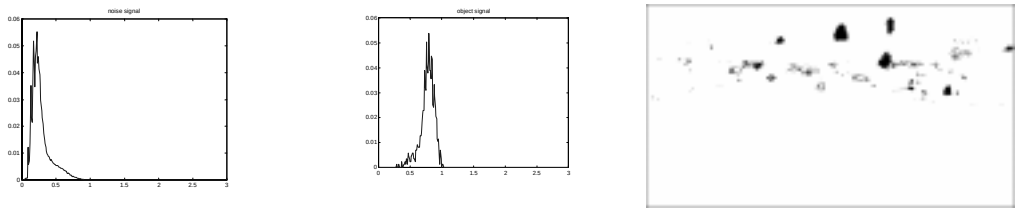
(2) Several 2D image templates, I_i, $i = 1 \cdots n$, are generated as 2D projections of traffic light primitives at major views.

(3) Color template images I_i, $i = 1 \cdots n$ are transformed from (R, G, B) to L^*, u^*, v^* space and histograms are computed as $H_i^{(L^*)}(j) = \frac{1}{Z}\sum_{s \in I_i} \delta(j - L^*(s))$ where j is each bin value in the domain of L^*, $\delta()$ is the Dirac delta function, and Z is the normalization term such that $\sum_j H_i^{(L^*)}(j) = 1$, $H_i^{(u^*)}(j) = \frac{1}{Z}\sum_{s \in I_i} \delta(j - u^*(s))$ where j is each bin value in the domain of u^*, and $H_i^{(v^*)}(j) = \frac{1}{Z}\sum_{s \in I_i} \delta(j - v^*(s))$ where j is each bin value in the domain of v^*. As for geometric invariants, an edge map is obtained using the color edge detection method in Lee and Cok (1991) at $\sigma = 1.0$. A large scale factor is not satisfied because the aspect image is small. The image showing edge pixel can be denoted as $I_i^E(s) = \begin{cases} 1 & s \ is \ an \ edge \ pixel \\ 0 & otherwise \end{cases}$. The histogram of gradients of edge points is computed as $H_i^{(E)}(j) = \frac{1}{Z}\sum_{s \in I_i^E \ and \ I_i^E(s)=1} \delta(j - g(s))$ where j is each bin value in the domain of discrete gradients values, $g(s)$ is the gradient at s, and Z is the normalization term so that

(4) Three square windows that have different sizes, that is, under different scales, are moved convolved with the image. Let $W_1(s')$, $W_2(s')$ and $W_3(s')$ be three windows centered at pixel s'. The histogram of each window centered at every pixel is computed by $H_{W_i(s')}^{(L^*)}(j) = \frac{1}{Z}\sum_{s \in W_i(s')} \delta(j - L^*(s))$, $H_{W_i(s')}^{(u^*)}(j) = \frac{1}{Z}\sum_{s \in W_i(s')} \delta(j - u^*(s))$ and $H_{W_i(s')}^{(v^*)}(j) = \frac{1}{Z}\sum_{s \in W_i(s')} \delta(j - v^*(s))$. Similarly, we have $H_{W_i(s')}^{(E)}(j) = \frac{1}{Z}\sum_{s \in I_{W_i(s')}^E} \delta(j - g(s))$. The histograms are actually Probability Distribution Functions (PDF) that describe the distributions of L^*, u^*, v^* and edge gradients. The overall measurements of the similarity between $H_i^{(L^*)}$ and $H_{W_i(s')}^{(L^*)}$, the similarity between $H_i^{(u^*)}$ and $H_{W_i(s')}^{(u^*)}$, and the the similarity between $H_i^{(v^*)}$ and $H_{W_i(s')}^{(v^*)}$ tell us how likely the template I_i appears at s' with size of W_t. Let the overall photometric similarity be $\hbar(i,t) = \sqrt{D^{L^*}(i,t)^2 + D^{u^*}(i,t)^2 + D^{v^*}(i,t)^2}$ where the distance between two PDF's, $D(i,t)$, may be computed by $D(i,t) = 1 - \|p_i \mid p_t\| = 1 - \sum_j \min(p_i(j), p_t(j))$. Other methods, such as Kullback-Leibler divergence, also could be used to compute the distance between two probability distributions.

(5) The geometric similarity is computed by $\lambda(i,t) = D^E(i,t)$ where the distance between two PDF's are obtained the same way as defined above.

(6) We will use the derived similarity map to determine the possible hot spots. One approach would be to use a thresholding method, with every value that is larger than a fixed threshold set to 1 and every one that is smaller set to 0. However, it is difficult to select the proper threshold. Here, rather than just thresholding, we will approach this problem as a typical signal detection problem in which the noise or signal is determined in terms of some criteria using their probability distributions. With this method, the system could be trained with training data. Again, we omit the details due to space limitations.



(a) (b) (c)

Figure 6. Pdf function of signal vs. noise and the hot spot map for one color image, (a) pdf function of noise, (b) pdf function of object signal, and (c) hot spot map.

We pick up those pixels that appear in samples as object signals and treat all the others as noise so that we have their PDF's as shown in Figure 6. With this method, it is straightforward to generate an object signal map where the dark pixels mean object signal and the bright pixels mean noise. By cominbing these maps at different window sizes and different aspects, we obtain the final hot spot map. This hot spot detection algorithm is robust over different levels of occlusion and illumination.

3.2.5 Traffic light recognition by MCMC in the Top-down approach

Figure 6 (c) shows the final hot spot map from which the candidate regions may be extracted. We here make the assumption that traffic lights showing up in the image don't have any occlusion. This assumption allows us to extract several rectangular pieces of image, each of which encloses a connected hot spot region. The size of every piece of image may be larger than its enclosed region because initially we do not know the exact position and size of the traffic light in the image. The remaining step is to work on every piece of image in which traffic lights are recognized with the best-fitted parameters.

Given every piece of image y and IOP parameters e, we want to find the x that maximizes the posterior probability $\pi(x \mid y;e)$. In this case, traffic lights are generally imaged with the sky as the background and we can assume a simple Gaussian distribution of the background pixels. Suppose the parameter x is composed of $[t, c_s(R,G,B), c_r(R,G,B), c_y(R,G,B), c_g(R,G,B), (x_o, y_o, z_o), (w,h), (\nu, \kappa, \varphi)]$ with each term defined as in 3.2.1. In Ullman and Basri (1991), the authors proved that the perspective projection of a 3D object, when viewed from some distance, could be approximated by an orthogonal projection. We also assume that $\nu = 0$ and $\kappa = 0$, which are true in the real scene. These requirements could be met in our cases reasonably and the parameters may be simplified to $[t, c_s(R,G,B), c_r(R,G,B), c_y(R,G,B), c_g(R,G,B), (x_I, y_I), (w,h), \varphi]$, where (x_I, y_I) are the 2D coordinates of the center of the traffic light in an image piece. Let $F(x,e)$ be the orthogonal projection of a traffic light paramterized by x. Let ${F_{(x,e)}}^{(L^*)}(s)$, ${F_{(x,e)}}^{(u^*)}(s)$ and ${F_{(x,e)}}^{(v^*)}(s)$ be the L^*, u^*, v^* value at each pixel location s in $F_{(x,e)}$. Let $\mu^{(L^*)}$, $\mu^{(u^*)}$ and $\mu^{(v^*)}$ be the average value of L^*, u^*, v^* of the background and $\sigma^{(L^*)}$, $\sigma^{(u^*)}$ and $\sigma^{(v^*)}$ be their corresponding variances respectively. The likelihood is

$$L(y \mid x;e) = \prod_{s \in F_{(x,e)}} \prod_{c=L^*,u^* \text{ and } v^*} \left[\frac{1}{\sqrt{2\pi}\sigma^{(c)}} \exp\left(-\frac{1}{2(\sigma^{(c)})^2} (y^{(c)}(s) - {F_{(x,e)}}^{(c)}(s))^2 \right) \right] \times$$
$$\prod_{s \notin F_{(x,e)}} \prod_{c=L^*,u^* \text{ and } v^*} \left[\frac{1}{\sqrt{2\pi}\sigma^{(c)}} \exp\left(-\frac{1}{2(\sigma^{(c)})^2} (y^{(c)}(s) - \mu^{(c)})^2 \right) \right]. \quad (3)$$

The log likelihood becomes

$$\log(L(y \mid x;e)) = \sum_{s \in F_{(x,e)}} \sum_{c=L^*,u^* \text{ and } v^*} -\frac{1}{2(\sigma^{(c)})^2} (y^{(c)}(s) - {F_{(x,e)}}^{(c)}(s))^2 + \sum_{s \notin F_{(x,e)}} \sum_{c=L^*,u^* \text{ and } v^*} -\frac{1}{2(\sigma^{(c)})^2} (y^{(c)}(s) - {F_{(x,e)}}^{(c)}(s))^2 + g \quad (4)$$

where g is a constant value which equals $m \sum_{c=L^*,u^* \text{ and } v^*} \log\left(\frac{1}{\sqrt{2\pi}\sigma^{(c)}}\right)$ where m is the number of pixels in y. The posterior distribution then becomes

$$p(x \mid y;e) \propto e^{(\log(\pi(x)) + \log(L(y|x;e))/B} \quad (5)$$

where B is the so called "temperature" used for annealing. The introduction of B won't change the x^* that maximizes the posterior probability because the exponentional function is montone. We note here that $p(x)$ is exactly the well-known Gibbs distribution, which was originated by Geman and Geman (1986). We may rewrite the above equation as

$$p(x \mid y;e) \propto e^{-H(x)/B} \tag{6}$$

where $H(x) = -(\log(\pi(x)) + \log(L(y \mid x;e)))$ is the energy function. We could simply denote it as $p(x) = e^{-H(x)/B}$. The Metropolis sampler, specifically the Metropolis-Hastings method, is used here to find the solution to the MAP. The basic Metropolis sampling method is stated in Winkler (1995):

(1) A new configuration x_2 is proposed by sampling from a probability distribution $G(x_1,\cdot)$ on X where $G(x_1,\cdot)$ is called the *proposal matrix.*

(2) The energy at x_2 is computed and is compared with that at x_1

(a) If $H(x_2) \le H(x_1)$ then x_2 is accepted.

(b) If $H(x_2) > H(x_1)$ then x_2 is accepted with the probability $\exp((H(x_1) - H(x_2))/B)$.

(c) If x_2 is not accepted then x_1 will be kept.

The transformation matrix $\pi(x_1, x_2)$ becomes

$$\pi(x_1,x_2) = \begin{cases} G(x_1,x_2)\exp(-(H(x_2)-H(x_1))^+ / B) & if\ x_1 \neq x_2 \\ 1 - \sum_{z \in X \setminus \{x\}} \pi(x,z) & if\ x_1 = x_2 \end{cases} \tag{7}$$

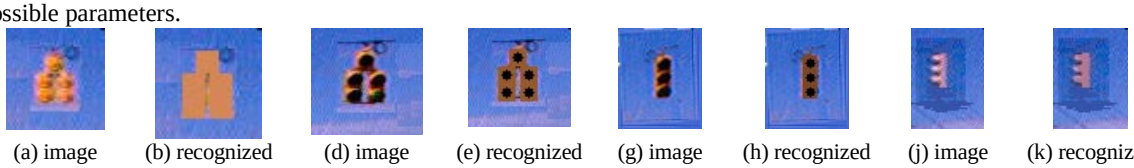
where $(H(x_2)-H(x_1))^+ = \begin{cases} 0 & H(x_2)-H(x_1) \ge 0 \\ -(H(x_2)-H(x_1)) & H(x_2)-H(x_1) < 0 \end{cases}$.

It can be proven easily that $p(x_1)\pi(x_1,x_2) = p(x_2)\pi(x_2,x_1)$, which meets the requirement for the convergence of a Markov Chain. This so-called *detailed balance equation* is crucial because it insures that the Markov Process is reversable. A more efficient implementation of the Metropolis algorithm is the Metropolis-Hastings algorithm whose Markov transformation matrix can be derived by

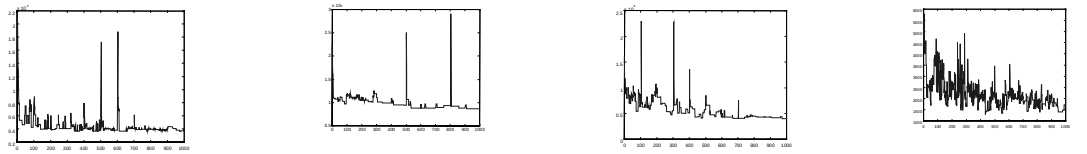
$$\pi(x_1,x_2) = \begin{cases} G(x_1,x_2)A(x_1,x_2) & if\ x_1 \neq x_2 \\ 1 - \sum_{z \in X \setminus \{x_1\}} \pi(x_1,z) & if\ x_1 = x_2 \end{cases} \tag{8}$$

where $A(x_1,x_2) = \min\left\{1, \frac{p(x_2)G(x_2,x_1)}{p(x_1)G(x_1,x_2)}\right\}$.

It is trival to prove the convergence of the Markov random process, $p(x_1)\pi(x_1,x_2) = p(x_2)\pi(x_2,x_1)$. The important thing remaining is to generate proposal matrix $G(x_1,x_2)$. As we stated before, traditional methods like the Generalized Hough Transform, which uses voting for a solution x, may or may not produce the MAP for the given image y. To take advantage of both the speed of the GHT, and the ability of MCMC to search for globally optimal solution, we use the result of the GHT as the proposal matrix $G(x_1,x_2)$. The voting space of GHT actually gives a distribution of every possible parameters.



(a) image piece | (b) recognized model | (d) image piece | (e) recognized model | (g) image piece | (h) recognized model | (j) image piece | (k) recognized model



(c) Energy curve along MCMC of (a) and (b) | (f) Energy curve along MCMC of (d) and (e) | (i) Energy curve along MCMC of (g) and (h) | (l) Energy curve along MCMC of (j) and (k)

Figure 7. Original image pieces with the recognized traffic lights and the energy curve along MCMC. We can see the nice match between the original images and the imposed 3D object. In (c) it takes around 2 minutes to reach the final status. In (f) it takes one and a half minutes. It takes just less than one minute for (i) and (l) to reach the final steps. These image pieces were extracted by the algorithm we proposed in 3.2.4.

4 CONCLUSIONS

In this article, a framework for 3D-object recognition was discussed. Within this framework, we proposed a novel system that integrates bottom-up and top-down approaches by Markov Chain Monte Carlo to recognize traffic lights in real image sequences taken by the Mobile Mapping System. It takes fifteen minutes for the system to recognize the objects in a color image of size 720X400 starting from the low-level processing. The results are promising and this novel system shows the great potential of using the Markov Chain Monte Carlo method for recognizing 3D objects. In this combination of bottom-up and top-down by MCMC, we combined traditional techniques such as indexing and the Generalized Hough Transform together to show that they could be nicely integrated in random processes. Due to space limitations we have shown here only the major ideas and omitted many details.

ACKNOWLEDGMENT

We would like to acknowledge the support from the National Science Foundation (CMS-9812783). We also thank Dr. Song Chun Zhu for stimulating discussions and thoughtful suggestions.

REFERENCES

Brillault-O'Mahony, B., 1991. New Method for Vanishing Point Detection. CVGIP, 54(2), pp. 289-300.

Bulthoff, H.H., Y.E. Shimon and J.T.Michael, 1994. How are 3D objects represented in the brain? A.I. Memo No. 1479, MIT.

Dickinson, S.J., A.P. Pentland and A. Rosenfeld, 1992. From Volumes to Views: An Approach to 3-D Object Recognition. CVGIP: Image Understanding, 55(2), pp. 130-154.

Funt, B.V. and G.D. Finlayson, 1995. Color Constant Color Indexing. IEEE Trans. PAMI, 17(5), pp. 522-529.

Geman, S. and D. Geman, 1984. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. PAMI, 6(6), pp. 721-741.

Lee H.-C. and D.R. Cok, 1991. Detecting Boundaries in a Vector Field. IEEE Trans. Signal Proc. (39), pp.1181-1194.

Li, R., W. Wang and H.-Z. Tseng 1999. Detection and Location of Object from Mobile Image Sequences by Hopfield Neural Networks. Photogrammetric Engineering and Remote Sensing, 65(10), pp.1199-1205.

Lutton, E., H. Maitre, and J. Lopez-Krahe, 1994. Contribution to the Determination of Vanishing Points Using Hough Transform. IEEE Trans. PAMI, 16(4), pp. 430-438.

Miller, M.I., U. Grenander, J.A. O'Sullivan and D.L. Snyder, 1997. Automatic Target Recognition Organized via Jump-Diffusion Algorithms. IEEE Trans. Image Processing, 6(1), pp. 157-174.

Miller, M.I., A. Srivastava and U. Grenander, 1995. Conditional-Mean Estimation Via Jump-Diffusion Processes in Multiple Target Tracking/Recognition. IEEE Trans. Signal Processing, 43(11), pp. 2678-2689.

Modestino, J.W. and J. Zhang, 1989. A Markov random filed model-based approach to image interpretation. In Proceedings of the IEEE CVPR, pp. 458-465.

Salgian, G. and D.H. Ballard, 1998. Visual Routines for Autonomous Driving. Proc. of the 6-th ICCV, pp. 876-882.

Shufelt, J.A., 1996. Projective Geometry and Photometry for Object Detection and Delineation. CMU-CS-96-164.

Slater D. and G. Healey, 1996. The Illumination-Invariant Recognition of 3D Objects Using Local Color Invariants. IEEE Trans. PAMI, 18(2), pp. 206-210.

Swain, M. and D. Ballard, 1991. Color indexing. International Journal of Computer Vision, 7(1), pp. 11-32.

Ullman S. and Basri R., 1991. Recognition by Linear Combinations of Models. IEEE Trans. PAMI, 13(10), pp. 992-1006.

Winkler, G., 1995. Image Analysis, Random Fields and Dynamic Monte Carlo Methods. Springer Verlag.