A Framework for Automatic Recognition of Spatial Features from Mobile Mapping Imagery

Zhuowen Tu and Rongxing Li

Abstract

Mobile mapping is a new technology for capturing georeferenced data. It is, however, still not practical to extract spatial and attribute information of objects such as infrastructure elements fully automatically. In this article, a new framework for 3D object recognition by hypothesis-and-test *techniques is proposed and developed. An example of traffic-light recognition from mobile mapping images is given in detail. The hypothesis is generated according to the* viewpoint dependent *theory. We formulate the hypothesis test problem based on Bayesian inference and, in particular, the MAP (Maximize A Posteriori Probability). This approach functions in two major steps: (1) generation of hot-spot maps by vanishing point detection and template matching, and (2) estimation of the parameters of 3D objects (traffic lights) by Markov Chain Monte Carlo (MCMC). The developed hot-spot map generation method is, in general, faster than general color image segmentation algorithms. For example, it can handle the recognition problem with a color image of 720 by 400 pixels within a couple of minutes rather than tens of minutes to even hours when using the segmentation algorithms. The parameter estimation method uses MCMC to simulate an ergodic stochastic process so that a robust and global optimal solution can be found. The approach shows great potential for automatic object recognition in image sequences acquired by mobile mapping systems.*

Introduction

Automatic recognition of 3D objects from color images is a challenging, yet unsolved, problem. Furthermore, recognition of spatial features from images acquired in outdoor scenes, outside of a controlled laboratory environment, by a mobile mapping system (Li, 1997) poses an even more difficult research topic. The ways in which the data are acquired, for example using active or passive sensors, may affect the methods of object recognition. In this paper, we mainly discuss object recognition from color mobile mapping images and show how traffic lights, in particular, are recognized by the proposed system.

The human stereo vision system is an extremely comprehensive and effective system that functions very fast and accurately to support human decision-making processing in an ever-changing environment. "How are 3D objects represented in the human visual system?" becomes the initial question we ask if we want to produce a similar visual system (Bulthoff *et al.*, 1994). Different answers to this question yield different model representations, and thus lead to different approaches. Two common answers to this question are *viewpoint independent* and *viewpoint dependent* approaches that are further referred to as object-centered and view-centered approaches, respectively. The viewpoint independent approach insists that people actually "store" viewpoint invariant properties of objects in their brains and match them with the invariant properties extracted from a 2D image. In this way, a list of invariant properties, either photometric or geometric, are extracted to match those rooted in 3D objects. The viewpoint dependent answer explains that multiple views of 3D objects are "stored" to match 2D projections of the 3D objects. Template matching is an old and well-known technique that can be used to implement a view-centered approach. However, it is impossible to compare a 2D image with infinite numbers of views of an object by simple template matching. Dickinson *et al.* (1992) introduced an improved framework of object recognition through multiple views. Bulthoff *et al.* (1994) emphasized that, if an object-centered reference frame can recover objects independently from their poses, neither recognition time nor accuracy should be related to the viewpoint of the observer with respect to the objects. Otherwise, in a viewpoint dependent model, both recognition time and accuracy should be systematically related to the viewpoint of the sensor with respect to the objects. It was further concluded that, based on psychophysical and computational studies, 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 with those stored in familiar viewpoints. However, from a computational point of view, matching 3D invariant properties between a 3D model and a 2D scene would be much more efficient than that between a large number of 2D images of a 3D model viewed at different poses and the 2D scene. The view-centered approach might be the one that humans use, and it is also the basis for this research that uses very few poses, although 3D invariant properties may ultimately have the potential to guide a more effective visual system for object recognition.

A Framework of the View-Dependent Approach

Dickinson *et al.* (1992) proposed a hierarchy of model representations that organizes 3D models based on a finite number of primitives, which are further decomposed into aspects and faces. In this research we expand this hierarchy into a more general framework for an object recognition system (ORS) as shown in Figure 1.

Z. Tu is with the Department of Computer and Information Science and R. Li is with the Department of Civil and Environmental Engineering and Geodetic Science, both at the Ohio State University, Columbus, OH 43210 (li.282@osu.edu).

Photogrammetric Engineering & Remote Sensing
Vol. 68, No. 3, March 2002, pp. 267–276.

0099-1112/02/6803–267$3.00/0

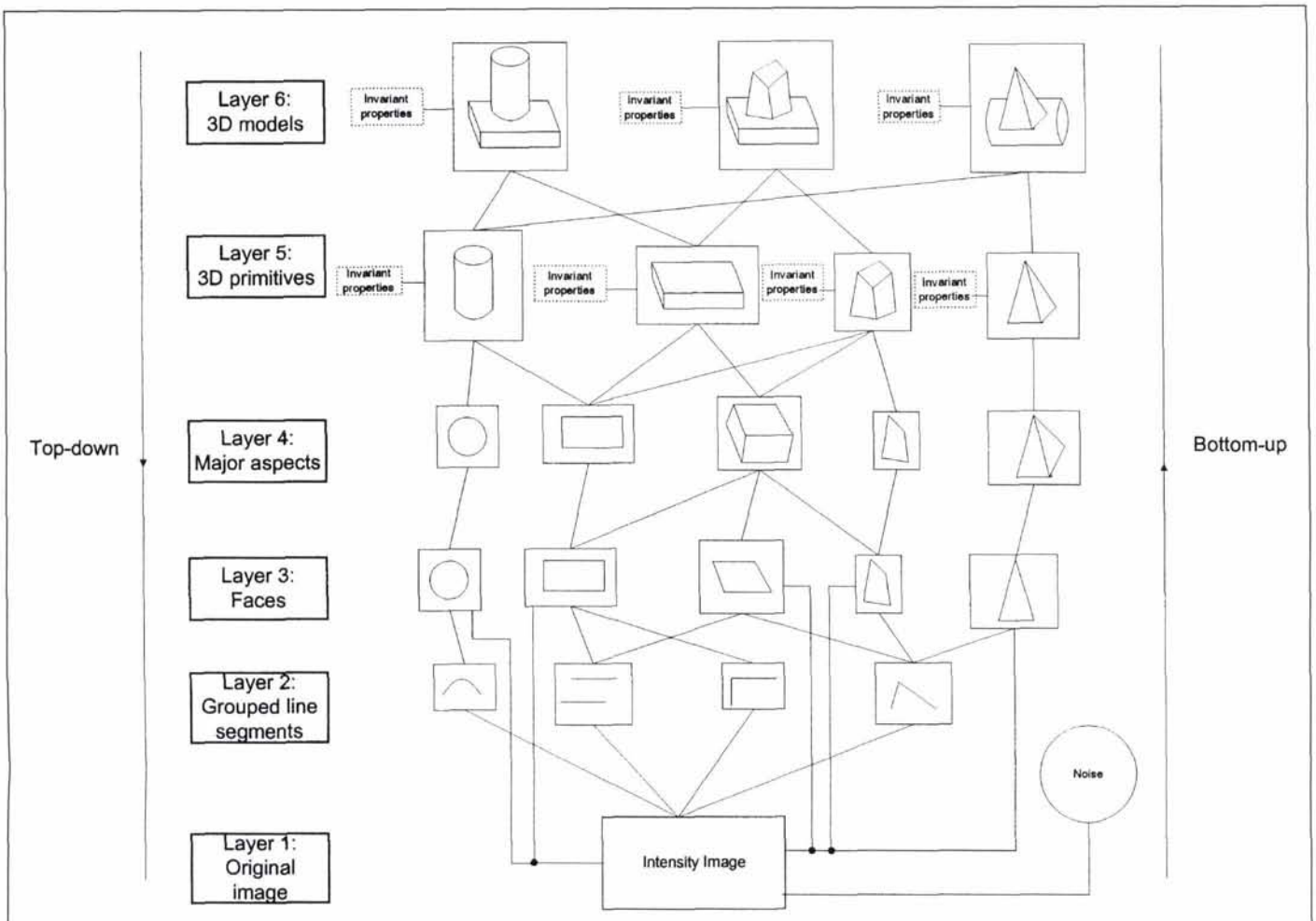


Figure 1. A hierarchy of model representations.

A bottom-up approach starts from Layer 1, the original intensity image, followed by edge detection, segmentation, perceptual organization, and matching. The process that works in the reverse order, starting from Layer 6, the 3D model, followed by decompositions and verifications with the original images, is called the top-down approach. Various ORSs may have different connections between these various layers and result in various degrees of complexity and flexibility. Dickinson *et al.* (1992) gave a detailed comparison of the primitive complexity, model complexity, and search complexity of the systems. It was shown that the 3D volumetric primitive representation method has the best overall performance.

Scene and Model Interpretation

Interpretation of a Scene

Recognition of 3D objects appearing in 2D images requires proper models to represent 2D images, and thus to match with a 3D scene. Miller *et al.* (1995; 1997) gave a basic random model to represent 3D scenes for object recognition by jump-diffusion. Suppose we have 3D models $\{O_i, i = 1, ..., n\}$ that describe existing objects in a 3D scene and each of these models is parametrized by 3D coordinates, pose, and others. Any possible scene x can be denoted as $x \in \chi \subset \cup_{i=1}^{n} \cup_{m=0}^{\infty} O_i^m$ where m is the number of occurrences of each type of objects and n is the overall number of objects that appear in the scene. The image data can be denoted as $y \in Y$ where Y is the observation space.

We can further denote interior orientation parameters (IOP) as $e \in E$. In a Bayesian framework, the *a posteriori* density is

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

where $p(y)$ is the probability density function of y and L is a likelihood function. To recogize 3D objects in 2D images, we choose the Maximize A Posteriori Probability (MAP) estimator which finds the best x that makes $p(x|y; e)$ the maximum. Because each observed image is 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 and $\Re^2_{e_2}$ is a 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 is indexing of 3D invariants. The direct indexing method (Funt and Finlayson, 1995) is a rather straightforward method that is also effective in computation. However, 3D invariants may not always exist. The Generalized Hough Transform (GHT) is another method that finds the most significant evidence by voting in χ space according to a given y. The Hough transform space is actually a rough approximation of $p(x \mid y;e)$ and works only in well defined situations. We propose a method that utilizes advantages of the indexing method and Hough transform for a fast approximation. It then estimates x more accurately by Markov Chain Monte Carlo.

Figure 2. Four common types of traffic lights.

Description of Models—Traffic Lights

In order to estimate the object x in the MAP estimator, we shall study x explictly so that we know the parameters to be estimated. Throughout this paper we restrict ourselves to one of the important civil infrastructure objects—traffic lights. To describe traffic lights in outdoor images such as Figure 6a, we use the following parameters:

- *Type t:* The types of traffic lights considered in this study are shown in Figure 2;
- *Illuminance:* This refers to the colors of the shell and the individual lights, denoted as $c_s\{R,G,B\}$, $c_r\{R,G,B\}$, $c_y\{R,G,B\}$, and c_g $\{R,G,B\}$, respectively;
- *Size:* The dimension of primitives under the assumption that each type of traffic lights is made of several primitives;
- *Spatial position* (x,y,z); and
- *Rotation angles* (ω,φ,κ) about the (X,Y,Z) coordinate axes.

Figure 3 shows the two coordinate systems we use in object description. The first one is the local coordinate system (X,Y,Z), also a view-centered coordinate system that can be implemented as a camera coordinate system. The second one is the aspect coordinate system (X',Y',Z') with its aspects defined as, for example, four traffic directions of a road intersection. To depict the traffic lights according to the aspect properties in our framework (Figure 1), the rotation angles $(\omega',\varphi',\kappa')$ of the traffic lights may have approximate values of $\omega' = 0$ and $\varphi' = 0$, because the traffic lights discussed in this study are always hung vertically. The rotation angle κ' is close to one of the four major aspects of $0, \frac{1}{2}\pi, \pi$, and $\frac{3}{2}\pi$. Suppose that the probability density distribution of κ' is the summation of four Gaussian distributions with four modes at the major aspects. The prior probability density function of κ' is

$$p(\kappa) = \frac{f(\kappa')}{Z}, \text{ with}$$

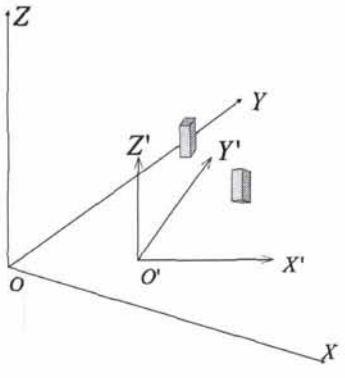


Figure 3. (X,Y,Z)—local coordinate system (view-centered) and (X',Y',Z')—aspect coordinate system.

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

where $Z = \int_0^{2\pi} f(\kappa')d\kappa'$.

Let $(\omega_1,\varphi_1,\kappa_1)$ be the rotation angles from the (X',Y',Z') to the (X,Y,Z) coordinate system. ω, φ, and κ of the object are functions of (ω_1,ω'), (φ_1,φ'), and (κ_1,κ'), respectively. The angles $(\omega_1,\varphi_1,\kappa_1)$ can be solved for by the vanishing points detection method, which will be discussed in the next section.

Traffic Light Recognition through Hypothesis Testing

A traffic light consists of several primitives. It has four major aspects that can be determined, for example, by a vanishing point detection method. We developed a new viewpoint dependnt bottom-up and top-down method based on hypothesis generation and test illustrated in Figure 4, which is a realization of the framework shown in Figure 1. The basic strategy can be characterized by the two processes, i.e., hypothesis generation and estimation of model parameters.

Hypothesis Generation

Finding Major Aspects

To compute (ω,φ,κ), the first step is to obtain $(\omega_1,\varphi_1,\kappa_1)$. 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, the *vanishing point*. Although an infinite number of parallel line sets exist in a real scene, in mobile mapping images the dominant directions are road directions along $(\omega_1,\varphi_1,\kappa_1)$. Based on this fact, we derive $(\omega_1,\varphi_1,\kappa_1)$ by extracting the vanishing points in a single image. We define a Gaussian sphere at the origin of the camera of the local coordinate system (X,Y,Z). Let $\vec{U}(\theta_u,\varphi_u)$ be the direction of a vanishing point in the Gaussian sphere and $\vec{N}_i(\theta_i,\varphi_i)$ be the normal of a plane that passes through the origin of the Gaussian sphere and the i-th straight parallel line segment (Figure 5) (Lutton *et al.*, 1994; Shufelt, 1996). Because

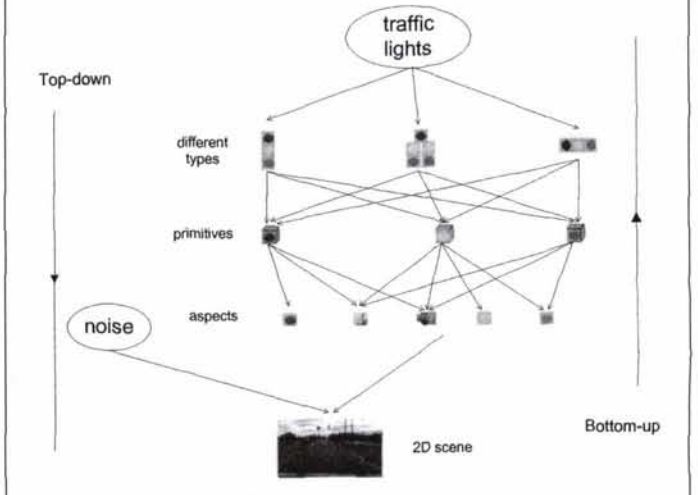


Figure 4. Traffic light recognition using bottom-up and top-down methods.

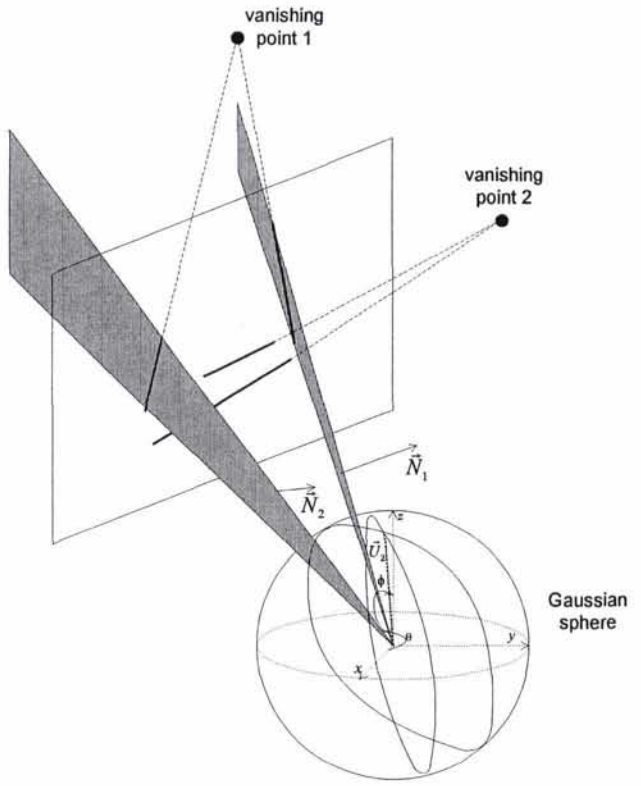


Figure 5. Vanishing point geometry and the corresponding Gaussian sphere.

$\vec{N}_i \cdot \vec{U} = 0$, we then have $\cos(\theta_i - \theta_u)\sin\varphi_i \sin\varphi_u + \cos\varphi_i \cos\varphi_u = 0$ because of their orthogonality. For each line segment in the 2D image, we can find the normal of the corresponding plane, $\vec{N}$. Using the above equations of all parallel line segments, we can determine the vanishing direction $\vec{U}$ using a voting process. A vanishing point direction is found at a local maximum in the voting space. The implemented method for detecting vanishing points has the following steps:

- Extract edge maps at a large scale σ using a color edge detector (Lee and Cok, 1991);
- Apply edge thinning and following methods to get line segments;
- Use the Lowe edge split method to split line segments into straight line segments;
- For each extracted straight line segment, $\vec{N}_i(\theta_i,\varphi_i)$ is computed and thus every $\vec{U}(\theta_u,\varphi_u)$ that meets the vanishing point geometry is voted into (θ,φ) space where θ is selected unevenly so that the patches with the θ-interval in different places of the Gaussian sphere cover the same area; and
- Apply a mean-shift clustering algorithm (Cheng, 1995) to find vanishing points in the voting space.

The above algorithm has been tested with a set of mobile mapping images (both color and black/white). The result shows its robustness under different circumstances. The directions $(\omega_1,\varphi_1,\kappa_1)$ of vanishing points are computed in single images.

Figure 6 shows an original image (color, printed in black and white) (a), its voting space for finding the vanishing point (b), an original black-and-white image (c), and its voting space for finding the vanishing point (d). Along with $(\omega',\varphi',\kappa')$, $(\omega_1,\varphi_1,\kappa_1)$ give us the rotation angles (ω,φ,κ) of the traffic lights in the local coordinate system. The cross signs indicate the locations of the vanishing points detected.

Finding Candidate Regions of Traffic Lights

We build 2D image templates by back-projections of 3D traffic light models with the computed aspects (ω,φ,κ). These 2D templates can be used to match traffic lights in the image. Swain and Ballard (1991) initiated a "color indexing" method that compares the histogram of an 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 filter and four directional first-order derivatives to convolute with the color image and compute the histogram again. Slater and Healey (1996) used local color pixel distributions instead of the entire image to recognize objects. The local color invariants are important to us because we only want to extract the candidate traffic light regions or "hot spots" that most likely contain traffic lights. We developed a new algorithm that captures both photometric and geometric invariants to build hot-spot maps using histogram filtering. The algorithm is as follows:

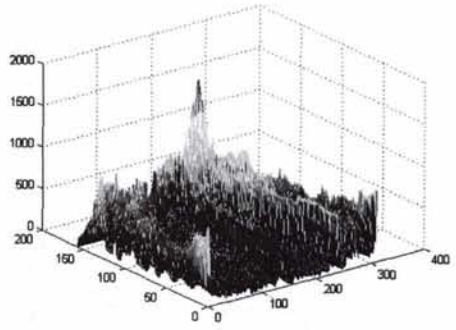
Step 1. The original color image in the (R,G,B) space is transformed into the L^*,u^*,v^* space (Wyszecki and Stiles, 1982) to achieve the equal distance property so that the distance between two colors can be computed as L-2 norm in the L^*,u^*,v^* space.

Step 2. 2D image templates of the traffic light model, I_t, $t = 1,2,3,4$ at four major aspects are generated as 2D projections of the traffic light primitives.

Step 3. Color template images I_t, $t = 1,2,3,4$ are transformed from the (R, G, B) to the L^*,u^*,v^* space and the corresponding histograms are computed as $H_t^{(L^*)}(l) = \frac{1}{Z}\sum_{s\in I_t}\delta(l - L^*(s))$ where l is each bin value in the domain of L^*, δ is the Dirac delta function, s is the pixel site, and Z is the normalization term such that $\sum_l H_t^{(L^*)}(l) = 1$. Similarly, we have $H_t^{(u^*)}(u) = \frac{1}{Z}\sum_{s\in I_t}\delta(u - u^*(s))$ where u is each bin value in the domain of u^*, and $H_t^{(v^*)}(v) = \frac{1}{Z}\sum_{s\in I_t}\delta(v - v^*(s))$ where v is each bin value in the domain of v^*. An edge map is obtained using the color edge detection method described in Lee and Cok (1991) at $\sigma = 1.0$. A large scale factor does not give a satisfactory result because the aspect image is relatively small. The image showing edge pixels can be denoted as $I_t^E(s) = \begin{cases}1 & \text{if s is an edge pixel}\\ 0 & \text{otherwise}\end{cases}$. The histogram of gradients of the edge points is computed as $H_t^{(E)}(g) = \frac{1}{Z}\sum_{s\in I_t^E \text{ and } I_t^E(s)=1}\delta(g - g(s))$ where j is each bin value in the domain of discrete gradient values, $g(s)$ is the gradient at s, and Z is the normalization term. Figure 7 displays the histograms in L^*, u^*, v^*, and edge gradients of a template.

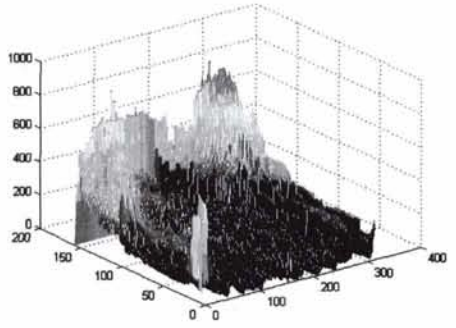
Step 4. Three square windows with different sizes corresponding to three scales are defined as $W_1(s)$, $W_2(s)$, and $W_3(s)$, which are centered at a pixel s in the original image. The histogram of each window is computed by $H_{W_i(s)}^{(L^*)}(l) = \frac{1}{Z}\sum_{s'\in W_i(s)}\delta(l - L^*(s'))$, $H_{W_i(s)}^{(u^*)}(u) = \frac{1}{Z}\sum_{s'\in W_i(s)}\delta(u - u^*(s'))$, and $H_{W_i(s)}^{(v^*)}(v) = \frac{1}{Z}\sum_{s'\in W_i(s)}\delta(v - v^*(s'))$, respectively. Similarly, we have the histogram $H_{W_i(s)}^{(E)}(g) = \frac{1}{Z}\sum_{s'\in I_{W_i(s)}^E}\delta(g - g(s'))$ for the edge map of the windows. The overall measurements of the distances between $H_t^{(L^*)}$ and $H_{W_i(s)}^{(L^*)}$, $H_t^{(u^*)}$ and $H_{W_i(s)}^{(u^*)}$, and $H_t^{(v^*)}$ and $H_{W_i(s)}^{(v^*)}$ tell us how the template I_t matches the image piece in the window $W_i(s)$ at the pixel site. We compute the overall photometric similarity as $\hbar_{t,W_i}(s) = \sqrt{D_{t,W_i}^{L^*}(s)^2 + D_{t,W_i}^{u^*}(s)^2 + D_{t,W_i}^{v^*}(s)^2}$ where the

a



b

c



d

Figure 6. Result of the vanishing points detection algorithm. (a) Black-and-white version of the original color image (the white cross in the upper central area is a computed vanishing point). (b) Voting space of (θ,φ) generated from extracted straight lines in (a). (c) The original black-and-white image (the black cross shows a vanishing point found). (d) Voting space of (θ,φ) generated from extracted straight lines in (c).

distance between two histograms is $D_{t,W_i}(s) = 1 - \|H_t \mid H_{W_i(s)}\| = 1 - \sum_j \min[H_t(j), H_{W_i(s)}(j)]$. Other methods, such as the Kullback-Leibler divergence, may also be used to compute the distance between two probability distributions.

Step 5. The geometric similarity is computed by measuring the edge map differences between the template and image pieces $\chi_{t,W_i}(s) = D_{t,W_i}E(s)$ where the distance between two histograms is obtained the same way as defined above.

Step 6. We use the similarity maps (Figure 8) derived from $\hbar$ and χ values to determine the possible hot spots. A possible approach is to use a thresholding method where every value greater than a hard threshold is set to 1, and otherwise to 0. However, it is difficult to select the appropriate threshold. We solve the problem in a signal detection approach in which the noise or signal is determined using its probability distributions. With this method, the system can be trained. To do this, we extracted some traffic light samples (image pieces). We tell the system the pixels of the traffic lights in the image piece that are used as signals, and all other pixels in the image piece are treated as noise. We then have their histograms as shown in Figure 9. With this training method, it is straightforward to generate an object signal map where the dark pixels mean object signal and the bright pixels mean noise. By combining these maps at different window sizes and different aspects, we finally obtain the final hot-spot map where candidate traffic light regions are marked.

Hypothesis Test—Parameter Estimation for Traffic Lights by MCMC

Figure 9b shows the final hot-spot map of the image of Figure 9a, from which the candidate regions may be extracted. We assume that the traffic lights in the image do not have occlusions. We now define a candidate region as a rectangular region in the image, which encloses a connected hot spot region. The size of the region should be slightly larger than its hot spot region since initially we do not know the exact position and size of the traffic lights enclosed. The remaining work is to recognize traffic lights and to estimate the parameters as a hypothesis test.

Given a candidate region y and IOP parameters e, we want to find the x (parameters defining a traffic light model) that maximizes the *a posteriori* probability. Suppose that x is composed of $[t, c_s(R,G,B), c_r(R,G,B), c_y(R,G,B), c_g(R,G,B), (x_o,y_o,z_o), (w,h), (\omega,\varphi,\kappa)]$. (w,h) represents width and height of the traffic light and the rest of the terms are defined in the section on Description of Models-Traffic Lights. 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 $\omega = 0$ and $\varphi = 0$, which are approximately true in the real scene. These requirements could be met in our cases reasonably and the parameters may be simplified to $[t, c_s(R,G,B), c_r(R,G,B), c_y(R,G,B), c_g(R,G,B), (x_I,y_I), (w,h), \kappa]$, where (x_I,y_I) are the 2D coordinates of the center of the traffic light in a candidate region. Let $F_{(x,e)}$ be the orthogonal projection of a traffic light

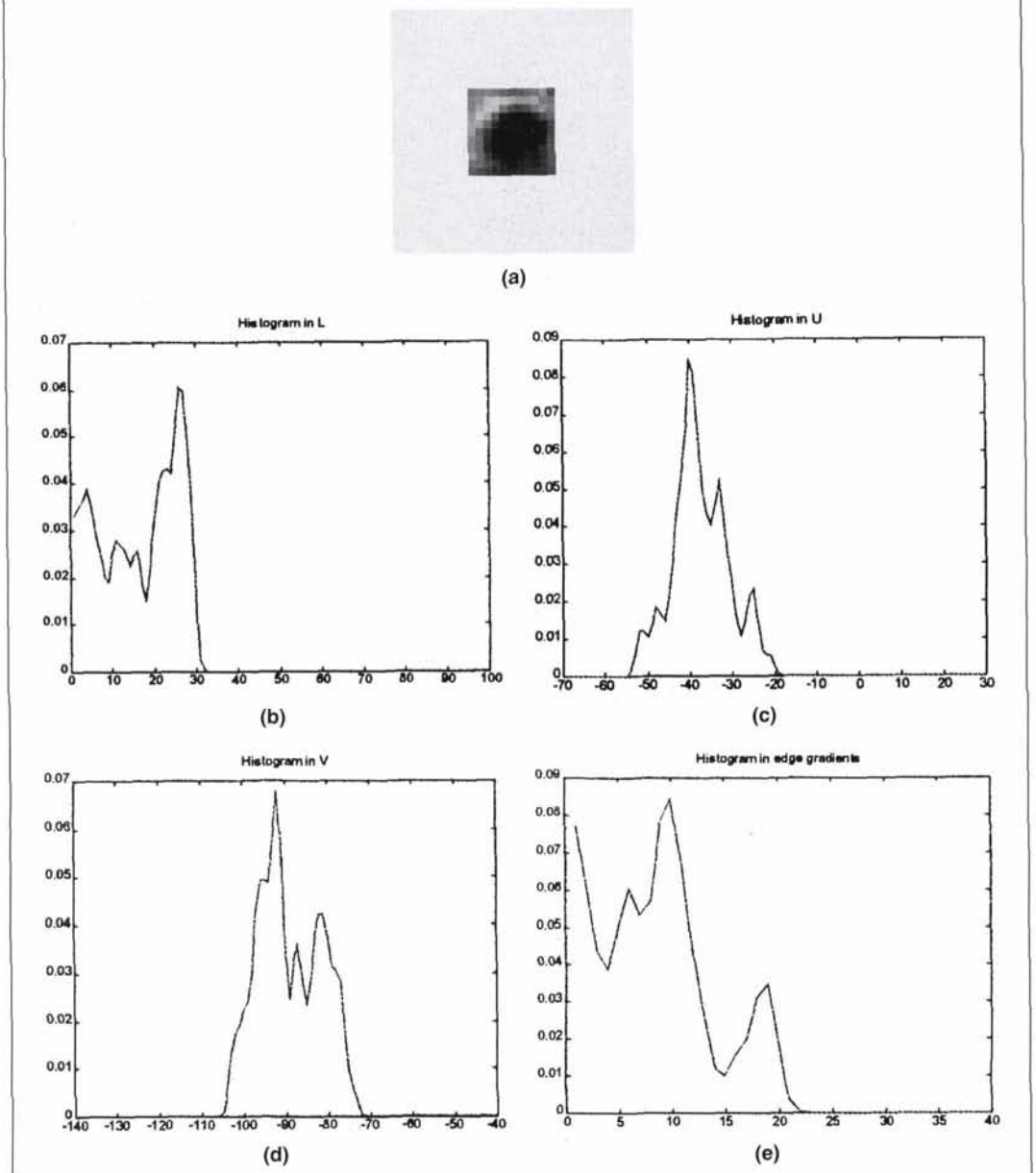


Figure 7. An aspect template and its corresponding histograms in L^*, u^*, v^* and edge gradients. (a) Template image at aspect ($\kappa' = 0$. (b) Histogram of the template in L^*. (c) Histogram of the template in u^*. (d) Histogram of the template in v^*. (e) Histogram of the template in edge gradients.

model parameterized 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. Recall that, to implement Equation 1, we must know the likelihood function that can be expressed as

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

The log likelihood function becomes

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

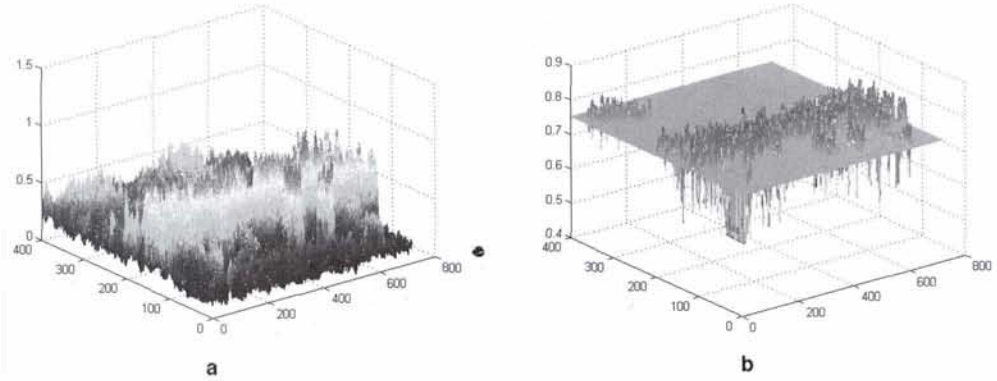


Figure 8. Histogram similarity maps. (a) Histogram similarity map $\hbar$ with a template applied on a color image. (b) The corresponding histogram similarity map $\mathfrak{X}$.

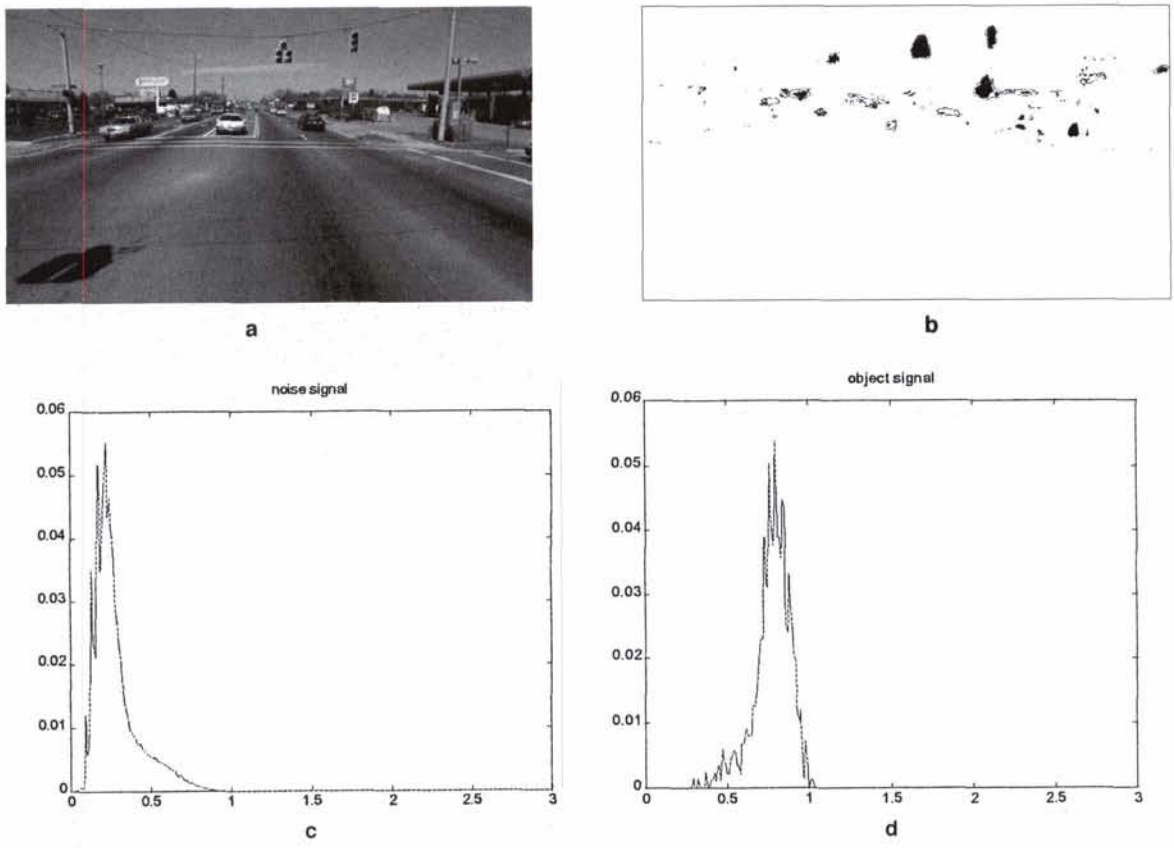


Figure 9. Histograms of signals and noise of a color image (shown in black and white) and its hot-spot map. (a) Original image. (b) Hot-spot map of (a). (c) Histogram of similarity value of noise (background). (d) Histogram of similarity value of object signal (traffic light samples).

where g is a constant value of $m \sum_{c=L^*u^* \text{ and } v^*} \log\left(\frac{1}{\sqrt{2\pi\sigma^{(c)}}}\right)$ with m being the number of pixels in y.

It is important to know the background image distribution to compute the likelihood function. In Equation 4 we treat the background as Gaussian noise because, in our case, the traffic lights generally appear with the sky as the background. Figure 10 shows the histograms of the background image in L^*,u^*,v^* with imposed Gaussian distributed noise functions. The two curves fit well, and we can thus use the Gaussian distributed noise as the background. Zhu and Mumford (1997) proposed a more general statistical description of background images wherein clutter image is learned. This method could be used if a more complicated background appears in our images.

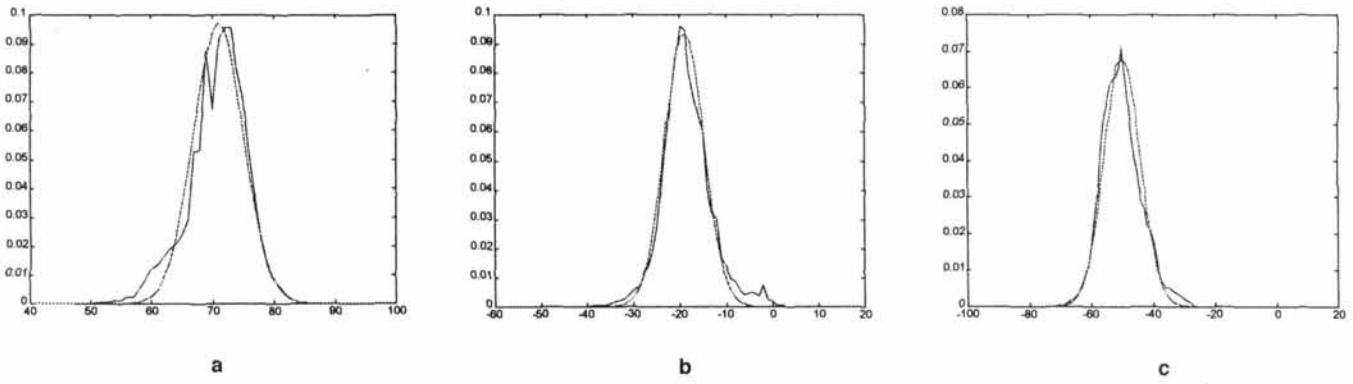


Figure 10. Histograms of the background image (solid lines) and the Gaussian-distributed noise distribution (dashed lines). (a) Histogram of the background in L^*. (b) Histogram of the background in u^*. (c) Histogram of the background in v^*.

The *a posteriori* probability then becomes

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

where B is the so-called "temperature" used in the annealing technique. The introduction of B does not change the x^* that maximizes the *a posteriori* probability because the exponential function is monotone. Note also that $p(x)$ is the Gibbs distribution introduced in Geman and Geman (1984). We rewrite the above equation as

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

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

- A set of new parameters x_2 is proposed by sampling from a probability distribution $G(x_1,)$ based on initial parameters of x_1;
- The energy at x_2 is computed and is compared with that at x_1;
- If $H(x_2) \leq H(x_1)$, x_2 is accepted;
- If $H(x_2) > H(x_1)$, x_2 is accepted with the probability $\exp((H(x_1) - H(x_2))/B)$;
- If x_2 is not accepted, then x_1 will be kept; and
- The transformation matrix $\pi((x_1,x_2)$ becomes

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

where $(H(x_2) - H(x_1))^+ =$

$$\begin{cases} 0 & H(x_2) - H(x_1) \geq 0 \\ -(H(x_2) - H(x_1)) & H(x_2) - H(x_1) < 0 \end{cases}$$

It can be proven that $p(x_1)\pi(x_1,x_2) = p(x_2)\pi(x_2,x_1)$ and the requirement for the convergence of a Markov Chain is met. 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 is

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

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

The detailed balance equation $p(x_1)\pi(x_1,x_2) = p(x_2)\pi(x_2,x_1)$ holds as well. It is now important to generate a proposal matrix. To take advantage of both the speed of the Generalized Hough Transform (GHT), and the ability of MCMC (Markov Chain Monte Carlo) for searching for the global optimal solution, we use the result of a GHT as the proposal matrix $G(x_1,x_2)$. The voting space of GHT actually gives a distribution of every possible parameter.

Results

In Figure 11, a color stereo image sequence (shown in black and white) captured by a Mobile Mapping System is displayed. The parameters of the traffic lights were estimated by MCMC when the minimum energy was reached. The annealing technique was applied so that the Markov chain would not be

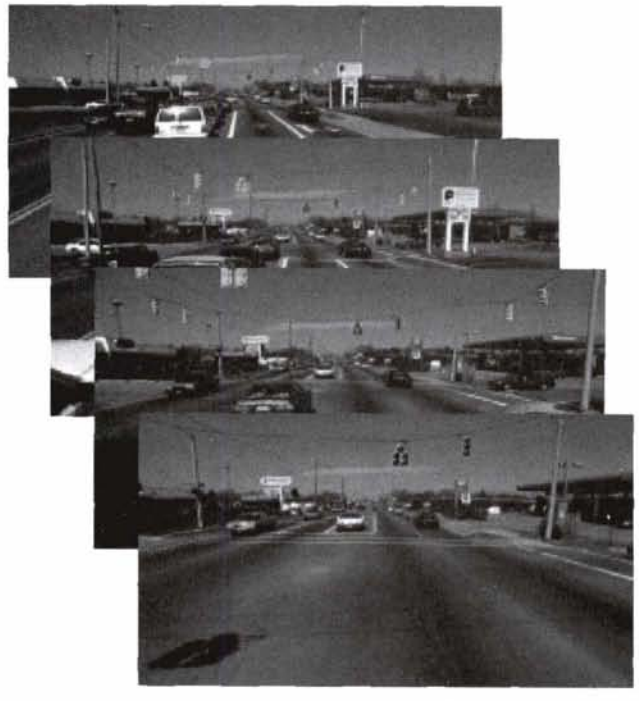


Figure 11. A color stereo image sequence (shown in black and white) taken by a Mobile Mapping System.

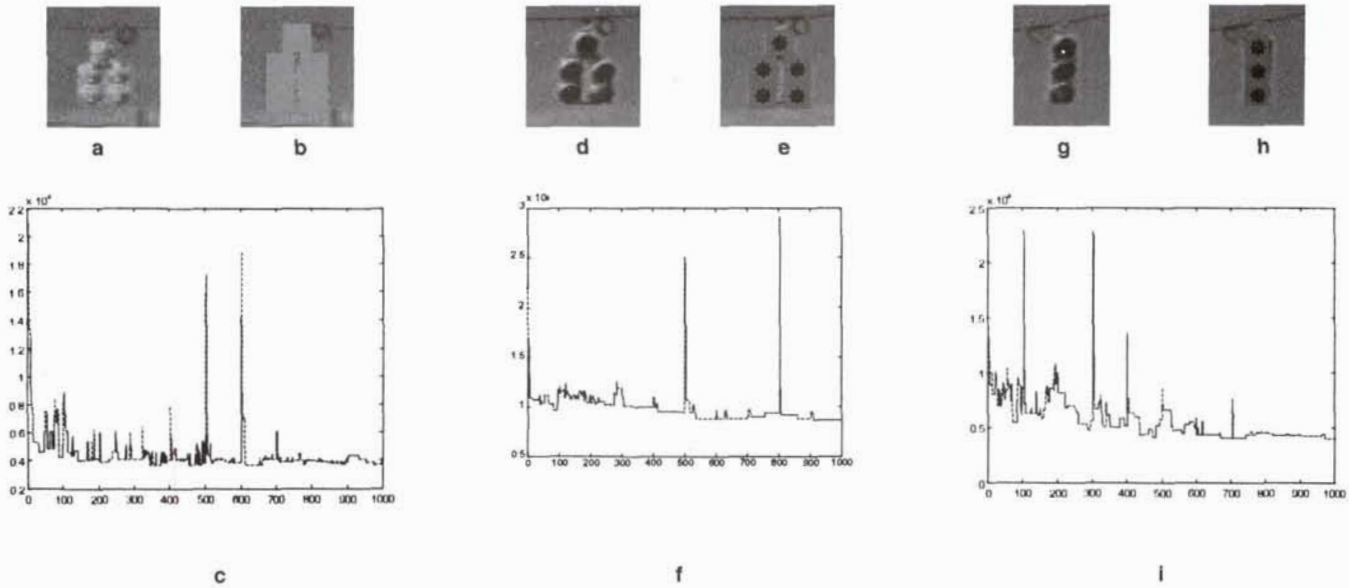


Figure 12. Original image pieces with the recognized traffic lights and the energy curve in MCMC. (a) Back of a traffic light in a candidate region. (b) Recognized traffic light model corresponding to (a). (c) Energies in MCMC for (a). (d) Front of a traffic light in a candidate region. (e) Recognized traffic light model corresponding to (d). (f) Energies in MCMC for (d). (g) Front of a single column traffic light. (h) Recognized traffic light model corresponding to (g). (i) Energies in MCMC for (g).

trapped at a local minimal. We can see in Figure 12 that the traffic lights in the input images (Figures 12a, 12d, and 12g) are correctly recognized. The system may find difficulties if the traffic lights in the input images appear too small or the color correspondance between the images and the model is too weak. The energy curves in Figures 12c, 12f, and 12i show that the Markov chains converge after some hundreds of iterations. However, due to the small size of traffic lights appearing in the input images, the energy curves did not go down dramatically during the Markov chain process.

Conclusions

We proposed a *hypothesis-and-test* approach that employs a viewpoint dependent framework and Markov Chain Monte Carlo to recognize traffic lights in real image sequences taken by a Mobile Mapping System. In *hypothesis generation*, the vanishing point detection finds the major aspects. A histogram filtering method is then used to generate a hot-spot map. In the *hypothesis test*, the parameters of the 3D traffic lights are estimated by Markov Chain Monte Carlo. Note that the *hypothesis generation* method overcomes the shortcomings of the traditional MCMC method that is computationally inefficient for high-dimensional feature estimation. Also, the use of MCMC in the hypothesis test guarantees a global optimal solution.

There are a few issues we would like to address in future research. First, the vanishing point detection method for estimating the major aspects may be sensitive in a very cluttered scene. Second, a more accurate background statistical model should be developed to replace the Gaussian model (Lee *et al.*, 1999). Finally, other hypothesis generation methods may be explored because a color histogram is not always a robust cue.

Acknowledgment

We acknowledge support from the National Science Foundation (CMS-9812783). We thank The Ohio State University Center for Mapping (CFM) for partial support of this research and TransMap Inc. for providing mobile mapping data. We also thank Dr. Song Chun Zhu for stimulating discussions and thoughtful suggestions.

References

Bulthoff, H.H., Y.E. Shimon, and J.T. Michael, 1994. *How Are Three-Dimensional Objects Represented in the Brain?* A.I. Memo No. 1479, MIT. Massachusetts Institute of Technology, Cambridge, Massachusetts, 20 p.

Cheng, Y., 1995. Mean Shift, Mode Seeking, and Clustering, *IEEE Trans. PAMI*, 17(8):790–799.

Dickinson, S.J., A.P. Pentland, and A. Rosenfeld, 1992. From Volumes to Views: An Approach to 3-D Object Recognition, *Journal of Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(2):130–154.

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

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

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

Li, R., 1997. Mobile Mapping—An Emerging Technology for Spatial Data Acquisition, *Photogrammetric Engineering & Remote Sensing*, 63(9):1085–1092.

Lutton, E., H. Maitre, and J. Lopez-Krahe, 1994. Contribution to the Determination of Vanishing Points Using Hough Transform, *IEEE Trans. PAMI*, 16(4):430–38.

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):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):2678–2689.

Shufelt, J.A., 1996. *Projective Geometry and Photometry for Object Detection and Delineation*, PhD dissertation, CMU-CS-96-194, Carnegie Mellon University, Pittsburg, Pennsylvania, 276 p.

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

Swain, M., and D. Ballard, 1991. Color Indexing, *International Journal of Computer Vision*, 7(1):11–32.

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

Wyszecki, G., and W.S. Stiles, 1982, *Color Science: Concepts and Methods, Quantitative Data and Formulas, Second Edition*, Wiley, New York, N.Y., 950 p.

(Received 01 September 2000; accepted 15 February 2001; revised 03 April 2001)