

A Hierarchical Bayesian Generation Framework for Vacant Parking Space Detection

Ching-Chun Huang and Sheng-Jyh Wang

Abstract—In this paper, from the viewpoint of scene understanding, a three-layer Bayesian hierarchical framework (BHF) is proposed for robust vacant parking space detection. In practice, the challenges of vacant parking space inference come from dramatic luminance variations, shadow effect, perspective distortion, and the inter-occlusion among vehicles. By using a hidden labeling layer between an observation layer and a scene layer, the BHF provides a systematic generative structure to model these variations. In the proposed BHF, the problem of luminance variations is treated as a color classification problem and is tackled via a classification process from the observation layer to the labeling layer, while the occlusion pattern, perspective distortion, and shadow effect are well modeled by the relationships between the scene layer and the labeling layer. With the BHF scheme, the detection of vacant parking spaces and the labeling of scene status are regarded as a unified Bayesian optimization problem subject to a shadow generation model, an occlusion generation model, and an object classification model. The system accuracy was evaluated by using outdoor parking lot videos captured from morning to evening. Experimental results showed that the proposed framework can systematically determine the vacant space number, efficiently label ground and car regions, precisely locate the shadowed regions, and effectively tackle the problem of luminance variations.

Index Terms—Bayesian inference, image labeling, parking space detection, semantic detection.

I. INTRODUCTION

USING AN INTELLIGENT surveillance system to manage parking lots is becoming practical nowadays. A recent technology review about smart parking system can be found in [1]. To assist users to efficiently find a vacant parking space, an intelligent parking space management system can not only provide the total number of vacant spaces in the parking lot but also explicitly identify the location of vacant parking spaces. Among those smart parking systems, vision-based systems have gathered great attention in recent years. Unlike using trip sensors or other types of sensors to monitor a parking lot, a vision-based system may provide many

Manuscript received January 12, 2010; revised April 18, 2010 and June 15, 2010; accepted July 18, 2010. Date of publication October 14, 2010; date of current version January 22, 2011. This work was supported in part by the Ministry of Economic Affairs, under Grant 98-EC-17-A-02-S1-032, and in part by the National Science Council of Taiwan, under Grant 97-2221-E-009-132. This paper was recommended by Associate Editor C. N. Taylor.

The authors are with the Department of Electronics Engineering, Institute of Electronics, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: chingchun.huang3@gmail.com; shengjyh@faculty.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2010.2087510

value-added services, like parking space guidance and video surveillance.

In practice, the major challenges of vision-based parking space detection come from occlusion effect, shadow effect, perspective distortion, and the fluctuation of lighting conditions. In Fig. 1, we show several parking lot images in our dataset. In these images, some environmental factors are mixed together in a sophisticated way. For instance, the illumination in a sunny day is quite different from that in a cloudy day, a parked car may occlude or cast a shadow over the parking space next to it, a shadowed region may be mistakenly recognized as a dark-colored vehicle, and a light-colored vehicle under strong sunlight may look very similar to a vacant parking space.

Up to now, many methods have been proposed to overcome the aforementioned difficulties. These methods can be roughly classified into two categories: car-driven and space-driven. For a car-driven method, cars are the major target and algorithms are developed to detect cars. Based on the result of car detection, vacant parking spaces are determined. To detect objects of interest, plentiful object detection algorithms can be used. For example, the object detection method proposed in [2] by Schneiderman and Kanade is a trainable detector based on the statistics of localized parts. The adaboosting-based detection algorithm [3] is another widely used technique for the detection of specific objects in 2-D images. The method proposed by Felzenszwalb *et al.* [4] offered an efficient way to match objects based on a part-based model that well represents an object by pictorial structures. A global color-based model had been proposed by Tsai *et al.* [5] to efficiently detect vehicle candidates. In that approach, a Bayesian classifier based on corner features, edge features, and wavelet coefficients was trained to verify the detection of vehicles. On the contrary, Lee *et al.* [6] and Masaki [7] kept tracking and recording the movement of vehicles to identify empty parking spaces. Even though these object detection-based frameworks had gained impressive achievement in many circumstances, such as highway and roadway, most of these algorithms are not specifically designed for vacant parking space detection in a typical parking lot. For example, as shown in Fig. 1, the captured images may include some cars with unclear details. Besides, due to the perspective distortion, a car far away from the camera only occupies a small area in the captured image. This fact may also affect the performance of car detection.

For a space-driven method, the property of a vacant parking space is the major focus and available parking spaces are

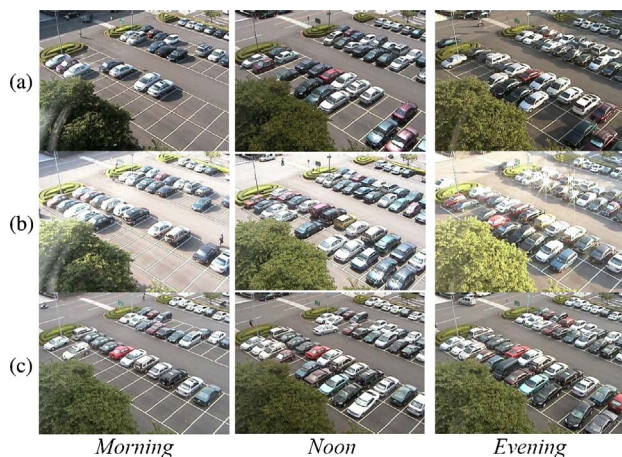


Fig. 1. Image shots of a parking lot. (a) Captured in a normal day. (b) Captured in a day with strong sunlight. (c) Captured in a cloudy day.

detected directly. When the camera is static, several background subtraction algorithms, such as [8]–[10], can be used to detect foreground objects. Typically, these algorithms assume that the variation of the background is statistically stationary within a short period. Unfortunately, this assumption is not always true for an outdoor scene. For example, a passing cloud that block the sunlight may suddenly change the lightness. To handle the dynamic variation of an outdoor environment, a possible solution is to build a complete background reference set under all kinds of lighting conditions. However, lots of memory and heavy computational cost will be needed to support this approach. To solve this problem, Funck *et al.* [11] proposed an eigen-space representation that models a huge set of background models with much less memory space and computational cost.

With a suitable background model, a typical way to determine the status of a parking space is to check the ratio of foreground pixel number to background pixel number. If the ratio is larger than a predefined threshold, that parking space is considered as occupied. However, even if the background model is well learned, this kind of method still suffers from the occlusions and shadows caused by neighboring cars. To improve the performance of detection, Huang *et al.* [12] proposed a Bayesian detection framework to take into account both ground plane model and car model. Both occlusion effect and illumination variation were modeled under that framework. Recently, Bong *et al.* [13]–[14] proposed a car park occupancy information system by using a “bi-stream” detector to overcome the shadow effect. In their approach, one stream used the background subtraction method to perform car detection, while the other stream adopted edge information to achieve shadow-insensitive detection. By using an “And” operator to combine both detection results, detection performance was improved.

On the contrary, some other space-driven methods assume that a vacant parking space possesses homogeneous appearance and use this property to detect vacant spaces. For example, Yamada and Mizuno [15] designed a homogeneity measure by calculating the area of fragmental segments. In principle, a vacant space has fewer but larger segments, while

the area of a parked car has an opposite property. Lee *et al.* [16] suggested an entropy-based metric to determine the status of each parking space. Once the entropy inside a space region is larger than a threshold, that space is considered as occupied. However, these two systems ignored the shadow and occlusion caused by adjacent cars. In [17], Fabian used a segment-based homogeneity measure similar to that in [15] and proposed a method for occlusion handling. By pre-training a weighting map to indicate the image regions that may get occupied by neighboring cars, the influence of the occlusion effect can be reduced. However, due to the perspective distortion, a distant parking space only occupies a small region in the captured image. This leads to instable measurement of homogeneous areas. In order to overcome the perspective problem, López-Sastre *et al.* [18] suggested the rectification of the perspective effect by transforming the original parking lot image into a top-view image. A Gabor filter bank was used to derive the homogeneity feature for vacant space detection. Even though their homogeneity measure is effective for most parking spaces, the environmental variations, especially the shadow effect and the over-exposure effect caused by strong sunlight, may fail the assumption of homogeneous appearance. In practice, the shadow effect makes a parking space less homogeneous while the over-exposure effect makes the appearance of a car more homogeneous.

Some other authors tried to detect vacant parking spaces via classification. For example, Dan [19] trained a general support vector machine (SVM) classifier by directly using the cascaded color vectors inside a parking space as the classification feature. However, the occlusion patterns were not well modeled in their approach. On the contrary, Wu *et al.* [20] grouped three neighboring spaces as a unit and defined the color histogram across three spaces as the feature in their SVM classifier. With this arrangement, the inter-space correlation can be learned beforehand to overcome the inter-occlusion problem. However, the performance of classification is greatly affected by the environmental variations. In general, the lighting changes may cause the variations of object appearance in both brightness and chromaticity. This effect may dramatically degrade the accuracy of classification-based detection.

Besides the aforementioned smart parking lot management (SPLM) systems, which detect vacant parking spaces based on static surveillance cameras installed around the parking lot, the automobile parking (AP) system is another approach that detects vacant parking spaces based on vehicle-embedded cameras. These in-car AP systems help drivers to identify available parking spaces while they are driving. For example, Suhr *et al.* [21] proposed an optical flow-based method to estimate the 3-D scene of the rear-view of a moving car. The reconstructed Euclidean 3-D scene is further analyzed to detect vacant parking spaces. In [22], the side-view images are captured for analysis as a car moves along a row of parking spaces. By classifying each captured image frame as either “vehicle” frame or “background” frame, they can identify vacant parking spaces. Even though the AP approach provides an interesting way for detecting vacant parking spaces, the focus of our approach is to improve the performance of an SPLM system, which can easily fit into today’s parking lot

management systems. In this paper, we propose a hierarchical Bayesian generation framework to model the generation of environmental variations, including the perspective distortion, the occlusion effect, the shadow effect, and the fluctuation of lighting condition. As will be shown in Section VI, accurate results can be obtained based on the proposed framework.

The rest of this paper is organized as follows. In Section II, we present the main idea of our algorithm. The top-down information from the 3-D scene model is detailed in Section III, while the message from image observation is presented in Section IV. The whole inference procedure is explained in Section V. Experimental results and discussions are presented in Section VI. Last, Section VII concludes this paper.

II. ALGORITHM OVERVIEW

In our system, the scene understanding and vacant parking space detection are accomplished based on the integration of scene prior and image observation. By treating the status of each parking space as a part of the scene parameters, the vacant space detection is achieved via the process of scene inference. The general concept of the proposed system is illustrated in Fig. 2. Based on a three-layer Bayesian hierarchical framework, called BHF, the bottom-up messages from image observation and the top-down knowledge from the scene model are effectively integrated. In BHF, the illumination variations are overcome by transferring the fluctuating red, green, and blue (RGB) observations into meaningful labels. The labeling process is treated as a color classification process between content labeling and image observation. Since the observation difference is mainly caused by the object type and the lighting condition, we decompose the image observation into an object component and a lighting component. The object type is either “car” or “ground,” while the lighting condition is either “shadowed” or “unshadowed.” To adapt to the time-varying lighting condition, we online build the color classification models for object type and lighting condition. On the contrary, some global knowledge of the 3-D scene offers useful information for the labeling of image pixels. The top-down knowledge is propagated downward to influence the labeling process via the generation of an “expected object map” and an “expected shadow map.” Here, we explicitly define a generative model that takes into account the inter-occlusion effect, the expected shadow effect, and the perspective distortion. The relationships among these effects and the status of parking spaces are explicitly modeled via a Bayesian probabilistic model. By compromising between the expected labeling maps and the labeling from image observation, the status hypotheses of each parking space are evaluated. Finally, to avoid incorrect inference caused by unexpected occlusions, the global status hypotheses from the scene model provides useful constraints to handle partially inconsistent labels. Under the proposed BHF, the vacant parking space detection problem and the optimal image content labeling problem are integrated in a unified manner.

In principle, we can formulate the vacant space detection problem as a status decision process based on image observations from a single camera. Since the status of a parking

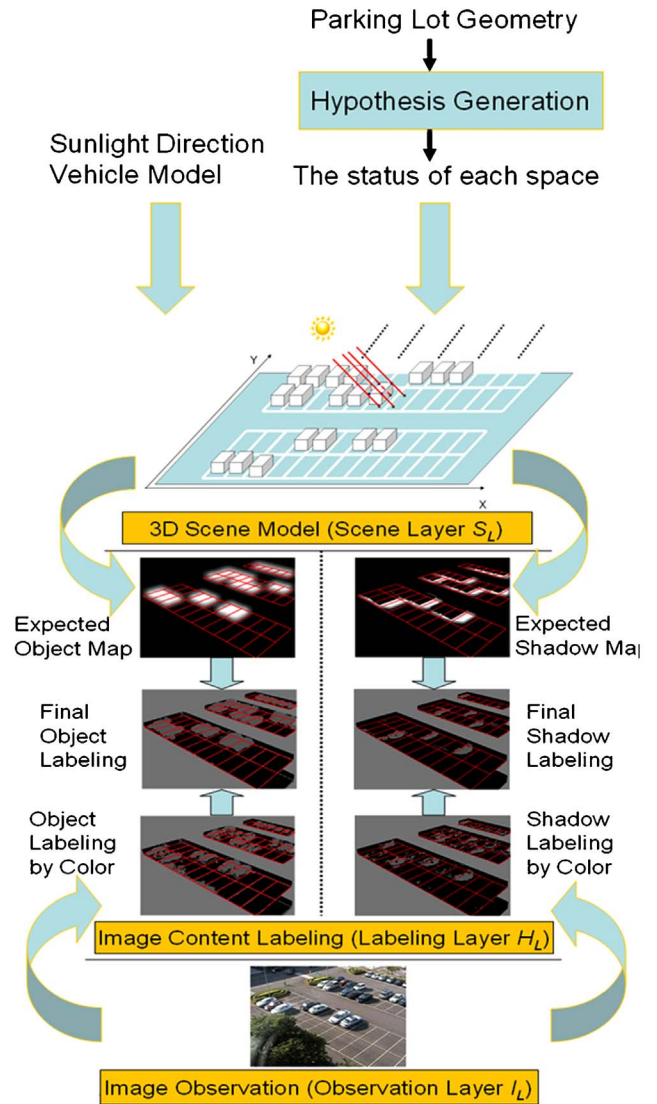


Fig. 2. Concept of the proposed vacant space detection process.

space may actually affect the inference of neighboring spaces, it is unsuitable to decide the status of each parking space individually. Instead, we analyze the status of neighboring parking spaces at the same time. Moreover, both bottom-up message and top-down knowledge are modeled as probabilistic constraints in the proposed BHF. The vacant parking detection process is regarded as a Bayesian inference problem and is solved by finding the most reasonable parking space status that fits both scene prior and image observation.

In Fig. 3, we show a simplified three-layer structure to explain the BHF framework. For vacant space detection, we define the image observation layer as I_L , where each node $I_L(m, n)$ indicates the RGB color feature at the (m, n) pixel of an image of size $M \times N$. On the contrary, we define the labeling layer as H_L , where each node $H_L(m, n)$ represents the categorization of the image pixel at (m, n) . The labeling result of $H_L(m, n)$ could be (C, S) , (G, S) , (C, US) , or (G, US) , where C denotes “car,” G denotes “ground,” S denotes “shadowed,” and US denotes “unshadowed.” Moreover, we define the scene layer as S_L , which indicates the status hypotheses of the

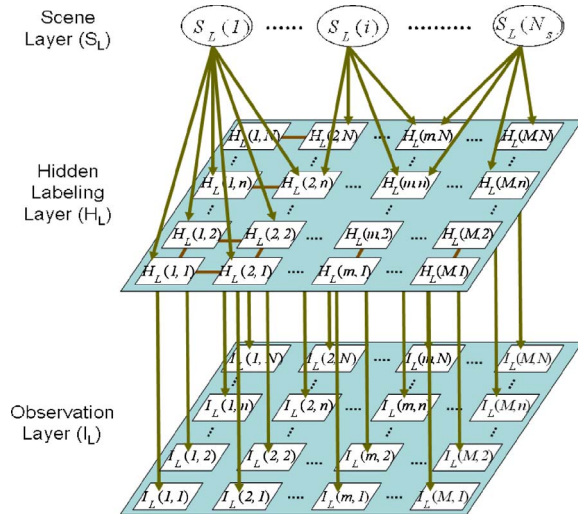


Fig. 3. Illustration of the three-layer BHF.

parking spaces. The node $S_L(i)$ in S_L denotes the status of the i th parking space. Its value can be either 1 (occupied) or 0 (vacant). Note that Fig. 3 is for illustration purpose only. The exact model of BHF is to be explained later in Section IV.

In this model, the topology of the inter-layer connections represents the probabilistic constraints between nodes. Given the observation I_L , the status of the parking spaces is determined by finding the pair (H_L^*, S_L^*) such that

$$H_L^*, S_L^* = \arg \max_{H_L, S_L} p(H_L, S_L | I_L). \quad (1)$$

Furthermore, (1) can be reformulated as follows:

$$\begin{aligned} H_L^*, S_L^* &= \arg \max_{H_L, S_L} \ln p(H_L, S_L | I_L) \\ &= \arg \max_{H_L, S_L} \ln [p(I_L | H_L, S_L) p(H_L | S_L) p(S_L)] \\ &= \arg \max_{H_L, S_L} \ln [p(I_L | H_L) p(H_L | S_L) p(S_L)] \\ &= \arg \max_{H_L, S_L} [\ln p(I_L | H_L) + \ln p(H_L | S_L) + \ln p(S_L)]. \end{aligned} \quad (2)$$

In (2), we assume $p(I_L | H_L, S_L) = p(I_L | H_L)$. That is, we assume the probabilistic property of the observed image data is conditionally independent of the scene model once if the pixel labels are determined. Moreover, $p(I_L | H_L)$ stands for the constraints between the labeling layer and the observation layer. In our approach, the labeling results should be consistent with the RGB values of the observed image, and the labels of adjacent pixels should follow some kind of smoothness constraint. On the contrary, $p(H_L | S_L)$ stands for the constraints between the scene layer and the labeling layer. In our approach, the labeling of parked cars and shadowed regions should match the expected inter-occlusion pattern and the expected shadow pattern in a probabilistic sense. Finally, $p(S_L)$ represents the prior knowledge of the parking space status. In our system, we assume that the ‘‘occupied’’ status and the ‘‘available’’ status are equally possible for every parking space. With this assumption, the $\ln p(S_L)$ term in (2) can be ignored. Moreover, to find the optimal solution in (2), we adopt the graph-cuts technique [23]–[25].

III. TOP-DOWN KNOWLEDGE FROM SCENE LAYER

Since the parking spaces in a parking lot are well structured, we can synthesize an expected object map once if we have the 3-D car model and a hypothesis about the status of parking spaces. On the contrary, if we know the lighting condition (sunny or cloudy) and have the direction of sunlight, we may also synthesize an expected shadow pattern. In our system, both expected object map and expected shadow map are created to help the labeling of image pixels.

In our approach, $p(H_L | S_L)$ is reformulated as follows:

$$p(H_L | S_L) = \prod_m \prod_n p(H_L(m, n) | S_L) \quad (3)$$

in which we assume the labeling nodes $H_L(m, n)$ are conditionally independent of each other once if the knowledge from the scene layer S_L is given. Since the object type and the lighting type are physically independent, we formulate $p[H_L(m, n) | S_L]$ as follows:

$$p(H_L(m, n) | S_L) = p(h^O(m, n) | S_L) p(h^L(m, n) | S_L). \quad (4)$$

In physics, the object labeling model $p[h^O(m, n) | S_L]$ includes the expected car mask and the inter-occlusion effect among neighboring cars, while the light labeling model $p[h^L(m, n) | S_L]$ includes the expected shadow mask to indicate shadowed pixels. To define these two labeling models, we first introduce a parametric model to define the 3-D structure of a parking lot. Based on the parametric scene model, we propose a generation process to generate the expected object labeling map and the expected shadow labeling map.

A. 3-D Scene Model

In our system, the number of parking space (N_s) and their locations on the 3-D ground plane are defined and learned in advance. In a normal situation, a car is parked inside a parking space. To simulate a parked car, we assume each car is a cube in the 3-D world. The length (l), width (w), and height (h) of the cube are modeled as three independent Gaussian random variables, with the probability density functions $p(l)$, $p(w)$, and $p(h)$. Besides, the random vector $(l, w, h)^T$ is assumed to be identically and independently distributed at different parking spaces. Here, the probability density functions $p(l)$, $p(w)$, and $p(h)$ are pre-learned based on 120 parked cars. On the contrary, the 3-D ground plane of the parking lot is defined as a 2-D plane $(X, Y, 0)$. Inside the i th parking space, we assume the projection of the car center on the ground plane is represented by $(X_i, Y_i, 0)$, where X_i and Y_i are modeled as two randomly distributed Gaussian random variables with the probability density functions $p(X_i)$ and $p(Y_i)$. The mean values of $p(X_i)$ and $p(Y_i)$ are set to be the center of the i th parking space on the ground plane. Moreover, we assume the location pattern of parked cars in different parking spaces is similar. That is, we assume the variances of $p(X_i)$ and $p(Y_i)$ are independent of i . To train the variance values of $p(X_i)$ and $p(Y_i)$, we measured for each of these 120 cars the deviation of the car center from the center of the parked space.

To predict the shadowed regions, we model the lighting condition in the 3-D scene. In general, we may assume there

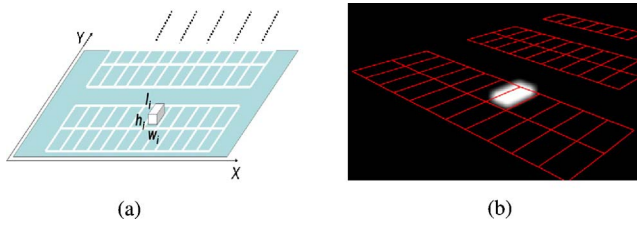


Fig. 4. (a) 3-D car model. (b) Expected car labeling map.

are two major types of illumination in an outdoor environment: direct illumination from the sun and ambient illumination from the sky. For each image pixel, it may be lighted by the skylight only, or lighted by both skylight and sunlight. Basically, shadow reflects the contrast of brightness for regions illuminated by different types of lighting. If the sunlight exists in the environment, the regions lighted by skylight only appear to be shadowed. On the contrary, when sunlight is absent, we assume there is no shadowed region.

Moreover, when sunlight is present, we assume the direction of sunlight is represented by a 3-D vector $[D_X(t), D_Y(t), D_Z(t)]^T$, which is a function of time t . In our approach, the 3-D scene model of a parking lot is determined by the parameter set Φ , where $\Phi = \{D_X(t), D_Y(t), D_Z(t), \{S_L(i), l_i, w_i, h_i, X_i, Y_i, \text{ for } i = 1, 2, \dots, N_s\}\}$. In Φ , $\{S_L(i)\}$ is the main unknown variable. The detailed deduction of the sunlight direction $[D_X(t), D_Y(t), D_Z(t)]^T$ is to be explained later in Section III-A and Appendix I.

B. Generation of Expected Labeling Maps

1) *Object Labeling Model*: In our system, once the 3-D scene parameters Φ are given, the expected object labeling and the expected shadow labeling on the captured images are automatically generated. Based on the projection matrix of the camera, a synthesized car parked at $(X_i, Y_i, 0)$ inside the i th parking space, with length l_i , width w_i , and height h_i , is projected onto the camera view to get the projection image $M_i(m, n|X_i, Y_i, l_i, w_i, h_i)$, which has the value 1 if the pixel (m, n) is within the projected region, and 0 otherwise. Since the size parameters (l_i, w_i, h_i) and the parked location (X_i, Y_i) may vary from car to car, we take into account the prior probabilities $p(l_i)$, $p(w_i)$, $p(h_i)$, $p(X_i)$, and $p(Y_i)$ and define the expected car labeling map to be a probabilistic map $C_i(m, n)$, which is the expectation value of $M_i(m, n|X_i, Y_i, l_i, w_i, h_i)$, that is

$$C_i(m, n) = E_{X_i, Y_i, l_i, w_i, h_i} [M_i(m, n|X_i, Y_i, l_i, w_i, h_i)]. \quad (5)$$

On the contrary, since the object type of an image pixel is either “car” or “ground,” the expected ground labeling map is deduced to be

$$G_i(m, n) = 1 - E_{X_i, Y_i, l_i, w_i, h_i} [M_i(m, n|X_i, Y_i, l_i, w_i, h_i)]. \quad (6)$$

In our system, we numerically calculate the expectation in (5) and (6) based on the Monte Carlo approach. Here, based

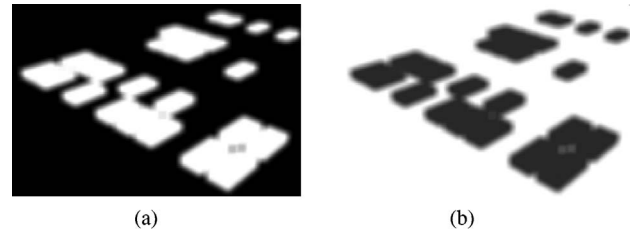


Fig. 5. (a) Expected car labeling. (b) Expected ground labeling.

on the prior probabilities $p(l_i)$, $p(w_i)$, $p(h_i)$, $p(X_i)$, and $p(Y_i)$, we draw a large set of sample tuples. For each sample tuple, say $(l_k, w_k, h_k, X_k, Y_k)$, we synthesize a projection image. By averaging all projection images for all sample tuples, we get a probability map that approximates $C_i(m, n)$. In Fig. 4(b), we show the expected car labeling map of the car in Fig. 4(a).

While taking all parking spaces into consideration, an image pixel at (m, n) in the i th parking space may get occluded not only by a car parked at that parking space but also by a car parked at an adjacent parking space. To model the inter-occlusion effect in the object labeling model, we define the probability

$$p(h^0(m, n) = 0|S_L) = \prod_{i=1}^{N_s} [G_i(m, n)^{S_L(i)}] \quad (7)$$

where $S_L(i)$ is the status of the i th parking space. With (7), the probability of car labeling at (m, n) given the status of all parking spaces can be formulated as follows:

$$p(h^0(m, n) = 1|S_L) = 1 - \prod_{i=1}^{N_s} [G_i(m, n)^{S_L(i)}]. \quad (8)$$

In Fig. 5(a) and (b), we show the examples of $p(h^0(m, n) = 1|S_L)$, and $p(h^0(m, n) = 0|S_L)$, respectively.

2) *Shadow Labeling Model*: Similarly, by using a cube model for a parked car, the expected shadowed regions on the ground plane can be quickly determined in the 3-D space whenever the sunlight direction is known and the status of parking spaces are determined. An example is illustrated in Fig. 6. Here, we define $T_i(m, n|X_i, Y_i, l_i, w_i, h_i)$ to be the projected shadow labeling image generated by a car parked at $(X_i, Y_i, 0)$ inside the i th parking space, with length l_i , width w_i , and height h_i . Similarly, by taking into account the prior probabilities $p(l_i)$, $p(w_i)$, $p(h_i)$, $p(X_i)$, and $p(Y_i)$, we define the expected shadow labeling map $S_i(m, n)$ in a probabilistic manner as follows:

$$S_i(m, n) = E_{X_i, Y_i, l_i, w_i, h_i} [T_i(m, n|X_i, Y_i, l_i, w_i, h_i)]. \quad (9)$$

Similarly to (6), the expected non-shadow labeling map is defined as $US_i(m, n) = 1 - S_i(m, n)$. In Fig. 6(b), we show the expected shadow labeling map of the car in Fig. 6(a).

To model the shadow labeling model $p(h^L(m, n)|S_L)$ with the consideration of all parking spaces, we define

$$p(h^L(m, n) = 0|S_L) = \prod_{i=1}^{N_s} [US_i(m, n)^{S_L(i)}] \quad (10)$$

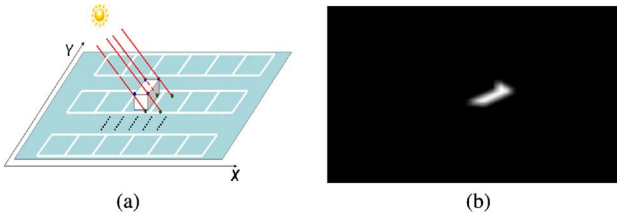


Fig. 6. (a) Shadow formation. (b) Expected shadow labeling map.

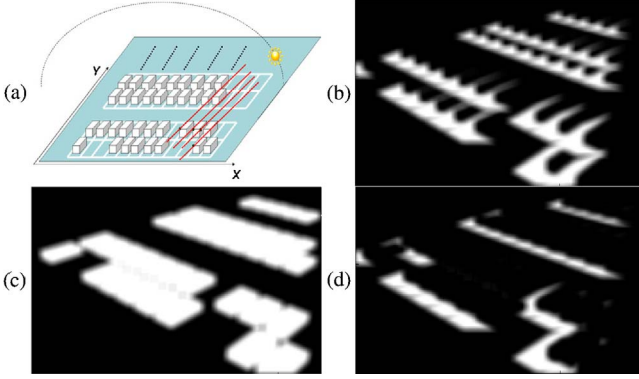


Fig. 7. (a) Illusion of shadow formation. (b) Expected shadow labeling map. (c) Expected car labeling map. (d) Refined shadow labeling map.

where $S_L(i)$ is the status of the i th parking space. With (10), the probability of shadow labeling at (m, n) given S_L can be modeled by

$$p(h^L(m, n) = 1 | S_L) = 1 - \prod_{i=1}^{N_s} [US_i(m, n)^{S_L(i)}]. \quad (11)$$

In Fig. 7(a) and (b), we show an example of the 3-D parking lot model and its expected shadow labeling map. To simplify the problem, we ignore the shadows cast upon the parked cars and only consider the shadows cast on the ground plane. With this assumption, a pixel with a higher probability of car labeling is less likely to be shadowed. Hence, we refine the probabilistic shadow labeling map to be

$$\begin{aligned} p(h^L(m, n) = 1 | S_L) \\ = (1 - p(h^O(m, n) = 1 | S_L)) \times (1 - \prod_{i=1}^{N_s} [US_i(m, n)^{S_L(i)}]). \end{aligned} \quad (12)$$

A refined shadow labeling map is shown in Fig. 7(d).

C. Sunlight Direction

To generate the expected shadow labeling map, we need the direction of sunlight. The information of sunlight parameters is available on the Internet, such as the U.S. Naval Observatory website [26]. By providing the date and the geo-location of the parking lot, including longitude and latitude from a global position system, the web service can provide samples of sunlight direction for every 10 min.

In our system, we adopt the concept proposed in [27] to calculate the sunlight direction. In principle, the solar motion model and the sunlight direction can be estimated based on the variations of intensity values in a day. In a single day,

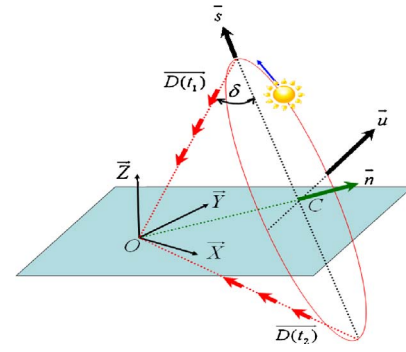


Fig. 8. Illustration of solar movement and sunlight direction.

the solar motion follows a circle on the solar plane in the 3-D space, with a constant angular frequency ω_s , as illustrated in Fig. 8. The angular frequency depends mainly on the self-rotation of the Earth and is known in advance. The whole set of sunlight directions in a day form a conical surface and the cone aperture is equal to $\pi - 2\delta$, where δ is the sun declination angle approximated as

$$\delta = -23.45^\circ \cdot \cos \left[\left(\left(\frac{360}{365} \right) \cdot (N_d + 10) \right)^\circ \right]. \quad (13)$$

In (13), N_d is the number of days counted from January 1 to the current date. With this cone model, the sunlight direction over time can be parametrically represented by

$$\overrightarrow{D}(t) = -\{\sin(\delta)\vec{n} + \cos(\delta)[\cos(\omega_s(t-t_\theta))\vec{u} + \sin(\omega_s(t-t_\theta))\vec{s}]\} \quad (14)$$

where \vec{u} is a unit reference vector on the solar plane at time t_θ , \vec{n} is the normal vector of the solar plane, and $\vec{s} = \vec{n} \times \vec{u}$.

On the contrary, we assume the scene surfaces are mainly Lambertian surfaces. Hence, the intensity value reflected from a surface is proportional to the incident angle of the incident light with respect to the surface normal. The intensity value at an image pixel will climb to its maximum when the subtended angle between the corresponding surface normal vector and the sunlight direction reaches the minimum. As explained in Appendix I, if \vec{P} is the normal vector of a surface patch in the 3-D scene, the intensity value at the corresponding image pixel can be approximated as follows:

$$I_{sun}(m, n, t) = B(m, n) \cos(\omega_s t - \theta_p(m, n)) + C(m, n) \quad (15)$$

which is a scaled cosine function plus a constant offset. Moreover, if θ represents the angle subtended by \vec{u} and the projection of \vec{P} on the solar plane, the phase shift θ_p of the cosine function is equal to θ up to a constant offset. In principle, if we pick up three image pixels whose 3-D scene points lie on different surfaces with linearly independent normal vectors, we can deduce the geometric relationship between the solar plane and these three surface normal vectors [27]. For detailed deduction, please refer to Appendix I.

In Fig. 9(a), we show three manually selected image pixels in the parking lot scene, one from the driveway and two from the bushes. These image pixels locate at three mutually orthogonal planes. The intensity profile of a pixel in the green

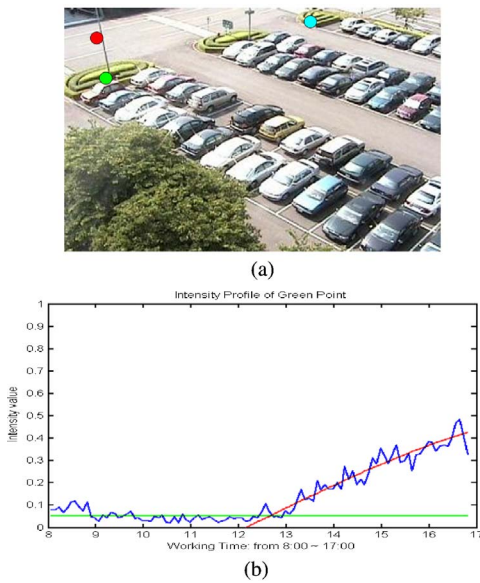


Fig. 9. (a) Parking lot image with three manually selected image pixels, marked in RGB. (b) Intensity profiles (blue) of the green pixel, overlapped with the fitted skylight profile (green) and the fitted skylight + sunlight profile (red).

region is shown in Fig. 9(b) as an example. By identifying the phase shift θ_p from each of these three intensity profiles, we can determine the sunlight direction $\vec{D}(t)$ at any time instant t . Moreover, if a parking lot cannot provide these three mutually independent planes, an artificial cube is recommended to be set up in the parking lot scene.

IV. BOTTOM-UP MESSAGES FROM OBSERVATION LAYER

In our system, the bottom-up messages are embedded in the likelihood function $p(I_L|H_L)$, which links the observation data with the labeling results. We assume the observation nodes are conditionally independent when the status of the labeling layer is given. In addition, we assume the connections between the observation layer and the labeling layer are one-to-one and these connections can be modeled in terms of a “classification energy” $E_D[I_L(m, n), H_L(m, n)]$. Moreover, since the local labeling results of adjacent nodes are usually highly correlated, an “adjacency energy” $E_A[I_L(m, n), H_L(m, n); N_p]$ is used. By combining these two kinds of energy, we have

$$p(I_L|H_L) = K \cdot \prod_m \prod_n e^{-E_D[I_L(m, n), H_L(m, n)]} e^{-E_A[I_L(m, n), H_L(m, n); N_p]} \quad (16)$$

In (16), N_p denotes a neighborhood around (m, n) and K is a normalization term.

A. Classification Energy

1) *Energy Model*: In our approach, we attempt to convert the RGB color features \mathbf{I}_{RGB} of each pixel into a semantic labeling. Here, we model the classification energy as follows:

$$E_D[I_L(m, n), H_L(m, n)] = -\ln(p(\mathbf{I}_{\text{RGB}}(m, n)|h^O(m, n), h^L(m, n))) \quad (17)$$

where $p(\mathbf{I}_{\text{RGB}}|h^O, h^L)$ is the conditional probability distribution of \mathbf{I}_{RGB} given the semantic labeling (h^O, h^L) . In (17), $h^O(m, n)$ could be C or G , and $h^L(m, n)$ could be S or US . Since the lighting condition changes from time to time, we need to dynamically adjust $p(\mathbf{I}_{\text{RGB}}|h^O, h^L)$. Based on the image formation model explained in Appendix II, the trichromatic color vector \mathbf{I}_{RGB} at an image pixel can be represented as $\mathbf{I}_{\text{RGB}} = \|\mathbf{I}_{\text{RGB}}\| \mathbf{R} \mathbf{i}$, where $\|\mathbf{I}_{\text{RGB}}\|$ is the norm of \mathbf{I}_{RGB} , \mathbf{R} is a 3×3 matrix depending on surface reflectance, \mathbf{i} is a vector depending on illumination, and $\|\mathbf{R} \mathbf{i}\| = 1$. With this image formation model, we formulate $p(\mathbf{I}_{\text{RGB}}|h^O, h^L)$ as follows:

$$p(\mathbf{I}_{\text{RGB}}|h^O, h^L) = p(\|\mathbf{I}_{\text{RGB}}\| | h^O, h^L) p(\mathbf{R}|h^O) p(\mathbf{i}|h^L). \quad (18)$$

Since the reflectance of target objects (ground or cars) can be learned beforehand but the lighting condition is varying over time, $p(\mathbf{R}|h^O)$ is learned off-line while $p(\|\mathbf{I}_{\text{RGB}}\| | h^O, h^L)$ and $p(\mathbf{i}|h^L)$ are determined dynamically. Here, we build those probability models similar to the approach of [28] with a few modifications. First, instead of training the reflectance functions of only two objects (grass and ground in [28]) based on a single singular value decomposition (SVD) over one set of data, our application needs to collect the reflectance functions of various cars at different positions and at different time instants. This requires multiple SVDs over different sets of data. Hence, we need to register the solutions of different SVDs to deal with the ambiguity in SVD. Second, instead of clustering the daylight spectrums into only three classes, we determine $p(\mathbf{i}|h^L)$ dynamically to deal with the continuously changing lighting condition. Third, in [28], the trained chromaticity values of different classes are used to initialize the classification of image content. Their intensity model is then on-line determined. However, owing to the wide range of car appearance, some cars may get confused with the ground in the chromaticity space. In our approach, we add in the scene knowledge to dynamically determine the intensity model $p(\|\mathbf{I}_{\text{RGB}}\| | h^O, h^L)$. Basically, given an image, there are two types of light: skylight and sunlight. Also, the ratio of reflectance between any two scene patches can be well pre-learned. These two facts offer a possibility to on-line determine the intensity model of scene patches based on a few reference patches. Below, we explain the details of our approach.

2) *Learning of $p(\mathbf{R}|h^O)$* : We collected 5000 training samples of ground and cars to learn $p(\mathbf{R}|h^O = G)$ and $p(\mathbf{R}|h^O = C)$, respectively. Since the camera pose in our system is fixed, the captured images can be easily registered. To get the reflectance function of an object, we select a small surface patch with uniform illumination. To simplify the problem, we normalized \mathbf{I}_{RGB} by its norm to get the normalized RGB $\mathbf{I}_{\text{RGB}}^N = \mathbf{I}_{\text{RGB}} / \|\mathbf{I}_{\text{RGB}}\|$. Assume there are P pixels inside the patch and we collect the samples for F registered frames. The illumination condition is the same for the whole patch at a certain time instant, but could be different at different time instants. On the contrary, the reflectance function could be different at different image pixels but is temporally invariant at the same pixel. Hence, for an image pixel at the spatial location \vec{p} , its normalized RGB value at time instant k can be

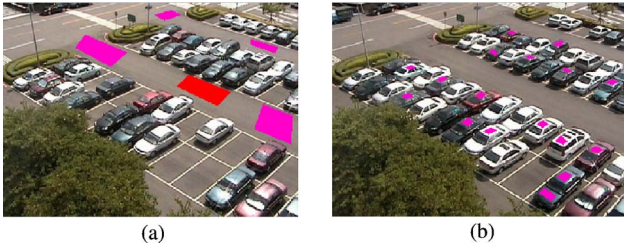


Fig. 10. (a) Reference ground patch (red) and the ground patches (pink) for the learning of ground reflectance function. (b) Car patches (pink) for the learning of car reflectance function.

expressed as follows:

$$\mathbf{I}_{\text{RGB}}^{\text{N}}(\vec{p}, k) = \mathbf{R}(\vec{p})\mathbf{i}(k). \quad (19)$$

By arranging the normalized RGB values of all pixels inside the surface patch over F frames into a $3P \times F$ matrix, we obtain the formula as follows:

$$\begin{aligned} \mathbf{M}_{\text{RGB}}(\vec{p}, \mathbf{k}) &\equiv \begin{bmatrix} \mathbf{I}_{\text{RGB}}^{\text{N}}(\vec{p}_1, k_1) & \cdots & \mathbf{I}_{\text{RGB}}^{\text{N}}(\vec{p}_1, k_F) \\ \vdots & \ddots & \vdots \\ \mathbf{I}_{\text{RGB}}^{\text{N}}(\vec{p}_P, k_1) & \cdots & \mathbf{I}_{\text{RGB}}^{\text{N}}(\vec{p}_P, k_F) \end{bmatrix}_{3P \times F} \\ &= \begin{bmatrix} \mathbf{R}(\vec{p}_1) \\ \vdots \\ \mathbf{R}(\vec{p}_P) \end{bmatrix}_{3P \times 3} \begin{bmatrix} \mathbf{i}(k_1) & \cdots & \mathbf{i}(k_F) \end{bmatrix}_{3 \times F} \equiv \mathbf{M}_{\mathbf{R}}(\vec{p})\mathbf{M}_{\mathbf{i}}(\mathbf{k}) \end{aligned} \quad (20)$$

where $\vec{p} = \{\vec{p}_1, \dots, \vec{p}_P\}$ is the spatial locations of the P pixels and $\mathbf{k} = \{k_1, \dots, k_F\}$ is the temporal indexes of the F frames.

Given \mathbf{M}_{RGB} , we can decompose it into a reflectance matrix $\mathbf{M}_{\mathbf{R}}$ and an illumination matrix $\mathbf{M}_{\mathbf{i}}$, up to a 3×3 non-singular matrix \mathbf{Q} . That is, if $\mathbf{M}_{\mathbf{R}1}$ and $\mathbf{M}_{\mathbf{i}1}$ is a pair of matrices that decompose \mathbf{M}_{RGB} , then $\mathbf{M}_{\mathbf{R}2} = \mathbf{M}_{\mathbf{R}1}\mathbf{Q}$ and $\mathbf{M}_{\mathbf{i}2} = \mathbf{Q}^{-1}\mathbf{M}_{\mathbf{i}1}$ is another decomposition pair. Fortunately, in the detection of vacant parking spaces, we only care about the difference in the surface reflectance matrix \mathbf{R} but not the true value of \mathbf{R} . As long as we fix the matrix $\mathbf{M}_{\mathbf{i}}$, two surface patches with different \mathbf{R} will always have different $\mathbf{M}_{\mathbf{R}}$.

To decompose \mathbf{M}_{RGB} , we adopt the SVD method. The SVD process is applied over several planar patches to collect samples for the ground reflectance function and car reflectance function. For the car samples, we select the car roof as the planar patch, which is usually parallel to the ground plane. To deal with the ambiguity in matrix decomposition, we collected a set of image frames and manually selected a ground region in the parking lot scene as the reference patch, shown as the red patch in Fig. 10(a). By performing SVD over the reference patch, we got the reference truth $\mathbf{M}_{\mathbf{R}0}$ and $\mathbf{M}_{\mathbf{i}0}$. The reference truth $\mathbf{M}_{\mathbf{i}0}$ is used to register the illumination matrix of another spatial patch that are under the same lighting condition in the same set of image frames. On the contrary, the reference truth $\mathbf{M}_{\mathbf{R}0}$ is used to register the reflectance matrix of the reference ground patch in another set of image frames. Based on SVD, with enough reflectance samples of cars and ground, we can construct the reflectance probability model $p(\mathbf{R}|h^O)$.

3) *Learning of $p(\mathbf{i}|h^L)$* : The illuminant probability model $p(\mathbf{i}|h^L)$ is determined based on the pre-trained model and the

current image observation. Given an image, there are two types of regions: shadowed regions and unshadowed regions. By collecting many illumination samples \mathbf{i} 's in shadowed regions and unshadowed regions, we can approximate $p(\mathbf{i}|h^L = 'S')$ and $p(\mathbf{i}|h^L = 'US')$. Since the reflectance matrix \mathbf{R} of a scene patch can be learned in advance, we extract the illuminant component of some manually selected shadowed and unshadowed regions to learn the off-line models $p_{\text{off}}(\mathbf{i}|h^L = 'S')$ and $p_{\text{off}}(\mathbf{i}|h^L = 'US')$. On the other hand, to deal with the continuously changing lighting condition, we also build the on-line models $p_{\text{on}}(\mathbf{i}|h^L = 'S')$ and $p_{\text{on}}(\mathbf{i}|h^L = 'US')$ based on the current image observation. The illuminant probability model is then determined based on a weighted combination of off-line and on-line models. That is

$$p(\mathbf{i}|h^L = 'S') = \omega_1 p_{\text{on}}(\mathbf{i}|h^L = 'S') + (1 - \omega_1) p_{\text{off}}(\mathbf{i}|h^L = 'S') \quad (21)$$

and

$$p(\mathbf{i}|h^L = 'US') = \omega_2 p_{\text{on}}(\mathbf{i}|h^L = 'US') + (1 - \omega_2) p_{\text{off}}(\mathbf{i}|h^L = 'US'). \quad (22)$$

Here, ω_1 and ω_2 are determined by the ratio of the on-line training samples to the total training samples.

During on-line modeling, we need to determine whether a given illuminant sample is shadowed or unshadowed. Here, for the period from 10:30 to 14:00, we suppose all samples are unshadowed. For the other periods, the lighting situation is more complicated. In our parking lot scene, we identified a few regions that are always unshadowed, like some regions in the driveway. These driveway regions can be used as the reference regions for the 'unshadowed' case for both skylight-plus-sunlight case and skylight-only case. On the contrary, as shown in Fig. 9(b), the green region in the bush in Fig. 9(a), together with all the other planes parallel to that green region, is only lighted by skylight in the morning, while the blue region in Fig. 9(a), together with all the other planes parallel to that blue region, is only lighted by skylight in the afternoon. These two types of regions can be used as the reference regions for the "shadowed" case when both sunlight and skylight are present. In Section V-A, we will further explain how we check the presence of sunlight in the current image.

4) *Learning of $p(\|\mathbf{I}_{\text{RGB}}\| | h^O, h^L)$* : The intensity information $\|\mathbf{I}_{\text{RGB}}\|$ is crucial in distinguishing cars from ground, especially when some cars may get confused with the ground in the chromaticity space. Unfortunately, $\|\mathbf{I}_{\text{RGB}}\|$ is affected by the lighting source, the object reflectance, the object geometry, and even some unknown factors in the imaging pipeline such as automatic gain control and white balance. Therefore, the modeling of the intensity model $p(\|\mathbf{I}_{\text{RGB}}\| | h^O, h^L)$ is more difficult. To build an adaptive intensity model based on current image observation, we propose a simplified linear model as expressed in (23) to model the intensity mapping from one object type (O_1) in a scene patch to another object type (O_2) in another scene patch, under the same illumination type (L) as follows:

$$g_{O_2, L} = a_{O_2, O_1, L} \cdot g_{O_1, L} + \mathbf{n}_{O_2, O_1, L}. \quad (23)$$

In (23), $g_{O,L}$ denotes an intensity sample from the object type O under the illumination type L . Note that $g_{O,L}$ value is equal to the norm $\|\mathbf{I}_{\mathbf{RGB}}\|$ of a color pixel. $a_{O_1,O_2,L}$ represents the intensity ratio between objects O_2 and O_1 under illumination type L . $\mathbf{n}_{O_1,O_2,L}$ is defined as a zero mean Gaussian noise that expresses the uncertainty in modeling the intensity ratio. Even though $a_{O_1,O_2,L}$ is actually a random variable, we found a deterministic setting works very well in our experiments. Here, we learn $a_{O_1,O_2,L}$ and the variance of $\mathbf{n}_{O_1,O_2,L}$ based on the equations as follows:

$$\hat{a}_{O_2,O_1,L} = \overline{g_{O_2,L}} / \overline{g_{O_1,L}} \quad (24)$$

and

$$\hat{\sigma}_{\mathbf{n}_{O_2,O_1,L}}^2 = \hat{\sigma}_{g_{O_2,L}}^2 - \hat{a}_{O_1,O_2,L}^2 \hat{\sigma}_{g_{O_1,L}}^2. \quad (25)$$

In (24) and (25), $\overline{g_{O,L}}$ and $\hat{\sigma}_{g_{O,L}}^2$ are the sample mean and sample variance of the intensity training samples. The training samples are manually collected from training image patches, with classified light type L and object type O .

In our system, a few transformation models were pre-learned to generate the intensity model $p(\|\mathbf{I}_{\mathbf{RGB}}\| | h^O, h^L)$ dynamically. Here, we adopt the aforementioned reference regions, like the driveway regions that are always unshadowed and the bush regions that are always lighted by the skylight only. By using these reference regions, in which the lighting condition is already known, we learned the transformation models from each of these reference regions to the parking space ground and to the cars, respectively. After that, based on the learned transformation models and the current intensity values at these reference regions, we can dynamically construct the intensity model $p(\|\mathbf{I}_{\mathbf{RGB}}\| | h^O, h^L)$. Similar to the deduction of the sunlight direction, if the parking lot scene cannot provide such reference regions, an artificial cube is suggested to be set up in the scene to form reference regions.

B. Adjacency Energy Model

In the parking lot scene, the local decisions of two adjacent labeling nodes are usually highly correlated. In our system, with the use of the original intensity image $I_L(m, n)$, we define the adjacency energy $E_A[I_L(m, n), H_L(m, n); N_p]$ in terms of a Markov random field model [29] to provide the smoothness constraint between adjacent labeling nodes. Here, we define

$$\begin{aligned} & E_A[I_L(m, n), H_L(m, n); N_p] \\ & \equiv \beta \times \sum_{\Delta m=-p}^p \sum_{\Delta n=-p}^p C_A[I_L, H_L, mn, \Delta m, \Delta n] \end{aligned} \quad (26)$$

where

$$\begin{aligned} & C_A[I_L, H_L, m, n, \Delta m, \Delta n] \\ & \equiv (1 - \delta[H_L(m, n), H_L(m + \Delta m, n + \Delta n)]) \\ & \quad \times G_S(\|I_L(m, n) - I_L(m + \Delta m, n + \Delta n)\|). \end{aligned} \quad (27)$$

In (26), N_p denotes the $(2p + 1) \times (2p + 1)$ neighborhood around (m, n) , and β is a pre-selected penalty constant. In (27), $\delta[p_a, q_a]$ is defined as 1 if $p_a = q_a$, and 0 otherwise. The

function G_S is designed to be a function similar to a logistic sigmoid function as follows:

$$G_S(U) = (1 - e^{\rho(U-C_{th})}) / (1 + e^{\rho(U-C_{th})}). \quad (28)$$

Here, $G_S(U)$ is used to preserve the discontinuity of the original image. Both C_{th} and ρ are determined empirically. With (26), $L_i(m, n)$ and $L_i(m + \Delta m, n + \Delta n)$ tend to share the same label when the difference between $I_i(m, n)$ and $I_i(m + \Delta m, n + \Delta n)$ is small, and tend to have different labels otherwise.

V. VACANT PARKING SPACE DETECTION

A. Optimal Inference of Parking Space Status

With the top-down knowledge and the bottom-up message, we can infer the optimal H_L^* and S_L^* by solving the optimization problem in (2). In our approach, we get the initial guess of $H_L(m, n)$ by find the labeling that minimizes the classification energy in (17). That is, we find the labeling image $H_L^i(m, n)$ such that

$$H_L^i(m, n) = \arg \min_{H_L} E_D[I_L(m, n), H_L(m, n)]. \quad (29)$$

On the contrary, since the status inference of a parking space depends on its neighboring parking spaces, we need to take into account relevant parking spaces when we infer the status of a parking space. In our experiments, a parked car casts a shadow to the right in the morning and to the left in the afternoon. Hence, we sequentially infer the status of each parking space from the bottom row to the top row and from left to right in the morning, and reverse the order in the afternoon. In Fig. 11, we show an example in the status determination of a parking space. Due to the direction of sunlight, we check the parking spaces from left to right and from bottom to top. The red regions indicate those parking spaces whose status has already been inferred. The yellow circle indicates the parking space to be inferred at this moment. The green triangles indicate the relevant parking spaces. In this case, by trying different status combination of A and B spaces, four status hypotheses are to be tested. For each status hypothesis, we deduce the optimal $H_L(m, n)$ by using the graph-cuts algorithm [23]–[25], with the initial guess $H_L^i(m, n)$. The status hypothesis that achieves the maximum posterior probability is picked to infer the status of the current parking space. In our process, since the status of a parking space is only affected by its adjacent spaces, the system complexity grows linearly as the number of parking spaces increases.

Moreover, in an outdoor environment, the sunlight does not always exist. In the inference of parking space status, we need to determine whether the sunlight is present or not. In our approach, we first perform the optimal labeling based on the assumption that sunlight is present. After the optimal inference for the whole image, we divide those ‘‘ground’’ pixels into shadowed pixels and unshadowed pixels. In principle, if the sunlight is present, the RGB values of these two pixel groups

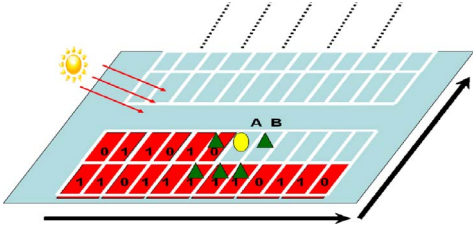


Fig. 11. Illustration of parking space status inference.

should reveal obvious difference. Hence, by calculating the Davies-Bouldin index (*DBI*) [30], which is defined as

$$DBI = (S_S + S_{US}) / (\|\mu_S - \mu_{US}\|) \quad (30)$$

we can decide whether to accept the “presence” hypothesis or not. In (30), μ_s and μ_{us} are the mean RGB values of the shadowed cluster and the unshadowed cluster. S_s and S_{us} are the centroid distance of these two clusters defined as follows:

$$S_c = \left(\sum_{i=1}^{n_k} \|f_i - \mu_c\| \right) / n_k \quad (31)$$

where $c \in \{S, US\}$, n_k is the total pixel number of the cluster, and f_i is the RGB value of the i th pixel. When the *DBI* is smaller than a predefined threshold, we accept the “presence of sunlight” hypothesis. Otherwise, we take the “absence of sunlight” hypothesis and perform the optimal inference over the whole image again to get the final detection result.

B. Refinement of Classification Energy Model

In our system, after performing the optimal inference over an image, we obtain a semantic labeling (h^O, h^L) of the image that may provide useful information for the refinement of $p(\mathbf{IRGB}|h^O, h^L)$. The inferred semantic labeling (h^O, h^L) includes not only the bottom-up information but also the top-down knowledge. With the inclusion of the top-down knowledge, some pixels, which would be incorrectly labeled if only based on the classification models, can be correctly labeled. Those pixels usually correspond to non-Lambertian surfaces, like the car windows. Hence, based on the inferred optimal labeling (h^O, h^L), we recompute the classification model $p(\mathbf{IRGB}|h^O, h^L)$ by checking the distribution of \mathbf{IRGB} in the current image over different object types and different lighting types. The new model is then merged into the existing model for refinement as follows:

$$p_{refind}(\mathbf{IRGB}|h^O, h^L) = w_{new} \cdot p_{new}(\mathbf{IRGB}|h^O, h^L) + w_{old} \cdot p_{old}(\mathbf{IRGB}|h^O, h^L). \quad (32)$$

In (32), w_{old} and w_{new} determine the weights of the existing model and the new model. In our system, we empirically select (w_{old}, w_{new}) to be (0.2, 0.8). Based on the refined model, the optimal labeling is re-estimated again. This optimization-refinement process is iteratively performed until the status inference of the parking spaces becomes stable. In our experiments, the refinement process usually converges in one or two iterations.

C. System Setup and Online Vacant Space Detection

To implement the whole system, several preparatory processes are required, as listed below.

- 1) Calibration steps.
 - a) Define a 3-D coordinate system for the parking lot. Measure the 3-D location of each parking space. Here, we record the 3-D information in a blueprint.
 - b) Perform camera calibration to compute the camera projection matrix.
- 2) Off-line learning of 3-D information.
 - a) Estimate the parameters of solar direction model based on the method introduced in Section III-C.
 - b) Collect 3-D training samples of vehicle length, width, and height to train the priors $p(l)$, $p(w)$, $p(h)$.
 - c) Collect 3-D location deviation samples to train $p(X)$ and $p(Y)$.
- 3) Off-line learning of 2-D information.
 - a) Collect reflectance samples to train the reflectance models of ground and cars, based on the method mentioned in Section IV-A2.
 - b) For different time periods, manually select unshadowed and shadowed reference regions in the image.
 - c) Collect illuminant samples to train the off-line illuminant probability model of the shadowed regions and unshadowed regions, based on the method mentioned in Section IV-A3.
 - d) Based on the method mentioned in Section IV-A4, learn the intensity mapping models from each of these reference regions to the ground and to the cars.

In our experiments, it took about five days to finish the above system setup processes for each parking lot. After system setup, the following processes are performed to dynamically detect vacant parking spaces.

- 1) Determine the current sunlight direction based on the pre-learned solar movement model. This solar movement model is updated for every few days.
- 2) Based on the learned 3-D information, the sunlight detection, and the projection matrix, generate the expected object and shadow labeling models.
- 3) Extract illuminant samples from pre-selected reference regions to update the illuminant probability model.
- 4) Based on the pre-learned intensity mapping models, establish the intensity model of different classes.
- 5) Combine object reflectance models, illuminant probability models, and intensity models to build the classification models.
- 6) Incorporate classification models, expected labeling models, and adjacency model into the BHF to detect vacant parking spaces.

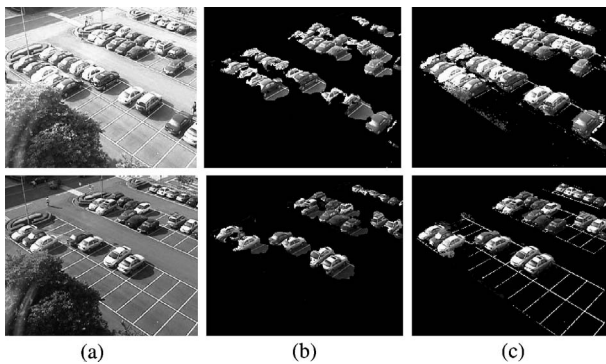


Fig. 12. Comparison of car pixel labeling. (a) Test images. (b) Regions labeled as car pixels based on [5]. (c) Regions labeled as car pixels based on the proposed method.

VI. EXPERIMENT RESULTS AND DISCUSSION

A. Experiment Setup and Test Data

In our experiments, we tested two different parking lots for performance evaluation. In each test, we set up an Internet protocol camera on the roof of a building near the parking lot. The camera was geometrically calibrated beforehand and monitored the status of parking spaces from morning to evening. Both experiments report similar detection accuracy. To avoid confusion, we mainly present the results and the analysis over the first parking lot. At the end of this section, we briefly present the detection performance over the second parking lot.

Fig. 1 shows a few image shots of the first parking lot. Within the image view, there are 46 parking spaces in total. To evaluate the performance of our system, we tested three image sequences under different weather conditions. The first sequence was captured in a normal sunny day. The second sequence was captured in a day with very strong sunlight so that there were plentiful over-exposed regions in the images. The third sequence was captured in a day with unstable lighting condition. In this sequence, the lighting condition dramatically switched between sunny and cloudy. For each sequence, the recording time was from 8:00 to 17:00. Since the status of the parking condition was slowly changing, we performed vacant parking space detection for every 5 min. In total, we tested the status of 14 766 spaces. In these three sequences, the shadow patterns varied from morning to evening. Sometimes, the shadowed regions suddenly disappeared when the sunlight was blocked by a cloud. The variations of illumination caused apparent drifts in color and brightness. These three sequences with vacant space detection results and ground truth are available at our website [31].

B. Object and Shadow Labeling

Many previous studies suggested the vacant spaces be detected by labeling the car pixels, such as Tsai *et al.* [5], or by labeling the ground pixels, such as Funck *et al.* [11]. In our method, we modeled both cars and ground plane for object labeling. In Fig. 12, we compare the results of car pixel labeling based on Tsai's method [5] and ours. Here, we show the image portions that were labeled as "car."

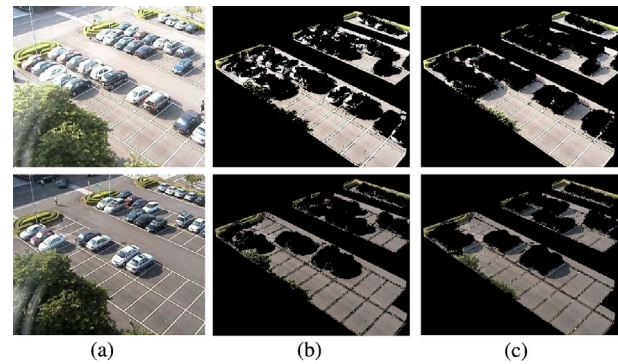


Fig. 13. Comparisons of ground pixel labeling. (a) Test images. (b) Regions labeled as ground pixels based on [11]. (c) Regions labeled as ground pixels based on our method.

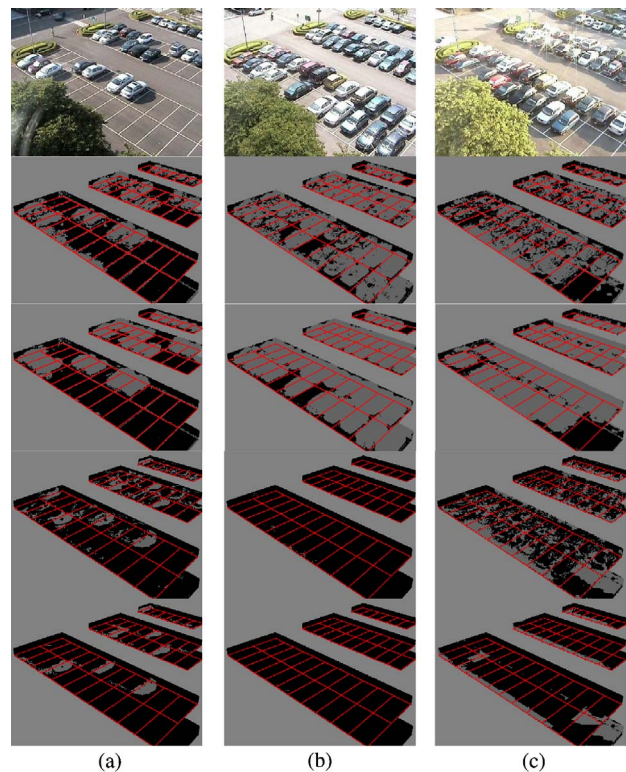


Fig. 14. Detection and labeling results at three different time instants. (a) Captured on a cloudy day. (b) Captured on a normal day. (c) Captured on a day with strong sunlight. For each case, the images from the top are the test image, the car labeling without scene knowledge, the car labeling with scene knowledge, the shadow labeling without scene knowledge, and the shadow labeling with scene knowledge.

Based on Tsai's method, many shadowed ground regions were labeled as car pixels, many over-exposed car regions were labeled as ground pixels, and some car regions were mistakenly labeled as ground pixels. In comparison, our parking space detection system provided more accurate car regions and was less sensitive to the shadow effect. In Fig. 13, we compare the results of ground pixel labeling based on [11] and our method. Both [11] and our method used adaptive models for labeling. However, the method in [11] did not take into account the shadow effect and many shadowed ground regions were classified as car pixels. In comparison,

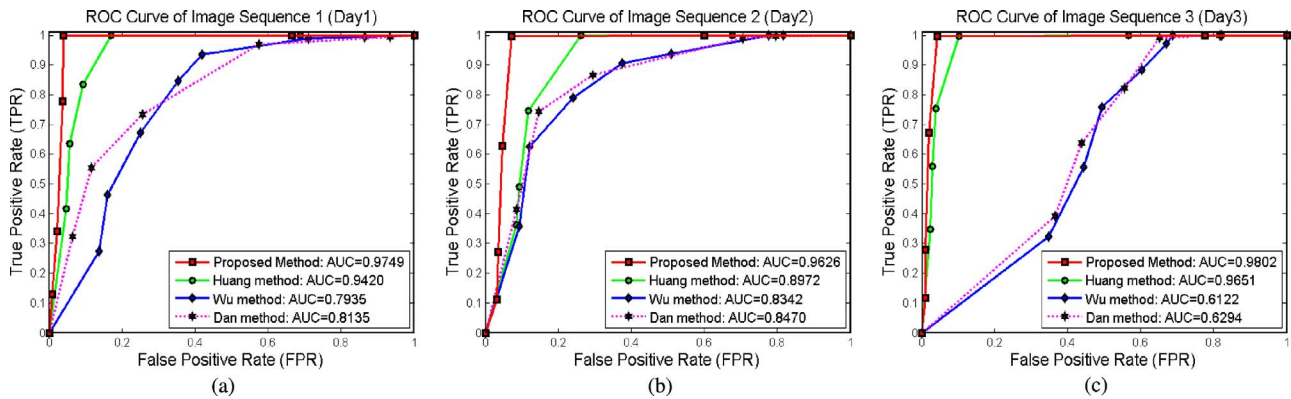


Fig. 15. ROC curves of our method, Huang’s method [12], Wu’s method [20], and Dan’s method [19], with the values of the area under ROC (AUC) for (a) Day 1, (b) Day 2, and (c) Day 3 image sequences.

most shadowed ground regions are correctly identified by our method.

Even though the proposed adaptive models can better handle the shadow effect, many pixels were still misclassified if the scene knowledge was not involved. An example is presented in Fig. 14, where we show the labeling results with and without the scene knowledge. Especially, there were some pepper-like errors inside the car regions as shown in Fig. 14(c) which were caused by the ambiguity in color appearance. It is difficult to remove those errors if we only rely on color models. In our system, the scene information in the expected labeling maps provides useful constraints to remove that kind of errors. To deal with the color ambiguity between dark cars and shadowed ground, the expected shadow labeling map clearly constrains the location of shadowed regions. On the contrary, if a region is to be occupied by a car, the expected object labeling map reveals the probable regions of car pixels and disfavors the occurrence of pepper-like labeling. Moreover, the expected object labeling map also reveals the expected occlusion effect and the perspective distortion. By taking into account these kinds of scene knowledge, more accurate and reliable detection results were obtained, as shown in Fig. 14.

C. Accuracy of Vacant Space Detection

To assess the detection accuracy of our system, we manually built the ground truth of 14 766 parking spaces. To evaluate our system from different aspects of environmental variations, we assessed the detection performance over a day, over different periods of a day, and over different regions of the parking lot. To quantitatively evaluate the performance, the false positive rate (FPR), false negative rate (FNR), and system accuracy (ACC) were calculated. In our simulation, the methods proposed by Dan [19], Wu *et al.* [20], and Huang *et al.* [12] were tested for comparison. The receiver operating characteristic (ROC) curves of the four methods are also plotted in Fig. 15 for comparison. Here, we consider three test image sequences. For each image sequence and each method, the area under the ROC curve (AUC) is also calculated and provided in Fig. 15 for reference.

As listed in Table I, the proposed method worked well in all three test sequences. We further divide a day into three

periods: morning (8:00 ~ 11:00), noon (11:00 ~ 14:00), and afternoon (14:00 ~ 17:00). Generally, the afternoon period has the most serious shadow effect, while the noon period has almost no shadow at all. By calculating the ACC of those three periods, we found the ACC is inversely proportional to the degree of shadow effect. Moreover, we also evaluated the performance of detection over different regions to evaluate the influence of perspective distortion. As shown in Table I, perspective distortion does not cause serious degradation in our experiments. Moreover, even though some portions of the first row were occluded by the trees, the proposed system still accurately inferred the status of the parking spaces.

We also implement our system in another parking lot. For each 320×240 tested image, there are 64 spaces inside. In total, we tested the statuses of 6912 spaces in that parking lot. In Fig. 16, we show some detection results in the second parking lot. The ACC, FPR, and FNR are 0.988, 0.0185, and 0.0097, respectively. The complete detection results of the second parking lot are also available at our website [31].

D. System Complexity

The whole system has been implemented in the Visual C++ environment on a personal computer with a 2.0GHz Pentium-4 central processing unit (CPU). It takes about 30 s to perform the space detection and labeling of parking spaces for a 320×240 color image with 46 spaces inside. The major CPU time is spent on building the online models, including the expected object labeling model, the expected shadow labeling model, and the color classification model. Even though the execution time takes a little while, the speed of the proposed system is still reasonably fast to support practical parking space detection systems.

E. Discussion and Future Works

Although the complexity of our system is already affordable for practical applications, the speed can be further boosted if we either adopt parallel programming techniques, such as open multiprocessing, to fully use the computing power of a multi-core processor, or to adopt general-purpose computing on graphics processing.

In our system, people in the parking lot may affect the detection of vacant parking spaces. However, people tend

TABLE I
PERFORMANCE COMPARISON OF FOUR VACANT SPACE DETECTION ALGORITHMS

Test Data	No. of Tested Spaces			Proposed Method			Huang [12]			Wu [20]			Dan [19]		
	Vacant	Parked	Total	FPR	FNR	ACC	FPR	FNR	ACC	FPR	FNR	ACC	FPR	FNR	ACC
Image Seq 1 (Day 1)	491	4431	4922	0.0004	0.0081	0.9988	0.0004	0.1690	0.9827	0.0111	0.7115	0.9193	0.0307	0.5748	0.9153
Image Seq 2 (Day 2)	278	4644	4922	0.0024	0.0324	0.9959	0.0002	0.2626	0.9850	0.0016	0.7837	0.9577	0.0101	0.7061	0.9537
Image Seq 3 (Day 3)	206	4716	4922	0.0040	0.0437	0.9943	0.0042	0.1019	0.9917	0.0018	0.7012	0.9739	0.0073	0.6524	0.9703
Morning period of Seq 3	380	4588	4968	0.0031	0.0105	0.9964	0.0011	0.2026	0.9835	0.0004	0.4955	0.9646	0.0097	0.4478	0.9594
Noon period of Seq 3	367	4601	4968	0.0015	0.0082	0.9980	0.0015	0.0381	0.9958	0.0045	0.8632	0.9360	0.0179	0.7629	0.9306
Afternoon period of Seq 3	228	4602	4830	0.0024	0.0658	0.9946	0.0024	0.3772	0.9799	0.0091	0.8920	0.9502	0.0195	0.6948	0.9494
First and second rows of Seq 3	644	6739	7383	0.0019	0.0233	0.9962	0.0025	0.1770	0.9823	0.0068	0.6960	0.9377	0.0179	0.5641	0.9381
Third and fourth rows of Seq 3	98	5359	5457	0.0015	0.0306	0.9980	0.0009	0.3163	0.9934	0.0028	0.6933	0.9871	0.0059	0.6933	0.9840
Fifth row of Seq 3	233	1693	1926	0.0065	0.0172	0.9922	0.0006	0.1373	0.9829	0.0024	0.8240	0.8982	0.0366	0.7554	0.8764

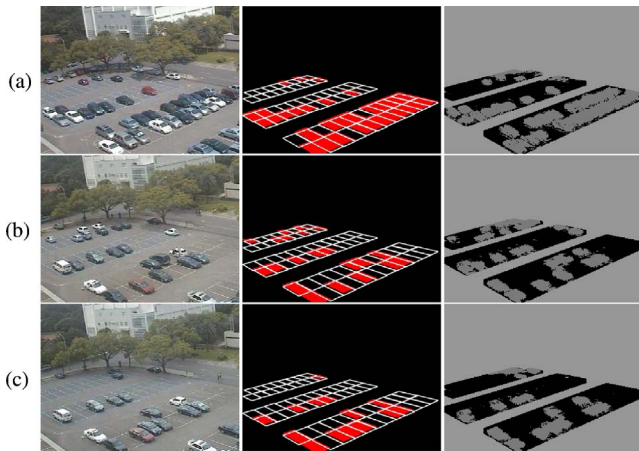


Fig. 16. Proposed detection and labeling results at three different time instants in another parking space. (a) Captured in the morning. (b) Captured in the afternoon. (c) Captured in the evening. For each case, the images from the left are the test image, the parking space detection results, and the car labeling results.

to dynamically move in the scene. By taking the temporal information into consideration, the problem of walking pedestrians can be relieved.

On the contrary, even though our system works very well in an outdoor parking area during the daytime, there exist still several challenging issues, like how to management an indoor parking area, how to detect vacant spaces in an outdoor parking lot during the night, and how to handle the unexpected shadow caused by other environmental objects. For an indoor parking area, the severe occlusion and the limited camera field of view could be the major challenges. Considering cost and efficiency, a possible solution is to build a low-cost camera sensor network. To detect vacant spaces in evening, we may need to consider multiple lighting sources while generating the expected shadow maps. We also require a new mechanism to handle the unpredictable lighting change caused by car headlights. All these discussions would be the future works of our vacant parking space detection system.

VII. CONCLUSION

We proposed a Bayesian hierarchical framework to simultaneously detect vacant parking spaces and interpret the scene content through image labeling. In practice, the challenges

of vacant space detection come from the shadow effect, the occlusion effect, the perspective distortion, and the dramatic luminance variations. In our system, we explicitly defined a scene model of the parking lot. Based on the model, the generation of shadow, the generation of occlusion, the variation of lighting, and the perspective distortion are closely coupled with the status of the parking spaces. By utilizing the proposed BHF, the scene generation process is well modeled and the optimal inference of the parking space status is resolved. Our results showed that our system can achieve up to 99% accuracy in vacant parking space detection under different lighting conditions. This system can also be integrated into an existing parking lot management system.

APPENDIX I

ESTIMATION OF SUNLIGHT DIRECTION

Based on the vectors \vec{u} , \vec{s} , and \vec{n} in Fig. 8, we define a coordinate system named USN and represent the sunlight direction as follows:

$$(-\cos(\delta)\cos(\omega_s(t-t_\theta)), -\cos(\delta)\sin(\omega_s(t-t_\theta)), -\sin(\delta))_{\text{USN}}. \quad (\text{A1})$$

Here, “USN” indicates the basis vectors \vec{u} , \vec{s} , and \vec{n} .

On the contrary, any unit vector \vec{P} in the 3-D scene can be represented as $(\cos\varphi\cos\theta, \cos\varphi\sin\theta, \sin\varphi)_{\text{USN}}$, where φ represents the angle between \vec{P} and the solar plane, and θ represents the angle subtended by \vec{u} and the projected vector of \vec{P} on the solar plane. In our system, we assume the scene surfaces are mainly Lambertian. Hence, if \vec{P} is the normal vector of a surface patch in the 3-D scene, the intensity value at the corresponding image pixel can be approximated as follows:

$$I_{\text{sun}} \propto \left\langle \vec{D}(t), \vec{P} \right\rangle \\ \propto -\cos(\delta)\cos(\phi)\cos(\omega_s t - \omega_s t_\theta - \theta) - \sin(\delta)\sin(\phi). \quad (\text{A2})$$

Based on (A2), I_{sun} can be modeled as follows

$$I_{\text{sun}}(m, n, t) = B(m, n)\cos(\omega_s t - \theta_p(m, n)) + C(m, n) \quad (\text{A3})$$

where the angular frequency of the cosine function is equal to the angular frequency of Earth’s self-rotation.

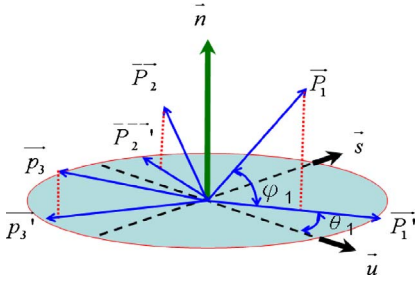


Fig. 17. Three normal vectors in the USN coordinate system.

Assume we denote $\vec{P}_1, \vec{P}_2,$ and \vec{P}_3 as the unit normal vectors of three selected surface patches in the parking lot. Since we manually select these three surface patches, the relative relationship among $\vec{P}_1, \vec{P}_2,$ and \vec{P}_3 are obtained beforehand. Suppose $\vec{P}'_1, \vec{P}'_2,$ and \vec{P}'_3 are the unit vectors along the projections of these three normal vectors onto the solar plane, and $\theta_1, \theta_2,$ and θ_3 are the angles subtended by \vec{u} and each of these three projected vectors, as illustrated in Fig. 17. Since the phase shift θ_p in (A3) is equal to θ up to a constant offset, the angles between these three projected vectors can be estimated by $(\theta_{p1}-\theta_{p2}), (\theta_{p2}-\theta_{p3}),$ and $(\theta_{p1}-\theta_{p3})$.

Assume we represent \vec{n} as a linear combination of $\vec{P}_1, \vec{P}_2,$ and \vec{P}_3 , i.e., $\vec{n} = a\vec{P}_1 + b\vec{P}_2 + c\vec{P}_3$. If we take the inner product of \vec{n} and \vec{P}_i , where $i = 1, 2, 3$, we obtain three equations to solve $a, b,$ and c as follows:

$$\begin{aligned} \langle \vec{P}_1, \vec{n} \rangle &= a + b \langle \vec{P}_1, \vec{P}_2 \rangle + c \langle \vec{P}_1, \vec{P}_3 \rangle = \sin(\varphi_1) \\ \langle \vec{P}_2, \vec{n} \rangle &= a \langle \vec{P}_1, \vec{P}_2 \rangle + b + c \langle \vec{P}_2, \vec{P}_3 \rangle = \sin(\varphi_2) \quad (\text{A4}) \\ \langle \vec{P}_3, \vec{n} \rangle &= a \langle \vec{P}_1, \vec{P}_3 \rangle + b \langle \vec{P}_2, \vec{P}_3 \rangle + c = \sin(\varphi_3). \end{aligned}$$

In (A4), the inner products $\langle \vec{P}_i, \vec{P}_j \rangle$, with $i, j = 1, 2, 3$, are known beforehand. To estimate $\{\varphi_1, \varphi_2, \varphi_3\}$, we formulate the vector \vec{P}'_i as $\vec{P}'_i = (\vec{P}_i - \sin \varphi_i \vec{n}) / \cos \varphi_i$. As we take the inner products among $\vec{P}'_1, \vec{P}'_2,$ and \vec{P}'_3 , we have

$$\begin{aligned} \langle \vec{P}'_1, \vec{P}'_2 \rangle &= \left(\langle \vec{P}_1, \vec{P}_2 \rangle - \sin \varphi_1 \sin \varphi_2 \right) / (\cos \varphi_1 \cos \varphi_2) \\ &= \cos(\theta_{p1} - \theta_{p2}) \\ \langle \vec{P}'_2, \vec{P}'_3 \rangle &= \left(\langle \vec{P}_2, \vec{P}_3 \rangle - \sin \varphi_2 \sin \varphi_3 \right) / (\cos \varphi_2 \cos \varphi_3) \quad (\text{A5}) \\ &= \cos(\theta_{p2} - \theta_{p3}) \\ \langle \vec{P}'_3, \vec{P}'_1 \rangle &= \left(\langle \vec{P}_3, \vec{P}_1 \rangle - \sin \varphi_3 \sin \varphi_1 \right) / (\cos \varphi_3 \cos \varphi_1) \\ &= \cos(\theta_{p3} - \theta_{p1}). \end{aligned}$$

Hence, with $\{(\theta_{p1}-\theta_{p2}), (\theta_{p2}-\theta_{p3}), (\theta_{p1}-\theta_{p3})\}$, the geometric direction of \vec{n} with respect to $\{\vec{P}_1, \vec{P}_2, \vec{P}_3\}$ can be deduced.

After the determination of \vec{n} , the choice of $\{\vec{u}, t_\theta\}$ is rather arbitrary. In our approach, we simply align \vec{u} with one of $\{\vec{P}'_1, \vec{P}'_2, \vec{P}'_3\}$ and the reference time t_θ is defined to be the time when the corresponding intensity profile has the maximum value.

APPENDIX II

IMAGE FORMATION MODEL

We assume that the surfaces in the 3-D scene of a parking lot are mainly Lambertian and the trichromatic RGB features at a pixel can be formulated as follows:

$$I_c = g \int_{\lambda} l(\lambda) r(\lambda) f_c(\lambda) d\lambda \quad (\text{A6})$$

where g is a geometric factor that depends on the included angle between the incident radiant flux and the normal vector of the corresponding surface, $l(\lambda)$ denotes the illuminant spectrum, $r(\lambda)$ represents the spectral reflectance function, and $f_c(\lambda)$ represents the filter sensitivity function of the c channel with $c \in \{R, G, B\}$. To discretize (A6) for computational analysis, several research works adopted finite-dimensional linear models to approximate both spectral reflectance function and illuminant spectrum [29], [32], [33]. In our approach, we adopted a 3-D linear model and (A6) can be reformulated as follows:

$$\begin{aligned} I_c &= g \int_{\lambda} \left(\sum_{i=1}^3 \beta_i l_i(\lambda) \right) \left(\sum_{j=1}^3 \alpha_j r_j(\lambda) \right) f_c(\lambda) d\lambda \\ &= g \sum_{i=1}^3 \beta_i \sum_{j=1}^3 \alpha_j \int_{\lambda} l_i(\lambda) r_j(\lambda) f_c(\lambda) d\lambda \quad (\text{A7}) \\ &= g \beta^T \mathbf{M}_c \alpha = g \beta^T \alpha_c \end{aligned}$$

where $\beta = (\beta_1, \beta_2, \beta_3)^T$ is the vector of illuminant coefficients, $\alpha = (\alpha_1, \alpha_2, \alpha_3)^T$ is the vector of reflectance coefficients, \mathbf{M}_c is a 3×3 matrix with its entries defined as follows:

$$\mathbf{M}_c(i, j) = \int_{\lambda} l_i(\lambda) r_j(\lambda) f_c(\lambda) d\lambda \quad (\text{A8})$$

and $\alpha_c = \mathbf{M}_c \alpha$. With (A8), the trichromatic color vector \mathbf{I}_{RGB} can be represented as follows:

$$\mathbf{I}_{\text{RGB}} = \begin{bmatrix} I_R \\ I_G \\ I_B \end{bmatrix} = g \begin{bmatrix} \alpha_R^T \\ \alpha_G^T \\ \alpha_B^T \end{bmatrix} \cdot \beta = g \mathbf{A} \beta \quad (\text{A9})$$

where $\mathbf{A} = [\alpha_R \ \alpha_G \ \alpha_B]^T$ is a 3×3 matrix.

In an outdoor parking lot, the lighting condition is varying over time. This makes both g and β change accordingly. To simplify the detection process, we focus mainly on the chromatic information. Since the absolute magnitude of α_c and β does not affect the chromatic information, we arbitrarily rescale \mathbf{A} and β by two constants a and b so that (A9) can be reformulated as follows:

$$\mathbf{I}_{\text{RGB}} = g \mathbf{A} \beta = gab \left(\frac{1}{a} \mathbf{A} \right) \left(\frac{1}{b} \beta \right) = (gab) \mathbf{Ri} = \|\mathbf{I}_{\text{RGB}}\| \mathbf{Ri} \quad (\text{A10})$$

APPENDIX III

TABLE OF VARIABLE NOTATIONS

	Notation	Meaning
1	H_L, I_L, S_L	Labeling layer, observation layer, and scene layer
2	M and N	Image dimensions
3	(m, n)	Pixel coordinates
4	$C, G,$	Car label and ground label
5	S, US	Shadowed label and unshadowed label
6	N_s	Number of parking spaces
7	$h^O(m, n)$	Object label at (m, n)
8	$h^L(m, n)$	Light label at (m, n)
9	l, w, h	Car length, car width, and car height
10	X_i, Y_i	Projected position of car center on the ground plane in the i th parking space
11	$C_i(m, n)$	Expected car labeling map at (m, n) given the i th parking space being occupied
12	$G_i(m, n)$	Expected ground labeling map at (m, n) given the i th parking space being occupied
13	$S_i(m, n)$	Expected shadow labeling map at (m, n) given the i th parking space being occupied
14	$US_i(m, n)$	Expected non-shadow labeling map at (m, n) given the i th parking space is occupied
15	ω_s	Angular frequency of solar movement
16	\vec{u}	A unit reference vector on the solar plane
17	\vec{n}	Normal vector of the solar plane
18	E_D	Classification energy
19	E_A	Adjacency energy
20	N_p	Neighborhood around (m, n)
21	\mathbf{I}_{RGB}	RGB color features of a pixel
22	$\mathbf{I}_{\text{RGB}}^{\text{N}}$	Normalized \mathbf{I}_{RGB}
23	\mathbf{R}	A 3×3 matrix depending on surface reflectance
24	\mathbf{i}	A vector depending on illumination
25	$g_{O,L}$	An intensity sample from the object type O under the illumination type L

26	$a_{O_1, O_2, L}$	The intensity ratio between the objects O_2 and O_1 under the illumination type L
27	$n_{O_1, O_2, L}$	A zero mean Gaussian noise that expresses the uncertainty in modeling the intensity ratio
28	$\overline{g_{O,L}}, \hat{\sigma}_{g_{O,L}}^2$	Sample mean and variance of the intensity training samples

ACKNOWLEDGMENT

The authors would like to thank the Associate Editor C. N. Taylor and the anonymous reviewers for their comments.

REFERENCES

- [1] M. Y. I. Idris, Y. Y. Leng, E. M. Tamil, N. M. Noor, and Z. Razak, "Car park system: A review of smart parking system and its technology," *Inform. Technol. J.*, vol. 8, no. 2, pp. 101–113, 2009.
- [2] H. Schneiderman and T. Kanade, "Object detection using the statistics of parts," *Int. J. Comput. Vision*, vol. 56, no. 3, pp. 151–177, Feb. 2004.
- [3] P. Viola and M. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, May 2004.
- [4] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
- [5] L.-W. Tsai, J.-W. Hsieh, and K.-C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 850–864, Mar. 2007.
- [6] C. H. Lee, M. G. Wen, C. C. Han, and D. C. Kou, "An automatic monitoring approach for unsupervised parking lots in outdoors," in *Proc. Int. Conf. Security Technol.*, 2005, pp. 271–274.
- [7] I. Masaki, "Machine-vision systems for intelligent transportation systems," in *Proc. IEEE Int. Conf. Intell. Transport. Syst.*, vol. 13, no. 6, Nov. 1998, pp. 24–31.
- [8] T. Horparasert, D. Harwood, and L. A. Davis, "A statistical approach for real-time robust background subtraction and shadow detection," in *Proc. IEEE Int. Conf. Comput. Vision*, Sep. 1999, pp. 1–19.
- [9] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Int. Conf. Comput. Vision Patt. Recog.*, vol. 2, Jun. 1999, pp. 246–252.
- [10] P. Power and J. Schoonees, "Understanding background mixture model for foreground segmentation," in *Proc. Image Vision Comput.*, 2002, pp. 267–271.
- [11] S. Funck, N. Mohler, and W. Oertel, "Determining car-park occupancy from single images," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2004, pp. 325–328.
- [12] C. C. Huang, S. J. Wang, Y. J. Chang, and T. Chen, "A Bayesian hierarchical detection framework for parking space detection," in *Proc. IEEE Int. Conf. Acou., Speech Signal Process.*, Apr. 2008, pp. 2097–2100.
- [13] D. B. L. Bong, K. C. Ting, and N. Rajae, "Car-park occupancy information system," in *Proc. 3rd Real-Time Technol. Applicat. Symp.*, Dec. 2006, pp. 65–70.
- [14] D. B. L. Bong, K. C. Ting, and K. C. Lai, "Integrated approach in the design of car-park occupancy information system," *IAENG Int. J. Comput. Sci.*, vol. 35, no. 1, pp. 7–14, 2008.
- [15] K. Yamada and M. Mizuno, "A vehicle parking detection method using image segmentation," *Electron. Commun.*, vol. 84, no. 10, pp. 25–34, Oct. 2001.
- [16] C. H. Lee, M. G. Wen, C. C. Han, and D. C. Kuo, "An automatic monitoring approach for unsupervised parking lots in outdoor," in *Proc. IEEE Int. Conf. Security Technol.*, Oct. 2005, pp. 271–274.
- [17] T. Fabian, "An algorithm for parking lot occupation detection," in *Proc. IEEE Comput. Inform. Syst. Indust. Manage. Applicat.*, Jun. 2008, pp. 165–170.
- [18] R. J. López-Sastre, P. G. Jimenez, F. J. Acevedo, and S. M. Bascon, "Computer algebra algorithms applied to computer vision in a parking management system," in *Proc. IEEE Int. Symp. Indust. Electron.*, Jun. 2007, pp. 1675–1680.

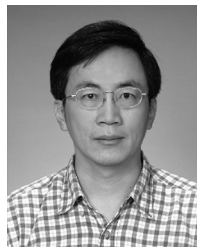
- [19] N. Dan, "Parking management system and method," U.S. Patent 20030144890A1, Jul. 2003.
- [20] Q. Wu, C. C. Huang, S. Y. Wang, W. C. Chiu, and T. H. Chen, "Robust parking space detection considering inter-space correlation," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 659–662.
- [21] J. K. Suhr, K. Bae, J. Kim, and H. G. Jung, "Free parking space detection using optical flow-based Euclidean 3-D," in *Proc. IAPR Conf. Mach. Vision Applicat.*, May 2007, pp. 563–566.
- [22] W. Yu and T. Chen, "Parking space detection from video by augmenting training dataset," in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 849–852.
- [23] Y. Boykov, O. Veksler, and R. Zabih, "Efficient approximate energy minimization via graph cuts," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
- [24] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
- [25] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
- [26] U.S. Naval Observatory. (2010). *Naval Oceanography Portal* [Online]. Available: <http://www.usno.navy.mil/USNO>
- [27] K. Sunkavalli, F. Romeiro, W. Matusik, Y. Zickler, and H. Pfister, "What do color changes reveal about an outdoor scene?" in *Proc. IEEE Int. Conf. Comput. Vision Patt. Recog.*, Jun. 2008, pp. 1–8.
- [28] Y. Tsin, R. Collins, V. Ramesh, and T. Kanade, "Bayesian color constancy for outdoor object recognition," in *Proc. IEEE Conf. Comput. Vision Patt. Recog.*, Dec. 2001, pp. 1132–1139.
- [29] T. Boykov, O. Veksler, and R. Zabih, "Markov random fields with efficient approximations," in *Proc. IEEE Int. Conf. Comput. Vision Patt. Recog.*, Jun. 1998, pp. 648–655.
- [30] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 2, no. 1, pp. 224–227, Apr. 1979.
- [31] C.-C. Huang. *Huang's Projects* [Online]. Available: <http://140.113.238.220/~chingchun/projects.html>
- [32] G. D. Finlayson, M. S. Drew, and B. V. Funt, "Color constancy: Generalized diagonal transforms suffice," *J. Optic. Soc. Am.*, vol. 11, no. 11, pp. 3011–3019, Nov. 1994.
- [33] D. Zmura and G. Iverson, "Color constancy: Basic theory of two stage linear recovery of spectral descriptions for lights and surfaces," *J. Optic. Soc. Am.*, vol. 10, no. 10, pp. 2148–2165, 1993.



Ching-Chun Huang received the B.S., M.S., and the Ph.D. degrees in electrical engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2000, 2002, and 2010, respectively.

His research interests are in image/video processing, computer vision, and computational photography.

Dr. Huang is a member of the IEEE Computer Society.



Sheng-Jyh Wang (M'95) received the B.S. degree in electronics engineering from National Chiao-Tung University (NCTU), Hsinchu, Taiwan, in 1984, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1990 and 1995, respectively.

He is currently a Professor with the Department of Electronics Engineering, NCTU. His research interests are in the areas of image processing, video processing, and image analysis.