Region-Level Motion-Based Background Modeling and Subtraction Using MRFs

Shih-Shinh Huang, *Student Member, IEEE*, Li-Chen Fu, *Fellow, IEEE*, and Pei-Yung Hsiao, *Member, IEEE*

***Abstract*—This paper presents a new approach to automatic segmentation of foreground objects from an image sequence by integrating techniques of background subtraction and motion-based foreground segmentation. First, a region-based motion segmentation algorithm is proposed to obtain a set of motion-coherence regions and the correspondence among regions at different time instants. Next, we formulate the classification problem as a graph labeling over a region adjacency graph based on Markov random fields (MRFs) statistical framework. A background model representing the background scene is built and then is used to model a *likelihood* energy. Besides the background model, a temporal coherence is also maintained by modeling it as the prior energy. On the other hand, color distributions of two neighboring regions are taken into consideration to impose spatial coherence. Then, the *a priori* energy of MRFs takes both spatial and temporal coherence into account to maintain the continuity of our segmentation. Finally, a labeling is obtained by maximizing the *a posteriori* energy of the MRFs. Under such formulation, we integrate two different kinds of techniques in an elegant way to make the foreground detection more accurate. Experimental results for several video sequences are provided to demonstrate the effectiveness of the proposed approach.**

***Index Terms*—Background subtraction, Markov random fields (MRFs), motion-based segmentation.**

I. Introduction

PROLIFERATION of cheap camera sensors and increased processing power have made acquisition and processing of the video information become feasible. Many analysis tasks such as object detection and tracking can be performed efficiently on standard PCs. In many applications, success of detecting foreground regions from a static background scene is an important step before high-level processing, such as object identification and event understanding. However, in real-world situations, there exist several kinds of environment variations that will make the foreground detection more difficult. In order to cope with that, the approach here should be able to immune to these variaitions, i.e., being invariant to them or adapting to them. The aforementioned variations that may cause the interested pixel intensity to change and, hence, lead to misdetection includes the following.

Manuscript received March 27, 2005; revised December 10, 2006. This work was supported by the National Science Council under the project NSC93-2752-E-002-007-PAE. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Anil Kokaram.

S.-S. Huang and L.-C. Fu are with the Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

P.-Y. Hsiao is with the Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, R.O.C.

Digital Object Identifier 10.1109/TIP.2007.894246

- **Illumination Variation**
  - **Gradual illumination variation** is the small change of light intensity due to the location change or fluctuation of light source.
  - **Sudden illumination variation** means the light intensity changes in an abrupt manner. For example, traffic light in an outdoor environment or lights switched on and off are cases of the sudden illumination variation.
  - **Shadow** is the area where direct light is entirely blocked (umbra) or partially blocked (penumbra).
- **Motion Variation**
  - **Global motion** means that small camera displacement will result in a global displacement of the captured scene. The causes of global motion include camera movement or a poorly fixed camera.
  - **Local motion** refers to the intrinsic motion in the scene due to movements of foreground or of nonstatic background objects, such as tree branches or cloud.

The aim of the proposed method in this paper is to handle all those variations except for global motion. In other words, our focus is on the stationary camera, that is, we assume that our input sequence has been properly compensated and the background scene is stationary.

A. Related Works

Generally speaking, techniques for foreground detection can be grouped into two categories, background subtraction and motion-based foreground segmentation. For background subtraction, the foreground represents regions that have different appearance from those in the reference image, which is normally referred to as background. For motion-based segmentation, regions that are subjected to a coherent and significant motion are considered meaningful foreground. We will give a brief review of these two kinds of techniques.

1) Background Subtraction: In order to adapt to changes, the background is usually represented by the background model and updated over time. This kind of technique is based on an assumption that the background scene is available and the camera is only subject to minimal vibration without loss of generality. Many approaches for background subtraction have been proposed over the past decades, but usually differ in the ways of modeling the background. A simple method is to represent the gray level or color intensity of each pixel in the image as an independent and unimodal distribution [1]–[4].

If the intensity of each pixel is due to the light reflected from a particular surface under a particular lighting, a unimodal distribution will be sufficient to model the pixel value. However,

in the real world, the appearance of a pixel in most video sequences is in multimodal distribution. The usage of a mixture of Gaussian distributions is common in modeling multimodel distribution. For example, Friedman [5] modeled the pixel intensity as a weighted mixture of three Gaussian distributions respectively corresponding to road, vehicle, and shadow. An incremental version of the expectation maximization (EM) algorithm was then used to learn and update the parameters of the Gaussian mixture. Stauffer [6] modeled each pixel as a K mixture of Gaussian distributions where K depends on memory. Incoming pixels are compared against the corresponding Gaussian mixture model. If a match is found, the parameters of the model are adjusted. An improved method [7] was proposed to learn faster and more accurately by introducing online EM algorithm.

However, not all distributions are in Gaussian form [8]. In [9], a nonparametric background model based on nonparametric density estimation was proposed to handle the situations where the background scene is nonstatic but contains minimal motion. Another approach [10] that represents the color of each pixel by a group of clusters was proposed to adapt to noise and background variation. The currently proposed approaches are used to represent the background scene by a set of independent models without taking any semantic information into consideration. This makes false detection likely when changes or noise occur. It is here where some sophisticated modeling or updating strategies are applied.

2) Motion-Based Foreground Segmentation: The technique of motion-based foreground segmentation is based on the idea that appearance of foreground objects are always accompanied by motion. In general, such technique consists of two steps, i.e., motion segmentation and region classification. The aim of motion segmentation is to divide an image into a set of regions with motion coherence, whereas that of region classification is to assign a label, foreground or background, to each segmented region.

Various approaches in the literature for motion segmentation have been proposed. For providing a meaningfully semantic description of video, Wang and Adelson [11] employed a k-means clustering algorithm in the affine parameter space to find a small number of motion classes. Finally, each flow vector is assigned to one of the resulting motion classes. Borshukov [12] later improved Wang and Adelson's algorithm through a merging and multistage approach to perform motion segmentation in a more robust way.

The aforementioned approaches incur inaccurate segmentation due to inexact motion estimation near the object boundary. In order to overcome this problem, color information is introduced to obtain more accurate segmentation. In [13]–[16], an initial segmentation proceeds with color segmentation. Then, regions are merged on the basis of temporal or spatial similarity. Underlying approaches of this kind are based on an assumption that motion boundaries are generally subsets of colored ones. Without prior knowledge on the foreground, a straightforward way is to consider the foreground as a segmented region with large motion velocity. In addition to motion, Tsaig [13] also adopted spatial and temporal continuity to perform region classification by maximizing *a posteriori* probability under MRFs framework.

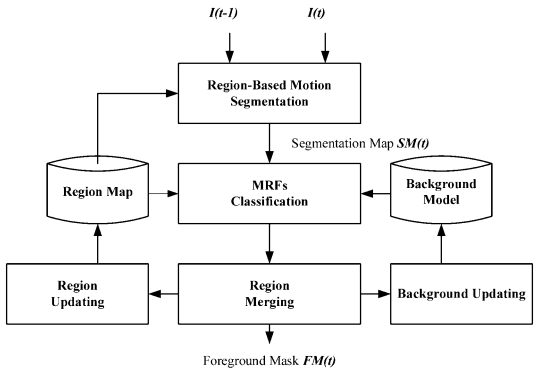

Fig. 1. Block diagram of the proposed algorithm.

B. System Overview

In this section, we present the main features of our approach for segmenting foreground regions from a sequence of images. It extends the work in [17] and [18], and Fig. 1 gives a block diagram of our proposed algorithm.

The main idea is to regard the background model as a portion of knowledge for classification, and motion-based segmentation is to generate a set of regions for classification in the semantic level. At first, a region-based motion segmentation algorithm based on information of both motion and color is applied to segment captured images into a set of regions. All pixels belonging to the same region have coherent motion. In order to save time, the segmentation result at a preceding time instant is used to facilitate the segmentation process and to build the correspondent mappings for regions at the next time instant.

After segmentation, the statistical framework, MRFs, is introduced to formulate the foreground detection problem as a labeling problem. By comparing the segmented region with the one built in the background model, a *likelihood* energy can be evaluated for classification. For the sake of maintaining spatial and temporal coherence, the similarity at boundaries of all neighborhood regions and the relation among all possibly corresponding regions at different time instants are taken into account to model the *a priori* energy.

The optimization over the MRFs model is then performed, or specifically *a posteriori* probability is maximized to obtain a classification result. Finally, regions which have the same classification label and similar colors are merged to derive a more meaningful segmentation. Finally, the background model and the resulting region map are updated accordingly.

C. Organization

The remainder of this paper is organized as follows. In Section II, we introduce the region-based motion segmentation algorithm to obtain a set of motion-coherence regions. Section III addresses problems of background modeling and updating. The classification process based on the MRFs statistical framework is described in Section IV. In Section V, we demonstrate the effectiveness of the developed approach by providing some appealing experimental results. Finally, we conclude the paper in Section VI with some relevant discussion.

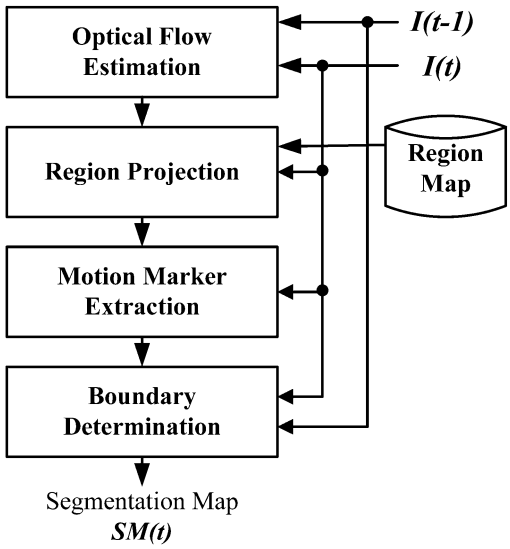

Fig. 2. Block diagram of the region-based motion segmentation algorithm.

II. Region-Based Motion Segmentation

The aim of the motion segmentation in this paper is to divide the entire image into a set of regions of which each associated with object(s) or part(s) that has (have) coherent motion. In general, the aforementioned approaches to motion segmentation involve color segmentation followed by the region fusion algorithm based on motion information. The approaches based on region-merging strategy usually result in over-segmentation, which makes region classification more difficult and computationally expensive. In this paper, we propose a region-based motion segmentation algorithm by using motion information followed by color information to overcome this shortcoming.

As shown in Fig. 2, the proposed algorithm mainly consists of three steps, namely, region projection, motion marker extraction, and boundary determination. First of all, Horn and Schunck's method [19] is used to estimate dense optical flow for describing motion vector, $(u(x,y), v(x,y))$, of every pixel (x,y) between two consecutive image frames $I(t-1)$ and $I(t)$. Segmented regions of the previous image frame, $I(t-1)$, will then be projected to the current image frame, $I(t)$. Regions with coherent motion are extracted as initial motion markers. Pixels not ascribed to any region are labeled uncertain ones. Finally, a watershed algorithm [20] based on motion and color is utilized to join uncertain pixels to the nearest similar marker.

A. Region Projection

The purpose of region projection, namely, projecting regions in the previous frame to the current one, is to facilitate the segmentation. Because of inaccuracy in estimating motion of region's boundary using Horn and Schunck's method, a parametric motion model is adopted to represent the motion of a region. For saving computation time, the affine motion model is chosen for adoption in this paper. Let the affine motion model A_i represent the motion of a region R_i, then it is a six-parameter model denoted as the parametric motion vector, i.e., $A_i(x,y;R_i)$, of any relevant pixel $(x,y) \in R_i$. Particularly, $A_i(x,y;R_i) = (U_i(x,y), V_i(x,y))$ can be expressed as

$$\begin{cases} U_i(x,y) = a_{i(1)} + a_{i(2)}x + a_{i(3)}y \\ V_i(x,y) = a_{i(2)} + a_{i(5)}y + a_{i(6)}y \end{cases} \quad (1)$$

where $(U_i(x,y), V_i(x,y))$ is referred to as the parametric motion vector of the pixel (x,y) with which $a_{i(1)}, a_{i(2)}, \ldots, a_{i(6)}$ being the six parameters of A_i.

The six parameters of A_i can be estimated by the least square method [21] as shown in (2)

$$\begin{bmatrix} a_{i(1)} & a_{i(4)} \\ a_{i(2)} & a_{i(5)} \\ a_{i(3)} & a_{i(6)} \end{bmatrix} = \left[\sum_{(x,y)\in R_i} [1,x,y]^T [1,x,y] \right]^{-1} \times \sum_{(x,y)\in R_i} [1,x,y]^T [u(x,y), v(x,y)] \quad (2)$$

where R_i denotes the region over which the parametric motion of every pixel is described by A_i.

Given the affine motion model, any pixel $(x,y) \in R_i$ in the previous frame, $I(t-1)$, should be projected to the location (x',y'), where $(x',y') = (x+U_i(x,y), y+V_i(x,y))$. The motion estimate of each pixel already has subpixel accuracy. However, due to occlusion and uncovering effect, the displaced frame difference is less likely to be useful especially at the boundaries of objects. To reduce this effect, the minimum displaced frame difference over the nearest four nearest pixels is taken as the error measure which is called projection error $e_p(x,y)$ and is defined as

$$e_p(x,y) = \min_{(i,j)\in N_4(x',y')\cup(x,y)} |I(x,y;t-1) - I(i,j;t)| \quad (3)$$

where $N_4(x',y')$ denotes the abbreviation of the set of four connected pixels surrounding the pixel (x',y'). $|I(x,y;t-1) - I(i,j;t)|$ is the Euclidean distance of RGB color vectors between two pixels (x,y) and (i,j) at different time instant. If $e_p(x,y)$ is less than a given threshold Th_p, then the region label of (x',y') is assigned to the same one of (x,y). Otherwise, the pixel which has large projection error is labeled as uncertain ones to indicate that the projecting from the previous frame to the current one is failed.

B. Motion Marker Extraction

The output of this step is a set of motion-coherent regions, that is, all pixels within a region comply with a motion model. Here, each such region is referred to as a motion maker. By starting to grow from these markers, we can eventually obtain a segmentation. In the next section, we will describe how to classify every uncertain pixel to only one of motion markers by region growing scheme.

Motion markers here are derived in two ways. First, the regions projected from the previous time frame are one kind of motion markers because each of them arises from an affine motion model. In addition to those, the regions resulting from the newly introduced object(s) may be another kind of motion

markers, for example, the trunk of a person who appears in the background scene. To handle this situation, a method similar to [15] is used to extract this kind of motion marker. That is, a k-means clustering algorithm [22] is applied to perform color quantization in RGB color space followed by connected-component finding algorithm [23] so as to extract a set of homogeneous color regions from uncertain pixels. Currently, the number of quantized colors used in this paper is 12.

Next, an affine motion model A_i is evaluated to describe the motion of each region, R_i, according to (2). We then exclude the pixel (x, y) from R_i if the motion error, $e_m(x, y)$, of the pixel (x, y) associated with A_i is larger than a predefined threshold Th_m, where the motion error is defined as

$$e_m(x, y) = |(u(x, y), v(x, y)) - A_i(x, y; R_i)|. \quad (4)$$

After exclusion, the regions that have the region size above a threshold are considered as the motion markers. The set of these motion markers is denoted as $\mathcal{M} = \{\mathcal{M}_i | i = 1, 2, \ldots m\}$, where m is the number of motion markers. Each motion marker, $\mathcal{M}_i$ stands for a segmented region.

C. Boundary Determination

After motion marker extraction, the number of the regions to be segmented is known. However, a large number of pixels are not yet assigned to any region. These uncertain pixels are mainly around the contours of the regions. Through the use of the watershed algorithm [20], uncertain pixels will be merged to one of the markers. The watershed algorithm is basically a region growing algorithm that merges the uncertain pixel to the nearest similar marker. The remaining problem is how to define the similarity measure between a uncertain pixel and the motion marker.

In general, the weighted sum of the intensity and motion compensation difference is a popular measure for estimating the distance between a pixel and a region in the literature of spatio-temporal segmentation [24], [14], [15]. Here, this measure is also adopted for the watershed algorithm. Suppose that A_i is the affine motion model for the motion marker, $\mathcal{M}_i$, and (x, y) is a uncertain pixel neighboring to $\mathcal{M}_i$. The distance between $\mathcal{M}_i$ and (x, y) is defined as

$$d(\mathcal{M}_i, (x, y)) = \alpha_1 d_c(\mathcal{M}_i, (x, y)) + (1 - \alpha_1) d_m(\mathcal{M}_i, (x, y)) \quad (5)$$

where α_1 is a weighting factor. $d_c(\cdot, \cdot)$ and $d_m(\cdot, \cdot)$ are intensity difference and displaced frame difference, respectively. Because the motion marker $\mathcal{M}_i$ is just a motion-coherent region rather than an intensity-coherent one, the intensity distance used in [15] may not be a suitable one. In this paper, $d_c(\mathcal{M}_i, (x, y))$ is defined as

$$d_c(\mathcal{M}_i, (x, y)) = \min_{(i,j) \in N_4(x,y) \cap \mathcal{M}_i} |I(x, y; t) - I(i, j; t)|. \quad (6)$$

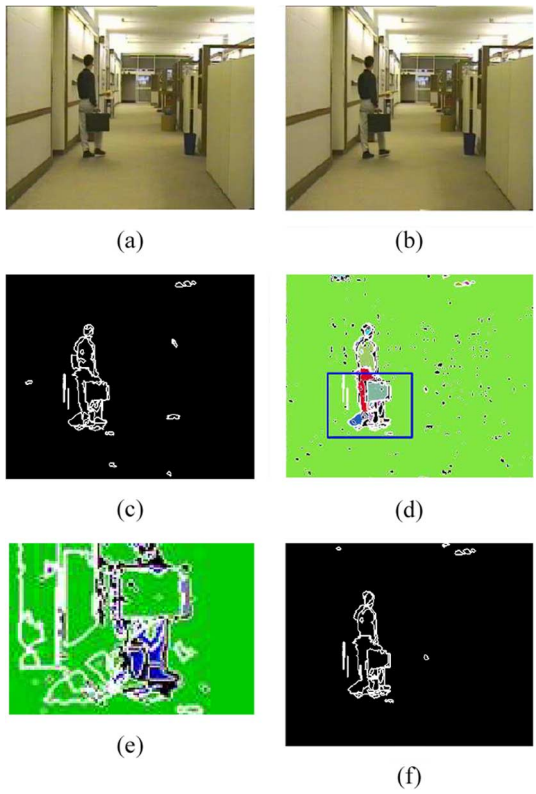

Fig. 3. Frame (a) 38 and (b) 39, respectively; (c) segmentation result stored in the **Region Map**. The regions given with different colors except the black one in (d) are motion markers projected from the region in (c) according to the motion vector represented by the affine motion model. Then, the area inside the blue rectangle in (d) is zoomed in and shown in (e) for better exhibition. Blue regions in (e) are motion markers obtained by using color information. After boundary determination process, a final segmentation result of region-based motion segmentation is shown in (f).

As for displaced frame difference, it is the intensity difference of two corresponding pixels at different frames. Thus, the displaced frame difference, $d_m(\mathcal{M}, (x, y))$ can be defined as

$$d_m(\mathcal{M}_i, (x, y)) = |I(x, y; t) - I((x, y) - A_i(x, y; \mathcal{M}_i); t - 1)|. \quad (7)$$

Fig. 3 shows the result of our proposed region-based motion segmentation algorithm applied to the image sequence for hall monitoring. As shown in Fig. 3(f), the segmentation result of frame 39 exhibits that the background behind the walking person is segmented as a single region by our approach. This makes the further region classification more efficient and effective.

III. Background Modeling

In the last decade, many sophisticated methods have been proposed to model the background scene. To deal with multiple appearances of the background, Stauffer and Grimson [6] modeled each background pixel by a mixture of K Gaussian distributions, and the details of its robustness were explained in their paper. In this paper, we use the same way to model and update the background scene. A brief description of Stauffer and Grimson's work is first given and then we introduce the Bhattacharyya distance as the difference measure between the region

from the region-based motion segmentation and the one represented by the background model.

A. Adaptive Gaussian Mixture Models

The probability of observing $o(t)$ of a specific pixel at time instant t can be expressed as

$$P(o(t)) = \sum_{i=1}^{K} w_i(t)\eta(o(t); \mu_i(t), \Sigma_i(t)) \tag{8}$$

where $w_i(t)$ is the weight of the ith Gaussian distribution at time t, $\mu_i(t)$ and $\Sigma_i(t)$ are the mean vector and covariance matrix of the ith Gaussian distribution at time t, and $\eta(o; \mu, \Sigma)$ is the normal Gaussian distribution expressed by

$$\eta(o; \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}}|\Sigma|^{\frac{1}{2}}} \times \exp\left\{-\frac{1}{2}(o-\mu)^T|\Sigma|^{-1}(o-\mu)\right\}. \tag{9}$$

Note that the observation variable, o, we use in this paper is an RGB color vector.

To adapt to illumination changes, the weight w_k of the kth Gaussian distribution is updated by the following equation:

$$w_k(t) = (1-\alpha_2)w_k(t-1) + \alpha_2 M_k(t) \tag{10}$$

where α_2 is the learning rate and $M_k(t)$ is the matching factor. In matching process, the K Gaussian distributions are first sorted by the value $w/|\Sigma|^{1/2}$. $M_k(t)$ equals 1 if kth Gaussian model is the first one which matches the observed color value, and $M_k(t)$ is set to 0 for the remaining models. A matching is defined as the observed color which falls within the range of 2.5 time standard deviation of the Gaussian model. The parameters of the distribution which matches the observation are updated by the following equation:

$$\begin{aligned}\mu(t) &= (1-\alpha_2)\mu(t-1) + \alpha_2 o(t)\\ \Sigma(t) &= (1-\alpha_2)\Sigma(t-1)\\ &\quad + \alpha_2\{(o(t)-\mu(t))(o(t)-\mu(t))^T\}.\end{aligned} \tag{11}$$

As discussed in [7], the likelihood term ρ in [6] is ignored to make the adaptation of the means and covariance matrices faster. If none of the K distributions matches the new observation, the least probable distribution is replaced by a newly created distribution with the current value as its mean, an initially high variance, and a low weight parameter. Initially, the background model is constructed by taking the first image as the reference one through the update process. In other words, a Gaussian distribution of each pixel is initialized by setting its color mean to the color of the corresponding pixel at the first frame, its color variance to an initially high value, and its weight to 1.0.

B. Bhattacharyya Distance

After describing the way to model the background scene and update it, we want to introduce how to measure the similarity between the currently observed image I and the constructed background model. Inspired by [25], the idea behind is to generate an image I_b representing the background scene according to the constructed background model, and, thus, the measure to evaluate the similarity between the current observation and the background model can be defined as a function of I and I_b in region level.

For each pixel observation $o(x, y)$ in the image I, the corresponding pixel observation $o_b(x, y)$ in the image I_b is defined as the mean vector of the Gaussian distribution at the pixel (x, y) which has the minimum Mahalanobis distance [22] from $o(x, y)$. Let R_s be the region at the image I obtained from the region-based motion segmentation process, and R_b be the same region as R_s but at the image I_b.

We assume that the colors of the regions, R_s and R_b, are both Gaussian distributions. Suppose that μ_s and Σ_s are the mean vector and covariance matrix of R_s, respectively, and similarly for μ_b and Σ_b are of R_b. The distance measure between R_s and R_b can be related to the probability of classification error in statistical hypothesis testing, which naturally leads to the Bhattacharyya distance [22], [26]. The Bhattacharyya distance, $d_{\text{bhat}}(\cdot, \cdot)$, is formally defined as follows:

$$\begin{aligned}&d_{\text{bhat}}(R_s, R_b)\\ &\quad= \frac{1}{8}(\mu_s-\mu_b)^T\left|\frac{\Sigma_s+\Sigma_b}{2}\right|^{-1}(\mu_s-\mu_b)\\ &\qquad+ \frac{1}{2}\ln\frac{\left|\frac{\Sigma_s+\Sigma_b}{2}\right|}{\sqrt{|\Sigma_s||\Sigma_b|}}.\end{aligned} \tag{12}$$

The first term of (12) gives the class separability due to the difference between class means, whereas the second term is the class separability due to the difference between class covariance matrices.

However, the region similarity defined in this way will lead to misclassification of the background region where direct light is blocked by the foreground object. The region of this kind is referred to as shadow. According to [8], the intensity of the pixel in shadow will be scaled down by a factor λ with $\lambda_f \le \lambda \le 1$, where λ_f is a constant.

If the actual color vector of a pixel is $v = (r, g, b)$, it will become $v' = (r', g', b')$ after being covered by shadow. In an ideal case, $v' = \lambda v$. Due to light fluctuation and noise effect, the ideal situation hardly takes place. Thus, the scaling factor is defined to be λ^*, which minimizes $f(\lambda) = (r' - \lambda r)^2 + (g' - \lambda g)^2 + (b' - \lambda b)^2$. By differentiating $f(\lambda)$ with respect to λ, we can obtain λ^* according to the following equations:

$$\begin{aligned}\frac{df(\lambda^*)}{d\lambda} &= 0\\ \Rightarrow \lambda^*(r^2+g^2+b^2) &= rr'+gg'+bb'\\ \Rightarrow \lambda^* &= \frac{rr'+gg'+bb'}{r^2+g^2+b^2}.\end{aligned} \tag{13}$$

In order to create invariance to the shadow effect, the pixel in the current image should be preprocessed first, but this is impractical due to expensive computation. Instead of doing this, we just use the mean vectors of R_s and R_b to evaluate λ^* and scale down the distribution (μ_s, Σ_s) of R_s to $(\lambda^*\mu_s, (\lambda^*)^2\Sigma_s)$. Fig. 4 shows the results of our proposed shadow elimination algorithm.

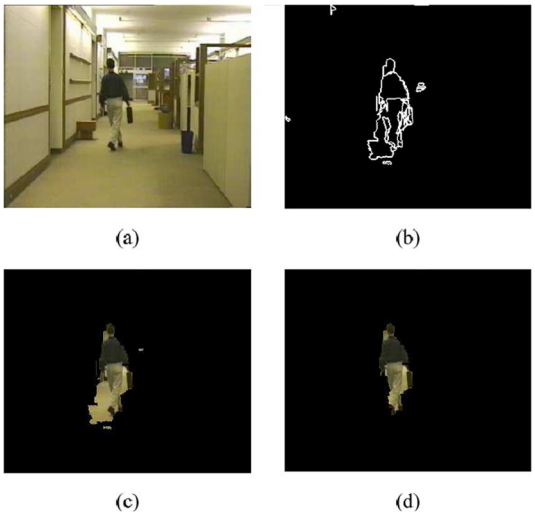

Fig. 4. Shadow effect elimination is (a) the original image and (b) is the segmentation result of our proposed region-based motion segmentation algorithm. The detected foreground without taking shadow effect into consideration, that is, $\lambda^* = 1$ is shown in (c); (d) result after applying shadow effect elimination.

IV. MRFS-Based Classification

In this section, we describe how to incorporate the background model to classify every region in the segmentation map SM into either a foreground object or a background one by MRFs. The regions obtained from the aforementioned motion segmentation algorithm are semantic ones where the pixel movements are consistent with a motion model due to the object's motion. Therefore, all pixels within each region should be classified to the same label, and, thus, we perform the label assignment in region level by MRFs. Before that, a graph called region adjacency graph (RAG) is used to represent the set of segmented regions. Let $G = (S, E)$ be an RAG, where $S = \{s_1, s_2, \ldots, s_n\}$ is the set of nodes in graph and each node s_i corresponds to a region R_i, and E is the set of edges with $(s_i, s_j) \in E$ if R_i and R_j being neighboring regions.

A. MRFs Framework

Then, we want to find an assignment of all sites (nodes) $s_i \in S$ to either background (B) or foreground (F). An assignment is called a configuration and is denoted as ω, and the set of all possible configurations is denoted as Ω.

Suppose that we obtain an observation O. The optimal configuration $\tilde{\omega}$ which we are interested is the one that will be a maximum *a posteriori* probability (MAP) under the observation O. That is

$$\tilde{\omega} = \arg\max_{\omega} P(\omega \,|\, O). \tag{14}$$

From Bayes' rule, we have

$$\tilde{\omega} = \arg\max_{\omega} \frac{P(O \,|\, \omega)P(\omega)}{P(O)}. \tag{15}$$

According to Hammersley–Clifford theorem [27], the *a posteriori* probability in (14) which follows a Gibbs distribution can be expressed as

$$P(\omega|O) = \frac{e^{-U(\omega|O)}}{Z} \tag{16}$$

where Z is a normalizing constant, called partition function. To maximize the *a posteriori* probability leads to minimize the posterior energy function $U(\omega \,|\, O) = U(O \,|\, \omega) + U(\omega)$, where the terms, $U(O|\omega)$ and $U(\omega)$ are the *likelihood* and the *a priori* energy over all sites, respectively.

For practical reasons, only singleton and pairwise cliques are considered. Therefore, the *posterior* energy function $U(\omega|O)$ can be decomposed into

$$\begin{aligned} U(\omega|O) = &\sum_{s_i \in S} U(o_i \,|\, s_i = \omega_i) \\ &+ \sum_{s_i \in S} U_1(s_i = \omega_i) \\ &+ \sum_{(s_i, s_j) \in E} U_2(s_i = \omega_i, s_j = \omega_j) \end{aligned} \tag{17}$$

where o_i is the observation of the site s_i, and, $U_1(\cdot)$ and $U_2(\cdot, \cdot)$ are the singleton and pairwise clique energies, respectively. In the subsequent sections, we will give the details about the definitions of the *likelihood* and the *prior* energies.

B. Region Classification

In this section, we describe how to define $U(O|\omega)$ and $U(\omega)$ so as to incorporate the background model as well as temporal and spatial coherence under MRFs framework.

1) Likelihood Energy $U(o_i|s_i = \omega_i)$: The term $U(o_i|s_i = \omega_i)$ represents the *likelihood* energy of the site s_i to be classified as the label ω_i. As a result, two energy functions, $U(o_i|\omega_i = B)$ and $U(o_i|\omega_i = F)$, should be defined to evaluate the background and foreground likelihood energies, respectively. The idea behind our design of the *likelihood* energy is described as follows. If the color distribution of the currently observed image I at site s_i is similar to the one of the background scene I_b represented by the background model, the site s_i will have high probability to be assigned to a label B; otherwise, it will be more likely classified as foreground.

Let R_{i_s} and R_{i_b} be the two regions of the site s_i but at the image I and I_b, respectively. The similarity measure between the color distributions of the regions, R_{i_s} and R_{i_b}, is through Bhattacharyya distance $d_{\text{bhat}}(R_{i_s}, R_{i_b})$. Thus, two functions, $f^F_{\text{likelihood}}(\cdot)$ and $f^B_{\text{likelihood}}(\cdot)$, as depicted in Fig. 5(a) are defined to evaluate $U(o_i|\omega_i = F)$ and $U(o_i|\omega_i = B)$, respectively. $U_{\text{likelihood}}$ and $\text{Th}_{\text{likelihood}}$ in Fig. 5(a) are two constants. The purpose of introducing the threshold $\text{Th}_{\text{likelihood}}$ in defining the energy functions is to avoid the outlier effect [28].

2) Prior Energy $U(\omega)$: The *prior* energy is composed of singleton, $U_1(\cdot)$, and pairwise, $U_2(\cdot, \cdot)$ energies. We relate $U_1(\cdot)$ to the temporal coherence, that is, the region obtained at current time instant tends to be classified as the same label as the corresponding region at previous time instant. Suppose that R^i stands for the region of the site s_i. The $U_1(\omega_i)$ can be defined as

$$U_1(\omega_i) = \begin{cases} -r_B d_{\text{bhat}}(R_i(t), R_i(t-1)), & \text{if } \omega_i = B \\ -r_F d_{\text{bhat}}(R_i(t), R_i(t-1)), & \text{if } \omega_i = F \end{cases} \tag{18}$$

where $R_i(t-1)$ is the corresponding region of $R_i(t)$ at frame $I(t-1)$ which can be obtained by using affine motion model (see

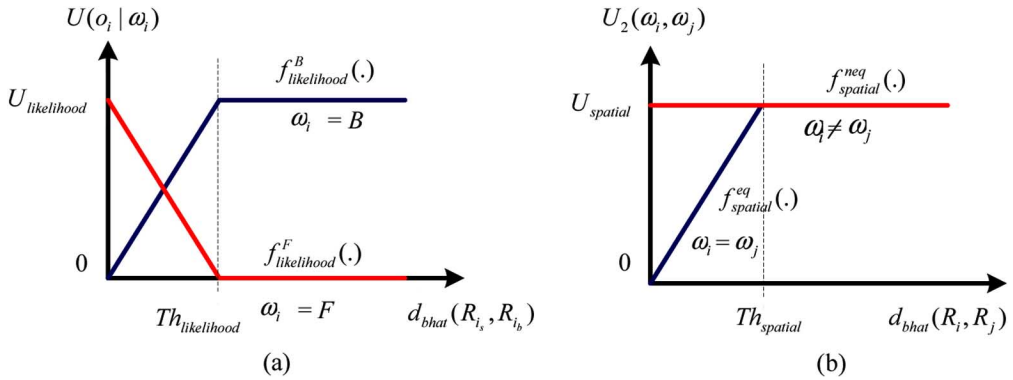

Fig. 5. Energy functions: (a) Two functions, $f^{B}_{\text{likelihood}}(\cdot)$ and $f^{F}_{\text{likelihood}}(\cdot)$, to evaluate the *likelihood* function, $U(o_i \mid \omega_i)$. (b) Two functions, $f^{\text{neq}}_{\text{spatail}}(\cdot,\cdot)$ and $f^{\text{eq}}_{\text{spatial}}(\cdot,\cdot)$, that are for evaluating spatial function $U_2(\cdot,\cdot)$.

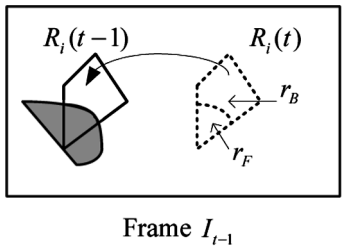

Fig. 6. Temporal projection of the region $R_i(t)$ at time t to the one $R_i(t-1)$ at time $t-1$. The shaded area is the region that has been classified as the foreground. r_B is the ratio of the pixels within $R_i(t-1)$ that have been assigned a background label B.

TABLE I
PARAMETERS TABLE

Th_p	Th_m	α_1	α_2	λ_f
8	0.3	0.5	0.02	0.7
$U_{likelihood}$	$Th_{likelihood}$	$U_{spatial}$	$Th_{spatial}$	
1.0	2.0	1.0	4.0	

Fig. 6). r_B is the ratio of the pixels in $R_i(t-1)$ that have been classified as background at time instant $t-1$, and $r_F = 1 - r_B$.

As for the term, $U_2(\cdot,\cdot)$, we relate it to spatial coherence. Spatial coherence means that two neighboring regions with similar color should be assigned to the same label. Assume that two sites s_i and s_j are adjacent. Then, s_i and s_j will be assigned to the same label if they have similar color distributions. According to this, two functions, $f^{\text{neq}}_{\text{spatial}}(\cdot)$ and $f^{\text{eq}}_{\text{spatial}}(\cdot)$, as depicted in Fig. 5(b) are defined for the cases, $\omega_i \neq \omega_j$ and $\omega_i = \omega_j$, respectively. Similarly for $U_{\text{likelihood}}$ and $\text{Th}_{\text{likelihood}}, U_{\text{spatial}}$ and $\text{Th}_{\text{spatial}}$ are two predefined constants in Fig. 5(b).

By such formulation, the foreground detection problem is mapped to optimization problem. The optimization is carried out by using iterative conditional mode (ICM) algorithm to find the most proper label assignment of every region. After classification, the regions neighboring to each other and with the same classified label and motion will then be merged and used to update the background model and region map.

V. EXPERIMENT

In this section, one standard MPEG-4 test sequence as well as three image sequences captured from intelligent home (e-home) demon room belonging to the Intelligent Robotics Laboratory at National Taiwan University are considered to validate our proposed method. The MPEG-4 test sequence we used here is the *Hall Monitoring* image sequence in CIF format at 10 fps, whereas the three e-home sequences with the image size 352×240 are all acquired via AXIS 2310 PTZ Network Camera, which is mounted on the ceiling for monitoring and perceiving human activity. Additionally, we compare our algorithm with the one proposed in [29], [30] which is used to extract foreground objects for further human identification. The threshold value $f(a,b)$ of Wang's algorithm is set to 0.8. Parameters used for foreground detection in our system are listed in Table I. In summary, Th_p and Th_m are thresholds for projection error and motion error, respectively. α_1 is a weighting factor of $d(\cdot)$ in (5). α_2 in (10) is factor for updating the weights of the Gaussain distributions. λ_f is a constant for shadow effect elimination. $U_{\text{likelihood}}, \text{Th}_{\text{likelihood}}, U_{\text{spatial}}$, and $\text{Th}_{\text{spatial}}$ are four constants for region classification as illustrated in Fig. 5.

A. Experimental Result

The *Hall Monitoring* benchmark is a commonly used MPEG-4 test sequence, especially for evaluating the effectiveness of background subtraction techniques. To effectively subtract foreground objects from the *Hall Monitoring* image sequence, the problem of noise and shadow effect caused by indoor illumination should be well handled. Fig. 7(a) illustrates the original frames 15, 25, 50, and 75 of the *Hall Monitoring* image sequence. Images in Fig. 7(b) and (c) shows the detection results of Wang's approach and ours, respectively. Frame 15 here is to exhibit that our algorithm can automatically detect newly introduced objects. The red circles in Fig. 7(b) indicate that noise due to light fluctuation is still considered as the foreground even when sophisticated measure in Wang's approach is adopted to perform background subtraction. However, by performing foreground detection in a more semantic region level, we can eliminate the noise effect as shown in Fig. 7(c).

In addition to *hall monitoring* sequence, three video sequences from our e-home demo room are used to demonstrate that our proposed approach can handle illumination variation and local motion. The first case is the image sequence exhibiting the gradual illumination variation and local motion. When a

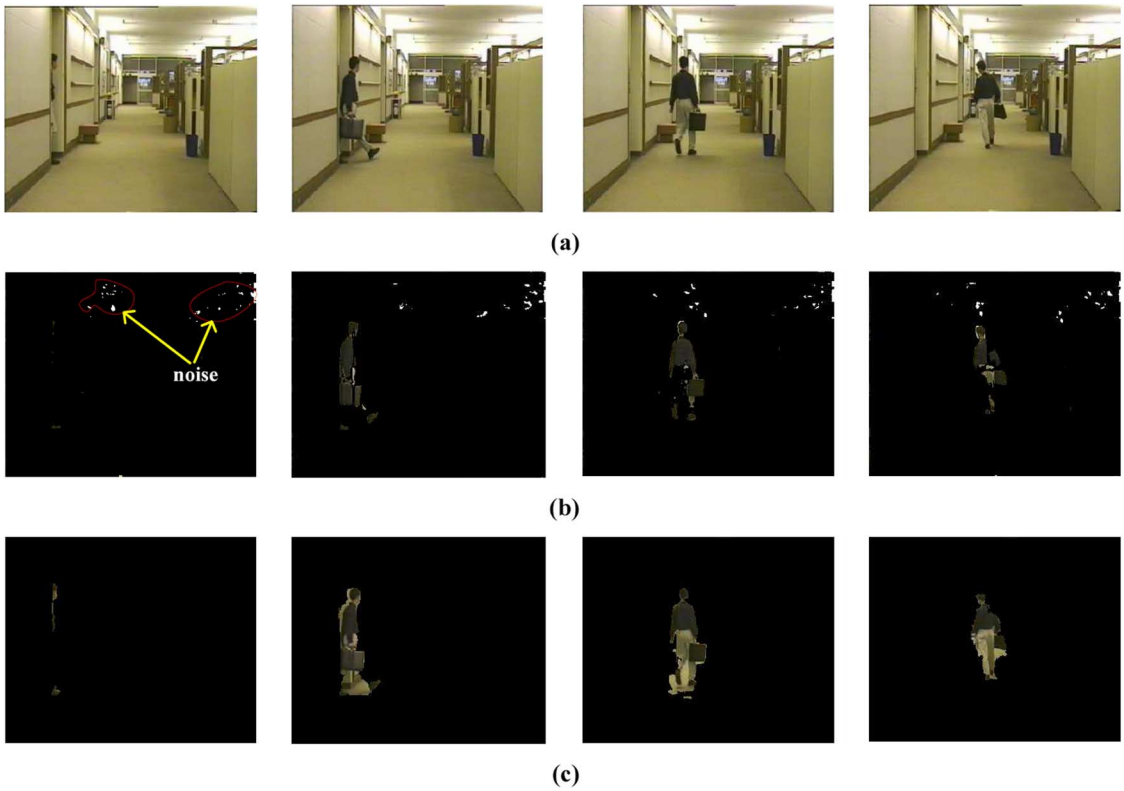

Fig. 7. *Hall monitoring* sequence: (a) 15, 25, 50, and 75 of the *hall monitoring* sequence. (b), (c) Detection result of Wang's approach and our proposed approach, respectively.

person enters, the background will gradually brighten. This is due to radiance from the fluorescent lamp that is reflected back into the background scene. After leaving the scene, he will wave the curtains to make it flutter. Some possible false positives due to Wang's algorithm under the condition with gradual illumination variation and local motion are shown in Fig. 8(b). As shown in Fig. 8(c), the detection results of our approach are more robust in such situations.

In the second case, the situation with the sudden illumination variation is used to test the effectiveness of our approach. The light in the living room of our demo e-home will be switched off automatically to save energy when the last occupant leaves that room. The original images of this case are shown in Fig. 9(a) and the two bottom-most pictures are for two scenes where the light in the living room is all turned off. The two bottom-most pictures of the detection results in Fig. 9(b) show that Wang's approach may lead to some misclassification when there is a sudden illumination change situation. This is expected because the intensity of an image is the only cue used in Wang's approach for pixel classification. Although our work cannot deal perfectly with this situation as shown in Fig. 9(c), at least most pixels of the background scene are correctly classified. The reason that our approach is able to deal so well with sudden illumination variation is because both spatial and temporal coherence are imposed. When illumination variation occurs, our motion segmentation algorithm will result in several newly segmented regions. If these regions are considered independently and only the background model is used for region classification, all of them will be classified as foreground ones. Yet, classifying them as the background will yield lower posterior energy due to integrate the *prior* energies that impose spatial and temporal coherence.

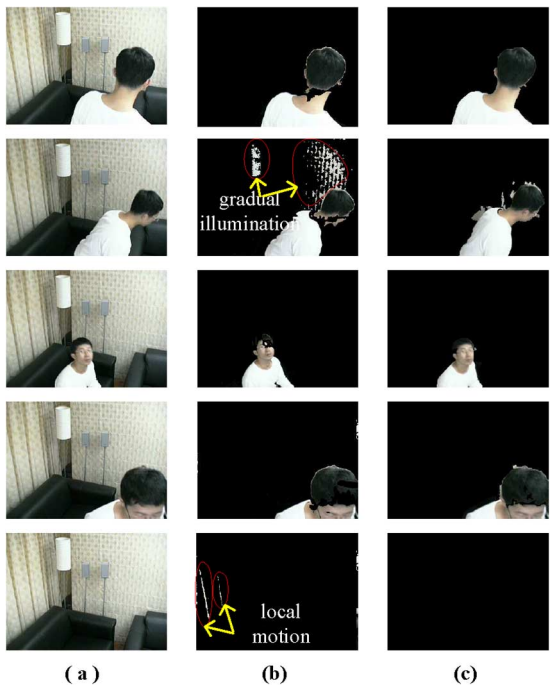

Fig. 8. Gradual illumination variation in e-home demo room. (a) Original images. (b) Detection result of Wang's approach. (c) Detection results of our approach.

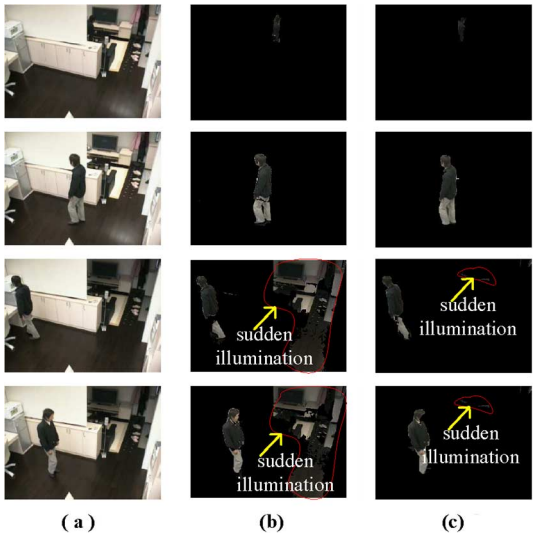

Fig. 9. Sudden illumination variation in e-home demo room. (a) Original images. (b) Detection results of Wang's approach. (c) Detection results of our approach.

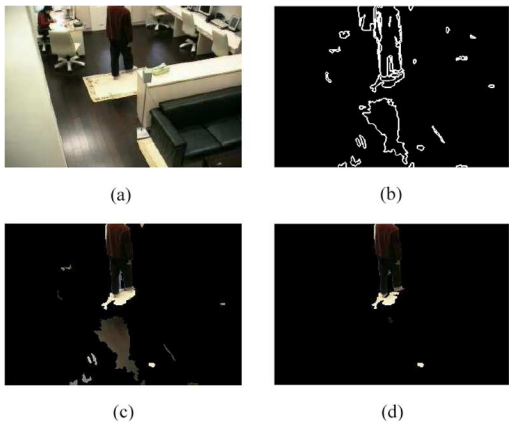

Fig. 10. Shadow effect elimination under the condition with gradual illumination variation.

The final case features two persons entering the scene in order and crossing each other at the top of the image. This produces severe shadow effect in our e-home demo room. Our proposed method will eliminate most of the shadow effect by applying the aforementioned scaling factor λ^* before evaluating Bhattacharyya distance. Empirically, λ_f is set to 0.7 in this paper. An example of shadow effect elimination is shown in Fig. 10. Instead of eliminating the shadow effect during the post processing (after classification step) with some sophisticated algorithm, our method naturally excludes shadows from the background. Fig. 11 shows some detection results in our approach. Notwithstanding, in some pathological cases, the areas that are severely covered with shadows will still be misclassifed as the foreground. This is because the scale-down ratio of intensity is less than 0.7. To overcome this problem, it is necessary to understand the contextual information or structure of the background scene more.

B. System Performance

Our system is implemented on a personal computer with 1.8-GHz Pentium IV processor. Table II shows a run-time analysis of our system. The listed numbers are the processing times of each operation for the *Hall Monitoring* sequence and three cases in our e-home room. The average time for processing each frame is 370 ms, which corresponds to 2.7 fps.

On the other hand, the processing time of the similar MRFs-based approach proposed in [13], which was performed on Pentium III 500 Hz, is 2000 ms per frame with CIF image format. According to the simulation result using software Sandra of SiSoftware company (http://www.sisoftware.co.uk/) and the technique report on their website, 1.8-GHz Pentium 4 is roughly 2.4 times faster than 500-MHz Pentium III. As a consequence, for the approach in [13] the time to process one frame on the personal computer with 1.8-GHz Pentium 4 will be $2000\ \text{ms}/2.4 = 833$ ms. It signifies that our proposed approach should be more than twice faster than the one proposed in [13].

The major reason why our approach could be more efficient is because the background scene is taken as a single region rather than being over-segmented. This makes the number of regions extracted from our proposed algorithm small and, hence, significantly reduces the complexity to perform MRFs-classification operation. In Fig. 12, we show that the number of segmented regions for the first 100 frames of four image sequences are 90 at most even when some are with the complex background scene, In this manner, the computing time of the MRFs-Classification is reduced to about 120 ms as shown in the fourth row of Table II. Although our proposed algorithm is still not ready for real-time applications yet, it has achieved significant improvement in performance. In our ongoing research, the tracking algorithm should be introduced to track the detected foreground objects in order to speed up the performance.

VI. Conclusion

In this paper, we performed the foreground detection at the region level which means that contextual information is taken into consideration. To achieve this, an segmentation approach consisting of region projection, motion marker extraction, and boundary determination is proposed to automatically obtain a set of motion-coherent regions. The main advantage of this method is that it avoids the over-segmentation problem to make the further region classification more efficient. The Bhattacharyya distance is used to measure the distance between two region distributions, and shadow effect is eliminated by applying a scale factor to the region distribution. A statistical framework, MRFs, fuses the cues from background model and the prior knowledge including temporal and spatial coherence to detect the foreground objects in a more accurate and elegant way. Experimental results demonstrate that our proposed method can successfully extract the foreground objects even under situations with illumination variation, shadow, and local motion.

Nevertheless, our proposed method is heavily dependent on the motion estimation algorithm. The segmentation may become unsatisfactory when the object has a large displacement. Therefore, our on-going research is to develop a tracking algorithm which can be used track the detected object. Moreover, the trajectory of the object should also be taken into account for region classification. Last but not least, the result after high-level

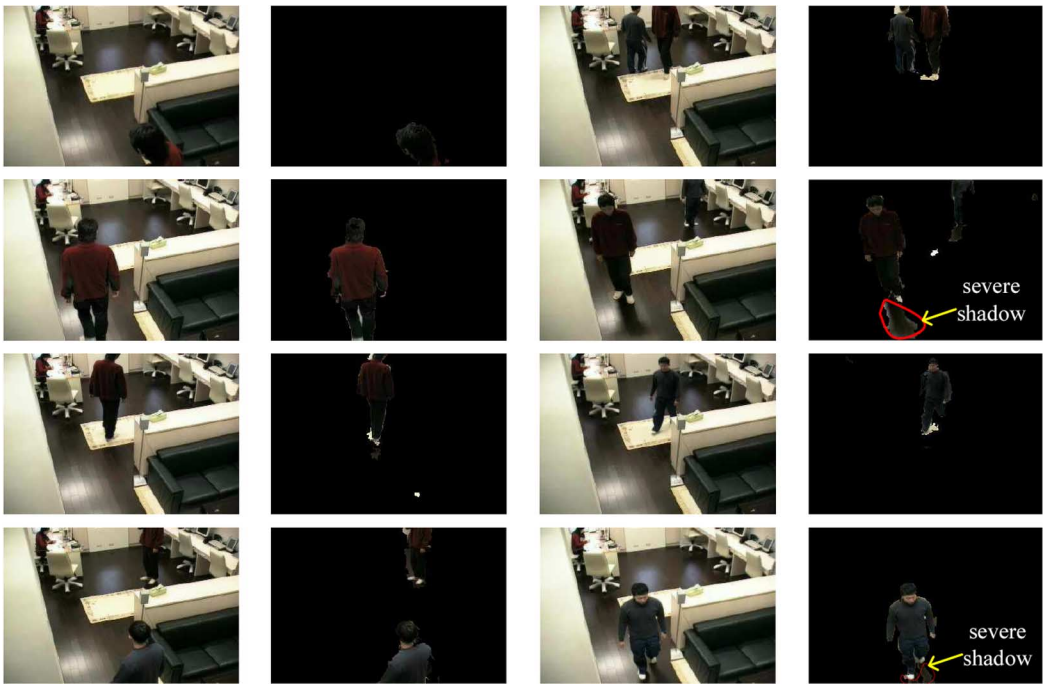

Fig. 11. Figures here show that our approach can successfully eliminate shadow.

TABLE II
RUN-TIME ANALYSIS

Operation	*Hall Monitoring*	Case 1	Case 2	Case 3
Motion Estimation	24.9	33.4	21.5	21.2
Motion Segmentation	166.8	171.6	183.7	152.3
MRFs Classification	97.7	126.0	120.7	118.9
Region Merging	47.6	38.2	33.5	48.2
Data Update	21.6	21.5	18.9	16.7
Total Time	358.6 ms	390.7ms	378.3ms	357.3 ms

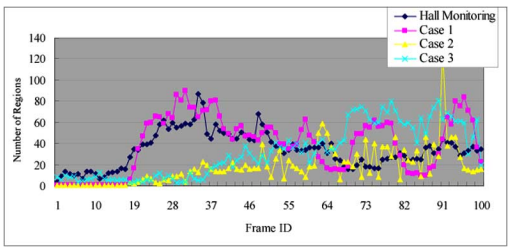

Fig. 12. Number of segmented regions after applying our proposed motion algorithm for four test video sequences. The average number of regions of these four sequences are 32.24, 39.49, 18.32, and 32.35, respectively. Furthermore, the average adjacency of each region of these four sequences are 5.7, 5.6, 3.8, and 5.2, respectively.

recognition will be fed back to update the background and region map to handle the situations with structure variation and severe shadow effect.

REFERENCES

[1] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Trans. Intell. Transport. Syst.*, vol. 3, no. 1, pp. 37–47, Mar. 2002.

[2] S. Y. Chien, S. Y. Ma, and L. G. Chen, "Efficient moving object segmentation algorithm using background registration technique," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 7, pp. 577–586, Jul. 2002.

[3] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 809–830, Aug. 2000.

[4] E. Stringa and C. S. Regazzoni, "Real-time video-shot detection for scene surveillance applications," *IEEE Trans. Image Process.*, vol. 9, no. 1, pp. 69–79, Jan. 2000.

[5] N. Friedman and S. Russell, "Image segmentation in video sequence: A probabilistic approach," presented at the Int. Conf. Uncertainty in Artificial Intelligence, Aug. 1997.

[6] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," presented at the IEEE Computer Society Conf. Computer Vision and Pattern Recognition, Jun. 1999.

[7] P. KaewTraKulPong and R. Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," presented at the 2nd Eur. Workshop Advanced Video-Based Surveillance Systems, 2001.

[8] A. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," presented at the IEEE Int. Conf. Computer Vision Frame-Rate Workshop, 1999.

[9] A. Elgammal, R. Duraiswami, D. Harwood, and L. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proc. IEEE*, vol. 90, no. 7, pp. 1151–1162, Jul. 2002.

[10] D. Bulter and S. Sridharan, "Real-time adaptive background segmentation," presented at the IEEE Int. Conf. Acoustics, Speech, and Signal Processing, 2003.

[11] J. Y. A. Wang and E. H. Adelson, "Spatio-temporal segmentation of video data," *Proc. SPIE*, 1994.

[12] G. D. Borshukov and G. Bozdagi, "Motion segmentation by multistage affine classification," *IEEE Trans. Image Process.*, vol. 6, no. 11, pp. 1591–1594, Nov. 1997.

[13] Y. Tsaig and A. Averbuch, "Automatic segmentation of moving objects in video sequences: A region labeling approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 7, pp. 597–612, Jul. 2002.

[14] Y. Altunbasak, P. E. Eren, and A. M. Tekalp, "Region-based parametric motion segmentation using color information," *Graph. Models Image Process.*, vol. 60, no. 1, pp. 013–023, Jan. 1998.
[15] J. G. Choi, S. W. Lee, and S. D. Kim, "Spatio-temporal video segmentation using a joint similarity measure," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 2, pp. 279–286, Apr. 1997.
[16] F. Dufax, F. Moscheni, and A. Lippman, "Spatio-temporal segmentation based on motion and static segmentation," in *Proc. IEEE Conf. Image Processing*, 1995, pp. 306–309.
[17] S. S. Huang, L. C. Fu, and P. Y. Hsiao, "A region-based background modeling and subtraction using partial directed Hausdorff distance," presented at the IEEE Int. Conf. Robotics and Automation, 2004.
[18] S. S. Huang, L. C. Fu, and P. Y. Hsiao, "A region-level motion-based background modeling and subtraction using MRFs," presented at the IEEE Int. Conf. Robotics and Automation, 2005.
[19] B. K. P. Horn and B. G. Schunck, "Determining optical flow," AI Memo 572, Massachusetts Inst. Technol., Cambridge, 1980.
[20] P. Salembier and M. Pardas, "Hierarchical morphological segmentation for image sequence coding," *IEEE Trans. Image Process.*, vol. 3, no. 9, pp. 639–651, Sep. 1994.
[21] J. Y. A. Wang and E. H. Adelson, "Representation moving images with layers," *IEEE Trans. Image Process.*, vol. 3, no. 5, pp. 625–638, Sep. 1994.
[22] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley, 2000.
[23] A. Rosenfeld and J. L. Pfaltz, "Sequential operations in digital picture processing," *J. Assoc. Comput. Mach.*, vol. 13, pp. 471–494, 1966.
[24] J. Lim and J. B. Ra, "A semantic video object tracking algorithm using three-step boundary refinement," in *Proc. IEEE Int. Conf. Image Processing*, 1999, pp. 159–163.
[25] A. Monnet, A. Mittal, N. paragios, and V. Ramesh, "Background modeling and subtraction of dynamic scenes," in *Proc. IEEE Int. Conf. Computer Vision*, 2003, pp. 1305–1312.
[26] B. Mak and E. Barnard, "Phone clustering using the Bhattacharyya distance," in *Proc. 4th Int. Conf. Spoken Language Processing*, Oct. 1996, vol. 4, pp. 2005–2008.
[27] S. Geman and D. Geman, "Stochastic relaxation gibbs distributions and the bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 11, pp. 721–741, Nov. 1984.
[28] M. J. Black, "The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Comput. Vis. Image Understand.*, vol. 63, no. 1, pp. 75–104, Jan. 1996.
[29] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1505–1518, Dec. 2003.
[30] Y. Kuno, T. Watanabe, Y. Shimosakoda, and S. Nakagawa, "Automated detection of human for visual surveillance system," in *Proc. Inf. Conf. Pattern Recognition*, 1996, pp. 865–869.

Shih-Shinh Huang (S'03) was born on November 8, 1974, in Taiwan, R.O.C. He received the B.S. degree from the National Taiwan Normal University in 1996 and the M.S. and Ph.D. degrees from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1998 and 2007, respectively.

His research interests include computer vision, pattern recognition, and intelligent transportation systems.

Li-Chen Fu (M'84–SM'94–F'04) received the B.S. degree from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1985 and 1987, respectively.

Since 1987, he has been on the faculty of and currently is a professor in both the Department of Electrical Engineering and the Department of Computer Science and Information Engineering of the National Taiwan University. His areas of research interest include robotics, FMS scheduling, shop floor control, home automation, visual detection and tracking, E-commerce, and control theory and applications.

Pei-Yung Hsiao (M'90) received the B.S. degree in chemical engineering from Tung Hai University in 1980 and the M.S. and Ph.D. degrees in electrical engineering from the National Taiwan University, Taiwan, R.O.C., in 1987 and 1990, respectively.

He currently is an Associate Professor with the Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung, Taiwan. His main research interests and industrial experiences are focused on VLSI/CAD, VLSI Design, DIP/SOC, image processing, fingerprint recognition and technology, FPGA rapid prototyping, embedded system, neural network, and expert system.

Dr. Hsiao was granted a scholarship in the 1985 Electronics Engineering Award Examination conducted by the Ministry of Foreign Affairs from the Taiwan, R.O.C., Government for studying microelectronics in Belgium, and he was awarded the 1990 Acer Long Term Ph.D. Dissertation Award from Acer Group.