Computing and Informatics, Vol. 23, 2004, 215–237

FROM GMM TO HGMM: AN APPROACH IN MOVING OBJECT DETECTION

Yunda Sun, Baozong Yuan, Zhenjiang Miao, Wei Wu

Institute of Information Science
Beijing Jiaotong University, Beijing, China, 100044
e-mail: samy.sun@163.com

Manuscript received 29 April 2004; revised 11 October 2004
Communicated by Ladislav Hluchý

Abstract. Background subtraction methods are widely exploited for moving object detection in many applications. A key issue to these methods is how to model and maintain the background correctly and efficiently. This paper describes a foreground detector used in our surveillance system characterized by multiple Gaussian statistics. Compared with the existing methods, our Gaussian mixture model (GMM) differs in model initialization, matching, classification and updating. We propose a fast on-line initialization algorithm to train GMM parameters quickly and correctly. All components of the GMM are classified into three kinds: moving object model, still life model and background model, which is effective for complete detection within a certain period of time. GMMs at different scales are organized in a hierarchical manner to handle sharp illumination changes as well as gradual ones. A convenient way to combine luminance distortion with chrominance distortion is presented for shadow detection in complex scenes. Extensive experimental results are provided to highlight the advantages of our detector.

Keywords: Moving object detection, background subtraction, background model, Gaussian mixture model, hierarchical GMM

1 INTRODUCTION

Moving object detection is known as a hot and challenging research field. The capability of extracting moving objects from a video sequence is fundamental and crucial for many vision systems, such as video surveillance, traffic monitoring, human de-

tection and tracking for video teleconferencing or human-machine interaction, video editing, among other applications. If a system can reliably perform the detection, it can track moving objects, analyze motion patterns, classify interested objects or understand human activities. So far many researches have been still directed toward this topic. A good reason is that failures at the detection level can determine the fate of the entire vision system.

Nowadays there exist three main categories of moving object detection [1]: temporal differencing [2], optical flow [3] and background subtraction [4, 5, 6]. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job for extracting all relevant feature pixels. Optical flow can be used to detect independently moving objects in the presence of camera motion. But most optical flow methods are computationally complex, and could not be applied to full-frame video streams in real-time without specialized hardware. Up to now background subtraction is the most successful category which provides the most complete feature data, though it is extremely sensitive to dynamic scene changes due to lighting and extraneous events. The basic idea of background subtraction is to statistically construct and evolve an estimate of background (often called a background model) frame by frame, and detect moving objects in the scene by the differences between the current frame and the current background model.

Many methods have been proposed in the literature as solutions to efficient and reliable background subtraction. Based on the statistical features used to construct the background model, most of the methods can be classified as minimum and maximum values [4], median value [7, 8, 9], single Gaussian [5, 10, 11], multiple Gaussians [12, 13, 14, 15] (i.e. Gaussian Mixture model abbreviated to GMM), etc. [7]. The GMM is considered as the most sufficient approximation to practical pixel process among them [12]. If each pixel results from a single surface under fixed or slowly-changing lighting, a single adaptive Gaussian per pixel will be enough. In practice, multiple surfaces often appear in the view of a particular pixel and the lighting condition changes as shown in Figure 1. Thus, GMM is required for the background model. Each Gaussian reflects the expectation that samples of the same scene point are likely to display Gaussian noise distributions. Multiple Gaussians reflects the expectation that more than one process may be observed over time. This choice once impressed on us high temporal complexity. However, it does not have such problem anymore because many simplifications assure its real-time application [12, 15]. Moreover, it needs no storage of numerous preceding frames in contrast with other class of methods such as median value, which is really a great virtue.

As reported in some relevant papers [12, 13], the GMM class of methods deals with gradual lighting changes by slowly adjusting the parameters of the Gaussians. It also deals with multimodal distributions caused by shadows, specularities, swaying branches, computer monitors, and other troublesome factors of the real world which are not often mentioned in computer vision. For instance, holes where objects leave or still lives that stop moving are absorbed into background model, which benefits subsequent detection. It recovers quickly when background reappears and has an automatic pixel-wise threshold for flagging potential points as moving object.

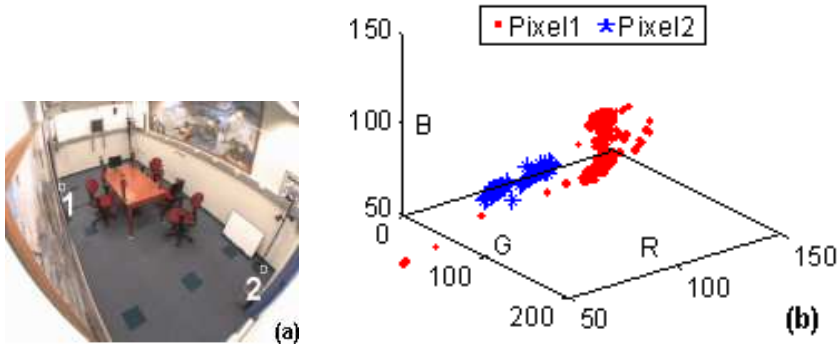

Fig. 1. Two pixels from video "Intelligent Room" over time a) Pixel location b) scatter plot Multimodal distribution can be seen caused by illumination changes

2 MOTIVATION AND NOVELTIES

Reliable and effective moving object detection should be characterized by some important features [7]: high precision, with the two meanings of accuracy in shape detection and reactivity to changes in time; flexibility in different scenarios (indoor, outdoor) or different light conditions; and efficiency, in order to provide detection in real time.

Based on these requirements, there are still some deficiencies in the GMM class of methods: the construction of background model is rather slow in training GMM parameters [15]; sharp illumination changes are seldom handled [1, 18]; shadows are not well eliminated [15], and according to our experiments (Section 5, Experiment C), the objects that temporarily stop moving are absorbed into background model too quickly, which is unfavorable in complete detection and tracking of complex motion, especially indoors.

Being the underlying motivation of our work, these problems may cause the consequent processes, e.g. tracking, recognition, etc., to fail, as the accuracy and efficiency of the detection are definitely very crucial to those tasks. Thus we want to develop a robust and efficient background subtraction algorithm that is characterized by multiple Gaussian statistics and able to cope with the above problems.

Before introducing our novelties, it is necessary and helpful to define some basic terms:

- Moving objects: Objects that keep moving saliently
- Still lives: Objects that have ever moved but stop for a while
- Background: Objects that keep stillness or do not move much around their initial locations, including "still lives" that stop moving for a long time
- Foreground (targets): Moving objects and still lives, contrary to the background.

In our work, a pixel-wise GMM is used to approximate the recent history of a scene point, and different components of the GMM is employed to represent moving objects, still lives or background. So we do not detect moving objects only, but extract the foreground targets consisting of moving objects and still lives, which we are actually interested in. Unless there exist explicit restrictions, the term "model" refers to the GMM.

In this paper, we propose an on-line algorithm to speed up the initialization of model parameters and improve the accuracy of model building by substituting "frequency count" for probability. This algorithm can be applied on even very small sample set such as tens of frames. Similar idea is also extended to model updating where an adaptive learning rate is used instead of a fixed one. Given a new sample, we don't look for the first match but the best one among all candidates, which benefits the complete detection. After the model matching, all components of the GMM are classified into moving object model, still life model and background model. This can be easily implemented with a 3-state finite state machine (FSM). Hence foreground targets contain moving objects and still lives, which is more complete than the existing methods detecting only moving objects; moreover, the time when still lives are absorbed into background can be customized for specific applications, and semantic information like moving and suspending can also be extracted from this bottom-level processing. This paper also proposes an idea of hierarchical GMM (HGMM) as a generic solution to deal with sharp changes of the scene, which uses hierarchical state models without temporal correlation at different scales: global, local and pixel-wise. We combine luminance distortion with chrominance distortion to implement shadow detection in complex scenes, where luminance ratio and inner product greatly simplify its computation.

3 CONVENTIONAL GMM AND IMPROVEMENTS

The conventional GMM method is proposed by Grimson and Stauffer [12]. It models the recent history of each pixel, $\{X_1, \ldots, X_t\}$, as a mixture of K (usually 3–5) Gaussian distributions, where X_i $(i = 1, \ldots, t)$ are measurements of $(\mathrm{R}, \mathrm{G}, \mathrm{B})$ values at each pixel. The probability of observing the current pixel value $X_t = (X_t^r, X_t^g, X_t^b)^T$ is:

$$P(X_t) = \sum_{k=1}^{K} \omega_{k,t} \cdot \eta(X_t, \mu_{k,t}, \Sigma_{k,t}) \tag{1}$$

$$\eta(X_t, \mu_{k,t}, \Sigma_{k,t}) = \frac{1}{(2\pi)^{3/2} \left|\Sigma_{k,t}\right|^{1/2}} \exp\left\{-\frac{1}{2}(X_t - \mu_{k,t})^T \Sigma_{k,t}^{-1} (X_t - \mu_{k,t})\right\} \tag{2}$$

where $\omega_{k,t}$, $\mu_{k,t}$, $\Sigma_{k,t}$ are weight parameter, mean vector and covariance matrix of the k^{th} Gaussian. $\Sigma_{k,t} = \sigma_k^2 \cdot I$. This assumes that red, green and blue color channels are independent and have the same variance. Different Gaussians are assumed to describe different color clusters, and weight parameter $\omega_{i,t}$ denotes the probability that the i^{th} color cluster occurs at time t. At each pixel the K Gaussians are

always ordered by the value of ω/σ. This value increases as a distribution gains more evidence and its variance decreases. By ordering, the most likely background distributions remain on top and the less probable transient background distributions gravitate toward the bottom and are eventually replaced by new distributions. Then for each pixel the first B_t distributions among K ordered Gaussians are chosen as current background model, where B_t is a time-varying value determined by the weight parameter $\omega_{k,t}$ and a user-supplied threshold $T \in (0, 1)$ as follows:

$$B_t = \arg\min \left(\sum_{k=1}^{b} \omega_{k,t} > T \right). \tag{3}$$

Given a new frame, new measurement X_{t+1} of each pixel is checked against its K Gaussians until a match is found, where a match is defined as a pixel value within 2.5 standard deviations of the mean value of a Gaussian. The first matched Gaussian updates all of its parameters with the following on-line K-means approximation, while the others only update their weight parameters and keep their mean vectors and variances unchanged.

$$\omega_{k,t+1} = (1 - \alpha) \cdot \omega_{k,t} + \alpha \cdot M_{k,t+1} \tag{4}$$

$$\text{(If match, } M_{k,t+1} = 1\text{; otherwise, } M_{k,t+1} = 0\text{)}$$

$$\mu_{k,t+1} = (1 - \rho)\mu_{k,t} + \rho \cdot X_t \tag{5}$$

$$\sigma^2_{k,t+1} = (1 - \rho) \cdot \sigma^2_{k,t} + \rho(X_t - \mu_{k,t})^T(X_t - \mu_{k,t}) \tag{6}$$

$$\rho = \alpha \cdot \eta(X_{t+1} \mid \mu_{k,t}, \sigma_{k,t}) \tag{7}$$

where α is the learning rate, and ρ is the learning factor for adapting current distribution. Constant ρ is advised in [12] to reduce computation. If no match is found, the last Gaussian is replaced with mean value X_{t+1}, a high variance and low prior weight. Finally, all pixels in the new frame that have no match in current background model are flagged as foreground pixels.

Pavlidis et al. [14] make a sizable step towards commercial application. In their state-of-art monitoring system DETER (Detection of Events for Threat Evaluation and Recognition), the above algorithms are improved in response to some problems that arose during field-testing. Given an incoming data point X_t, they construct a distribution $g \sim N_3(X_t, 25 \cdot I)$ and measure its dissimilarity against all the available distributions using Jeffreys number, where the choice of 25 (variance) is the result of experimental observation. The Jeffreys number $J(f, g)$ measures how unlikely it is that one distribution $g \sim N_3(\mu_g, \sigma_f^2 \cdot I)$ is drawn from the samples represented by the other . For a theoretical analysis of the Jeffreys number, see [16].

$$J(f, g) = \frac{3}{2}\left(\frac{\sigma_f}{\sigma_g} - \frac{\sigma_g}{\sigma_f}\right)^2 + \frac{1}{2}\left(\frac{1}{\sigma_f^2} + \frac{1}{\sigma_g^2}\right)(\mu_g - \mu_f)^T(\mu_g - \mu_f) \tag{8}$$

The matching process will not stop until all measurements are taken and compared. It strengthens the performance of the detector under certain weather scenarios such as fast moving clouds, when a new pixel value may be matched to multiple distributions resulting in many false negatives. They also make some improvements on model updating.

KaewTraKulPong and Bowden [15] point out two weaknesses of the conventional method: slow learning at the beginning and lacking of shadow detection. They reinvestigate the update equations (4)–(7) to speed up model initialization. An expected sufficient statistics update is introduced to provide a good estimate before enough samples are collected, and then switches to L-recent ($L = 1/\alpha$) window version as before. "L-recent" indicates the recent L samples which determine the current model theoretically. They use a computational color model similar to Horprasert et al. [17] to eliminate shadows. If the difference between a non-background pixel and the current background model in both chromatic and brightness components are within some thresholds, the pixel is not considered as moving object but shadow.

4 MOVING OBJECT DETECTION WITH HGMM

As a primary element of surveillance system, foreground detector implements moving object detection. It is often designed individually on different occasions. Particularly, robust performance of the foreground detector is necessary in the environments of complex lighting or scene changes, especially indoors.

For spatial efficiency as well as temporal efficiency, we select the GMM class of methods from background subtraction category. However, we still find some weaknesses of conventional GMM or improved GMMs (refer to Section 2), which make them inadequate to some aspects. So new improvements are introduced to the original algorithms to solve the problems, and the details will be described in the rest of this section consisting of six parts. Parts A–D are the differences between our GMM and those of the existing methods in model initialization, model matching, model classification and model updating. The idea of hierarchical GMM is explained in part E to handle sharp as well as gradual changes. Part F is our effort in convenient shadow detection.

A. Model Initialization. Though GMM has been used to model the background, the initialization of background model is rather slow with the normal learning rate (e.g. $\alpha = 0.002$), especially in dynamic scenes. For instance, if the first observation of a given pixel belongs to a moving object, it will be after several hundred frames that the genuine Gaussian distributions may be considered as background model. The situation is even worse in busy environments for lack of clean background. A possible solution is to use large learning rate (e.g. $\alpha = 0.1$), which accelerates the building of new Gaussians, but it leads inevitably to an unsteady background model. The resulting background is not clean either.

We find that the reason of this dilemma lies in the estimation of the weight parameter $\omega_{i,t}$, an indication of the posterior probability $P(i \mid X_1, X_2, \ldots, X_t)$

that pixel values have matched Gaussian i given measurements from time 1 through t. However, the incremental calculation of any probability is noise-sensitive on small sample set, and all errors during initialization have to be corrected in further observations. So we use "frequency count" as a more robust statistics than probability in this process. Although the update equations (4)–(7) are improved in [15] to speed up model initialization, they still use probability that leads to an imperfect performance. In comparison, our method is depicted as follows:

- Given the first value X_1 of a pixel, set all K Gaussians with the same distribution:
$$\mu_{k,1} = X_1 \sigma_{k,1} = 20 \qquad \omega_{k,1} = 1 \ (k = 1, 2, \ldots, K). \tag{9}$$
The choice of $\sigma_{k,1} = 20$ is the statistical result of actual distributions over a long time.
- Check a new value X_{t+1} against all K Gaussians to find a match (see part B for how to find the best match). If the match j exists, update it with equation (10)
$$\omega_{j,t+1} = \omega_{j,t} + 1 \qquad \mu_{j,t+1} = \mu_{j,t} + \frac{X_{t+1} - \mu_{j,t}}{\omega_{j,t+1}}$$
$$\sigma^2_{j,t+1} = \sigma^2_{j,t} + \frac{X_{t+1} - \mu_{j,t}(X_{t+1} - \mu_{j,t})}{\omega_{j,t+1}} \tag{10}$$
and reorder the first j Gaussians descendingly by $\omega_{k,t+1}$ $(k = 1, 2, \ldots, j)$. If not, replace the last Gaussian with
$$\mu_{K,t+1} = X_{t+1} \qquad \sigma_{K,t+1} = 20 \qquad \omega_{K,t+1} = 1. \tag{11}$$
- After all L training samples are processed, normalize "frequency count" to probability as below and choose the background model as usual.
$$\omega_{k,L} = \frac{\omega_{k,L}}{\sum_{k=1}^{K} \omega_{k,L}} \tag{12}$$

A small amount of frames (e.g. 30 frames) at the beginning of captured video serve as the training samples. Experiment A illustrates the success of our algorithm as the results show. We attribute our success to robust estimation of weight parameters using "frequency count" on noisy samples. Investigating into the above algorithms, each sample almost makes an equal contribution to the final probabilities, rather than the conventional version where the first sample is of the greatest importance. Meanwhile, mean vectors and variances are updated with adaptive factors instead of fixed ones. Then new Gaussians change quickly to reveal the genuine background, whereas steady Gaussians with many evidences change slowly to avoid destruction from the noise.

B. Model Matching. In conventional schemes, incoming pixels are matched with the ordered distributions in turn from the top toward the bottom of the list. This process stops if only a match is declared. As a new value may satisfy multiple distributions under certain circumstances, model matching is likely to stop before it reaches the right distribution. Jeffreys number is ever used to find the best match in [14], but it is not an efficient measure in pixel-wise computation, and it needs to preset a default variance by experimental observation. Instead, we evaluate and maximize the conditional probability of every match in our system.

$$\begin{aligned} j &= \arg_{k\in\Theta} \max P(X_{t+1} \mid k) \\ &= \arg\max \frac{1}{(2\pi)^{n/2}\,|\Sigma_{k,t}|^{1/2}} e^{-\frac{1}{2}(X_{t+1}-\mu_{k,t})^T \Sigma_{k,t}^{-1}(X_{t+1}-\mu_{k,t})} \\ \Theta &= \{k : \|X_{t+1} - \mu_{k,t}\| < 2.0 \cdot \sigma_{k,t}\} \end{aligned} \tag{13}$$

If incoming pixel has no match (i.e. $\Theta = \emptyset$), the last Gaussian in the ordered list is replaced by an initial distribution as mentioned before. The rule (13) is more straightforward than Jeffreys number, and applied to both initialization and detection. Experiment B proves that it eliminates many false negatives, which leads to complete foreground targets and greatly benefits successive processing.

C. Model Classification. Multiple Gaussian distributions are customarily classified into foreground or background model, which is enough for simple motion detection. As for complex motion, especially indoors, we find those objects that stop moving for a while (referred to as "still lives" in this paper) are absorbed into the background too quickly, which is unfavorable in complete detection and tracking. In addition, it is reasonable to put the time when still lives are absorbed under control for specific applications. To achieve this goal in our foreground detector, Gaussian mixture components of each pixel are grouped into three types after model matching: moving object model, still life model and background model.

Figure 2 shows these models at a single pixel and their transference. Among K Gaussian distributions of a GMM, a Gaussian is "hit" if it is the best match of current pixel value; otherwise it is "missed" by the current value. The times that a Gaussian is hit or missed within a certain period of time are defined as "hit times" (ht) or "missed times" (mt), respectively.

Conventionally, the first B_t Gaussian distributions among K ordered Gaussians are given background model ("bg") decision while the rests are given foreground model ("fg") decision; however, based on this elementary decision and accumulative hit times or missed times of each Gaussian, we can easily separate the transitional model, still life, from absolute background model and moving object model. For each pixel's GMM, our 5-rules advanced decision is applied:

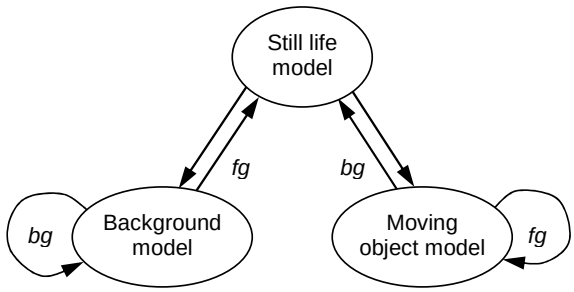

Fig. 2. The transference between three types of models at a single pixel

- Make a "bg" or "fg" decision for each Gaussian distribution as the above.
- Transfer a Gaussian with "fg" decision in background model into still life model; increase mt.
- Transfer a Gaussian with "bg" decision in moving object model into still life model; increase ht.
- Given a "hit" Gaussian with "bg" decision in still life model, if $ht > T_h$ or $mt \neq 0$, transfer it into background model and reset $ht \,\&\, mt$; otherwise increase ht.
- Given a "missed" Gaussian with "fg" decision in still life model, if $mt > T_m$ or $ht \neq 0$, transfer it into moving object model and reset $ht\&mt$; otherwise increase mt.

T_h and T_m are customizable thresholds for "hit times" and "missed times". At last, all pixels in current frame that hit Gaussians in moving object models or still life models with $ht \neq 0$ are flagged as current foreground targets.

The above diagram as a finite state machine (FSM) is used to classify the separate components of the GMM at a single pixel. Thus, our method can be easily implemented with a 3-state FSM depicted in Table 1. Each Gaussian distribution in the GMM has its own version of FSM. The FSM starts when its corresponding Gaussian is created and stops when the Gaussian is finally replaced. Its current state in execution stands for the type of the Gaussian distribution. Main items of the FSM are as follows:

States: moving object model, still life model and background model.

Start state: After model initialization, initial Gaussians have start states of "background model" or "moving object model" according to current background model. All Gaussians created later have start state of "moving object model".

Input alphabet is a set of two-tuples $\{(\text{fg hit}), (\text{fg missed}), (\text{bg hit}), (\text{bg } \textit{missed})\}$. Given a new pixel value, a Gaussian gets a "bg" or "fg" ele-

mentary decision, and it is "hit" or "missed" by the new value. Therefore we can convert the incoming value to an input symbol taking the form of two-tuples.

Transition function: refer to Table 1 for details, where × means an impossible combination of input symbol and state; "↦" represents a transition from current state to the following state; "↻" denotes a transition from current state to itself; "–" indicates that no action (output) happens.

Accepting states are "moving object model" and "still life model" with $ht \neq 0$. If the SFM finishes an input symbol and is in an accepting state, the corresponding pixel in current frame is flagged as foreground target.

<table>
<tr><td></td><td>(fg hit)</td><td colspan="2">(fg missed)</td><td colspan="2">(bg hit)</td><td>(bg missed)</td></tr>
<tr><td rowspan="2">background model</td><td rowspan="2">×</td><td colspan="2">↦ still life model</td><td colspan="2">↻</td><td>↻</td></tr>
<tr><td colspan="2">increase mt</td><td colspan="2">—</td><td>—</td></tr>
<tr><td rowspan="3">still life model</td><td rowspan="2">↻</td><td>$mt > T_m$ or $ht \neq 0$</td><td>otherwise</td><td>$ht > T_h$ or $mt \neq 0$</td><td>otherwise</td><td rowspan="2">↻</td></tr>
<tr><td>↦ moving object model</td><td>↻</td><td>↦ background model</td><td>↻</td></tr>
<tr><td>—</td><td>reset ht & mt</td><td>increase mt</td><td>reset ht & mt</td><td>increase ht</td><td>—</td></tr>
<tr><td rowspan="2">moving object model</td><td>↻</td><td colspan="2">↻</td><td colspan="2">↦ still life model</td><td rowspan="2">×</td></tr>
<tr><td>—</td><td colspan="2">—</td><td colspan="2">increase ht</td></tr>
</table>

Table 1. The 3-state finite state machine (FSM) to implement our method

Furthermore, we would like to show the effectiveness of our method with an example. When a person passes by a pixel and suspends, new Gaussian distribution occurs in this place, which is first deemed as moving object model. After enough evidence is collected, we get "bg" decision and transfer it into still life model rather than background directly. As new values hit the Gaussian continually, hit times increase while the pixel is still contained in foreground targets. Then the person moves on, the Gaussian is relabeled "fg" quickly and restored into moving object model; meanwhile, original distribution in background model, which is already transferred into still life model, is hit again and recovered soon without any misclassification as foreground targets at all. So it is excited that false negatives as well as false alarms are greatly suppressed within a certain period of time.

One may want to use small learning rate instead of the method above, but slow learning detector cannot duly adapt to slight changes of illumination or scene, resulting in much noise or even detector failure, and it is difficult to control the time when still lives are absorbed. Our 5-rules advanced decision works well with large learning rate, leading to both complete detection and nice adaptation; moreover, we could extract some semantic information like move, suspend and stop of foreground targets from this bottom-level process, which may benefit object tracking and behavior recognition. More details can be found in Section 5, Experiment C.

D. Model Updating. After model matching and classification, the parameters of GMM are selectively updated to reflect dynamic changes in the scene, including weight parameters of all Gaussians, mean vector and variance of the best match. An accurate and quick update algorithm is crucial to long-term robust detection.

In [12], Stauffer and Grimson advise to update all parameters with a fixed learning rate, which lacks flexibility on different conditions. In order to establish new Gaussians in good time and protect stable Gaussians from destruction, we apply adaptive ρ to the following formula instead:

$$\rho_{k,t+1} = \alpha/\omega_{k,t+1} \tag{14}$$

$$\mu_{k,t+1} = (1 - \rho_{k,t+1}) \cdot \mu_{k,t} + \rho_{k,t+1} X_{t+1} \tag{15}$$

$$\rho^2_{k,t+1} = (1 - \rho_{k,t+1}) \cdot \rho^2_{k,t} + \rho_{k,t+1} (X_{t+1} - \mu_{k,t})^T (X_{t+1} - \mu_{k,t}) \tag{16}$$

It is interesting that we have used this idea successfully in model initialization.

Periodical flicker of fluorescent lamp is a stubborn problem in indoor surveillance, which brings on nearly random fluctuation of each color channel, owing to variable frame rate. We find it is hard to model the pixel-wise range of variation accurately on this occasion, and much noise can be seen. As the conventional elimination of small regions is always a time-consuming operation, we propose a method to limit the values of σ_{t+1} to a certain number $\sigma_{\min} = 5$ called default minimum variance. The choice of $\sigma_{\min}$ is the result of experimental observation on the typical spread of successive pixel values in large time windows. Experiment D shows the relevant result.

When a new distribution occurs, it replaces the least probable distribution with a low value as its prior weight by convention. Obviously, the weight is no longer normalized and the sum of all weights may well be far away from 1 soon, which invalidates threshold T in determining background model. Pavlidis et al. apply normalization to DETER [14] and associate the weight of new distribution with threshold T. We find this association unfavorable in our experiments and compute the weights as follows:

$$\omega_{K,t+1} = 0.1 \qquad \omega_{i,t+1} = \omega_{i,t} + \frac{\omega_{K,t} - \omega_{K,t+1}}{K - 1} \ (i = 1, 2, \ldots, K - 1) \tag{17}$$

E. Hierarchical GMM. Many methods described in the literature are able to handle gradual changes of illumination by pixel-wise model updating, but can seldom deal with sharp changes in the overall or partial scene, caused by motion of the camera (which is mounted on a pan/tilt head) or occasional sharp lighting changes (e.g., the sun comes out from behind cloud cover, the light is turned off by an intruder, or a casual object blocks the light source). In the presence of sharp changes, nearly all pixels are flagged as foreground targets

for a long while, leading to a serious failure of detector and tracker. Among existing solutions, a built-in mechanism is introduced in [1]. If a majority of image pixels are found to be changing, the detection algorithm temporarily shuts down until the view stabilizes, as determined by a simple two-frame differencing algorithm, and then all pixel statistics are reinitialized. In comparison, the idea of [18, 19] is to model sudden global changes with Hidden Markov Model (HMM) and to switch among all states. For each state, a different background model is maintained. Although more storage is required, this solution results in a faster adaptation to a new state and makes it possible to keep on tracking. However, different states have little temporal correlation in most cases, and it is difficult to estimate the transition probabilities, just as the author hints. So state model as GMM without such correlation can give superior classification results to HMM unless sharp changes occur regularly. Based on this idea, we propose a hierarchical GMM (HGMM) to handle the sharp global or partial lighting changes.

As sharp changes cannot be detected on the pixel level alone, we employ state models (here: GMMs) at different scales. A GMM at the highest scale N is built on the whole view by extracting global mean vector $\overline{X}_{t+1} = \overline{X}^r_{t+1}, \overline{X}^g_{t+1}, \overline{X}^b_{t+1})^T$ of each frame, and used to detect sharp changes in the overall scene. Each GMM at intermediate scales $N-1, \ldots, 2$ is constructed on a part of view by extracting mean vector of the corresponding part of each frame, and employed to detect sharp changes in the partial scene. Each GMM at the lowest scale 1 is implemented on a corresponding pixel as in parts A–D. Hence, all state models are organized in a hierarchical mode, namely HGMM, as shown in Figure 3 a).

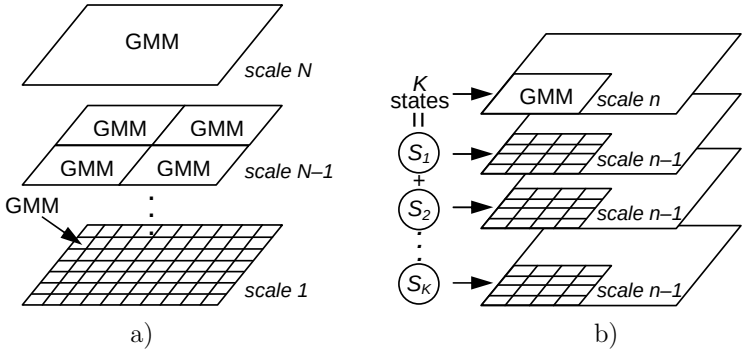

Fig. 3. The sketch map of HGMM a) The hierarchical mode of GMMs of different scales N, $N-1, \ldots, 1$ b) The relationship between GMMs of any two neighboring scales n, $n-1$

In HGMM, state models at different scales essentially store the information of background models of different resolutions. They have strong relations but

cannot be deduced from each other exactly, so we keep all of them in memory for speed and accuracy. Because state transition at the upper scale always requires different state models at the lower scale (e.g. sharp illumination change requires new pixel-wise background model), in order to access easily we make each state in state models at the upper scale point to its own version of state models at the lower scale covering the same region. For instance, if a GMM at the upper scale n has K states (i.e. K Gaussian distributions), each state corresponds to a version of state models at the lower scale $n-1$ which cover the same part of view as the GMM of scale n illustrated in Figure 3 b).

HGMM works in a top-down style. The results of the upper scale decide the performance of the lower scale. Take a two-scales HGMM (i.e. $N = 2$) for instance. Given current frame, the global mean vector $\overline{X}_{t+1}$ is first extracted and matched in the GMM at scale 2. As each Gaussian in the GMM at scale 2 corresponds to a version of pixel-wise GMMs at scale 1, the best match determines which version of GMMs at scale 1 is to be used. With the selected version of GMMs at scale 1, each GMM processes the new value of corresponding pixel in current frame, and classifies it into foreground targets or background. Meanwhile, all model parameters are updated with on-line algorithms, equations (4), (14)–(17), so that the changes within each scale can be tracked.

Initial GMM at scale 2 is learned during initialization while state transitions are detected online. In order to handle states that have not yet been observed previously, the number of states is limited to a maximum number (e.g. $K = 3$). If a new state is observed, the state that has been visited the longest time ago will be replaced by the new one. However, a new version of state models at scale 1 is required here. Thanks to our model initialization algorithm (Part A), we can construct it after a short while, only tens of frames even in busy environments.

Thus the rationale of HGMM is to use state models without temporal correlation at different scales. It is the higher scales that detect sharp global or local changes and switch between different states, which bring an immediate adaptation to various degrees of illumination change. But it does not mean that more scales lead to better performance, because low scales (small regions) are very sensitive to casual changes from large moving objects, and storage cost also increases exponentially with the number of scales. Therefore, the reasonable choice of N is 2-3.

F. Shadow Detection. While segmenting and extracting foreground targets, the misclassification of shadows may cause serious problems, such as object merging, object shape distortion, and even object loss. For this reason, shadow detection is at the core of many applications and has become an active research area. A two-layer taxonomy is developed in [20] for surveying various algorithms presented in this field, among which statistical nonparametric ap-

proach [17] and deterministic nonmodel-based approach [7] are two of the most generic and operable schemes. The former exploits the fact that shadows have similar chromaticity, but lower brightness than the background model. Its computation complexity is somewhat high, and long sequence of clear background is required to automatically determine appropriate thresholds in the statistical learning procedure. The latter works in HSV color space. As a shadow cast on a background does not change its hue significantly, and it often lowers the saturation of the points, 4 deterministic parameters are adjusted by experience to drive the decision process. Unfortunately, we cannot achieve satisfying result in our complicated environment. So the idea of saving computational resources leads us back to the RGB space and we still adopt deterministic parameters.

Similar to [17], brightness and chrominance information are used together. In a new manner, we define luminance distortion as average value ratio and chrominance distortion as the angle between two vectors. If the differences in both of them are within some thresholds, a foreground pixel is considered as the shadow. The decision consists of current pixel value $X_{t+1} = (X_{t+1}^r, X_{t+1}^g, X_{t+1}^b)^T$, background pixel value $X_b = (X_b^r, X_b^g, X_b^b)^T$, current luminance L_{t+1}, background luminance L_b, luminance distortion LD and chrominance distortion CD, where X_b is mean value of the first Gaussian in background model or still life model with $mt \neq 0$, and $T_{lo} < 1$ takes into account the intensity of the shadow, and the stronger the light source is, the lower T_{lo} should be chosen. The use of $T_{hi} > 1$ prevents the identification as foreground of those pixels where the background was slightly highlighted by noise or reflection. T_θ discriminates foreground from hue-preserved shadows.

$$L = \frac{X^r + X^g + X^b}{3} \qquad LD = \frac{L_{t+1}}{L_b} \qquad CD = \arccos \frac{\langle X_{t+1}, X_b \rangle}{\|X_{t+1}\| \cdot \|X_b\|} \tag{18}$$

Then the resulting decision process is as follows:

$$\text{foreground pixel} \begin{cases} \text{shadow} & \text{if } T_{lo} < LD < T_{hi} \text{ and } CD < T_\theta \\ \text{non-shadow} & \text{otherwise} \end{cases} \tag{19}$$

Obviously, our deterministic nonmodel-based approach has great advantage of convenient computation, and there are only three parameters to be determined. As no perfect choices exist in shadow detection, we can say that this method performs well based on experiment F.

5 RESULT EVALUATIONS

Because many new methods are introduced into our foreground detector, we design six different experiments to test them respectively. Some of them have substitutes,

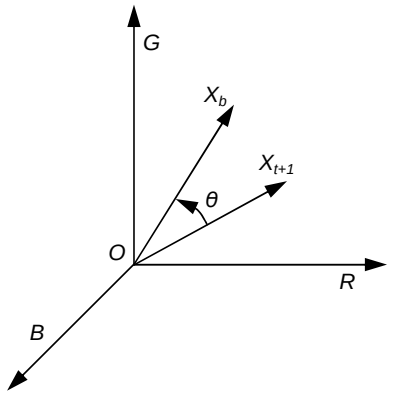

Fig. 4. Chrominance distortion

and then qualitative or quantitative comparisons are made on standard sequences from [21], where 5 video clips and ground truth from manually classifying "Intelligent Room" into foreground targets or shadows are available; others have no alternatives, so we demonstrate our algorithms on self-captured samples. The test platform is a PC with 2.4 GHz CPU and 256 MB RAM. For all GMMs, the number of mixtures is limited to $K = 4$. The standard image resolution is 320×240.

Experiment A: This experiment tests the performance of model initialization with "Highway I" sequence, where the first 30 frames are used to train different models, including conventional GMM [12], improved GMM [15] and our HGMM. As illustrated in Figure 5, it is obvious that the large learning rate cannot solve the problem of slow learning thoroughly, and the improved update equations based on probability cannot either. However, our on-line initialization algorithm can construct clear background model on such a small sample set, and provide the most accurate and complete detection result.

In model initialization, a GMM is parameterized to approximate the recent history of a pixel, and some dominant Gaussians in the GMM form the background model. So we define the recent history of each pixel (i, j), its ground truth, as normalized brightness histogram over the whole sequence. To evaluate the accuracy of different models quantitatively, we compute the average Euclidean distance of ground truth and initialized GMM after training on L frames of "Highway I". The ground truth GT is a 256-dimensional vector. The corresponding statistical vector IG of initialized GMM and the average distance $\overline{d}$ are calculated as follows:

$$IG^m = P(X_{L+1} = m) = \frac{\sum_{k=1}^{K}(\omega_{k,L} \cdot \eta(m, \mu_{k,L}, \Sigma_{k,L}))}{\Sigma_{i=0}^{255}\eta(i, \mu_{k,L}, \Sigma_{k,L})} \ (m = 0, \ldots, 255) \quad (20)$$

$$\overline{d} = \frac{\sum_{i=1}^{I}\sum_{j=1}^{J}\sqrt{\sum_{m=0}^{255}\left(GT_{i,j}^{m} - IG_{i,j}^{m}\right)^2}}{(I \cdot J)} \tag{21}$$

where I, J are width and height of an image. Figure 6 shows the evolutional distance curves of conventional GMM, improved GMM and HGMM, which indicates that HGMM is the best simulation of practical pixel process. Although the background model is more accurate with more evidences acquired, $L = 30 \sim 90$ frames is advised to train the initial models fast and precisely enough.

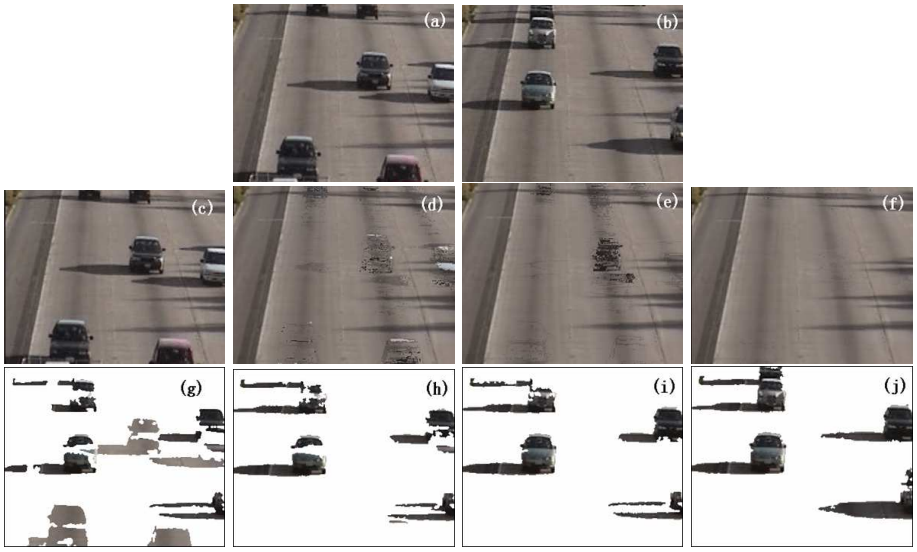

Fig. 5. Highway I a) is frame 1; b) is frame 31; c)–f) are the background models at frame 31 of conventional GMM ($\alpha = 0.002$), conventional GMM ($\alpha = 0.1$), improved GMM and HGMM respectively; g)–j) are corresponding detection results without shadow elimination at frame 31 of the above methods. (Default parameters: $\alpha = 0.002$, $T = 0.4$)

Experiment B: The objective of this experiment is to evaluate different methods of model matching introduced in conventional GMM [12], DETER [14] and HGMM. To make a fair comparison, they are all implemented under the framework of HGMM, where the detection rate and average matching time per pixel are computed on "Intelligent Room" sequence. The detection rate $DR = TP/(TP+FN)$ is often called true positive rate or also referred to in the classification literature, where TP is true positive, FN is false negative. As shown in Table 2, simplified calculation of maximum conditional probability in HGMM is more efficient than that of Jeffreys number in DETER, and achieves more complete foreground targets than conventional GMM method, which only finds

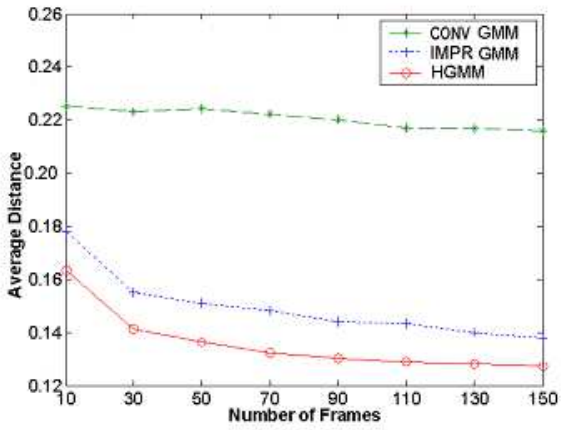

Fig. 6. Accuracy of model initialization (Default parameters: $\alpha = 0.002$, $T = 0.4$)

the first match rather than the best one. So Part 4 B is proved to be a practical idea.

Model matching method in	conventional GMM	DETER	HGMM
Detection rate	93.0 %	94.7 %	94.5 %
Average matching time	0.21 μs	0.40 μs	0.25 μs

Table 2. The detection rate and average matching time of different methods (Default parameters: $L = 30$, $\alpha = 0.005$, $T = 0.4$, $T_h = T_m = 1$ s)

Experiment C: Experiment C is made to illustrate the three types of models, which are expected to take effect in complete detection. Refer to Figure 7 a)–b). The man stops walking for a while and is absorbed into background model immediately in conventional GMM with high learning rate. However, the moving objects and still lives are differentiated in HGMM with 5-rules advanced decision. Within a customized period of time, these two parts still represent the correct foreground targets as Figure 7 f), no matter how large the learning rate is. This effort is of great benefit for indoor surveillance, which abounds in complex motion like suspending, loitering.

Experiment D: Simply, Figure 8 b) shows the noise resulting from periodical flicker of fluorescent lamp. After a default minimum variance $\sigma_{\min} = 5$ is set, most false positives are eliminated, which speeds up post-processing like region growth or morphological filtering, and the boundary of object is clearer than ever.

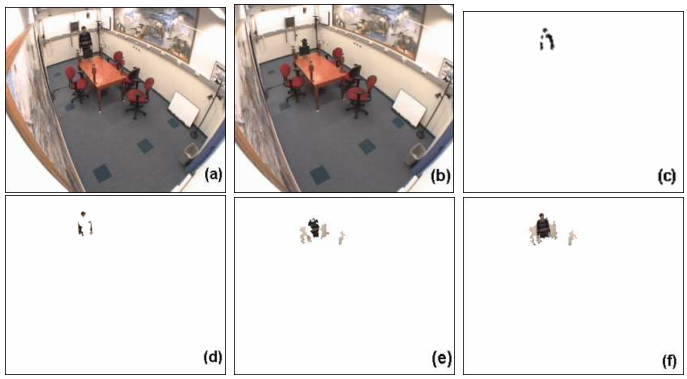

Fig. 7. Intelligent room a) is frame 190; b), c) are the background model and detection result at frame 190 of conventional GMM; d)–f) are moving object, still life and foreground target at frame 190 of HGMM (Default parameters: $L = 30$, $\alpha = 0.005$, $T = 0.4$, $T_h = T_m = 1\,\mathrm{s}$)

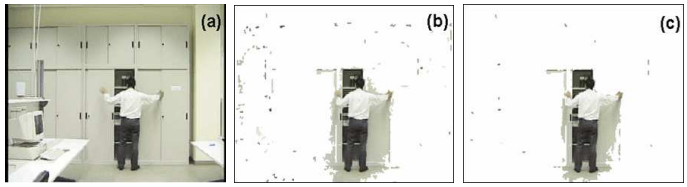

Fig. 8. Laboratory a) is frame 350; b), c) are detection results of HGMM without and with default minimum variance. (Default parameters: $L = 30$, $\alpha = 0.005$, $T = 0.4$, $T_h = T_m = 1\,\mathrm{s}$)

Experiment E: This experiment demonstrates the exciting ability of HGMM to handle irregular sharp illumination changes. In our self-captured sequence "intruder", the light is on at the beginning, and turned off at frame 65. It is switched on and off again at frame 132 and 228. Generally, conventional GMM spends several hundred frames initializing new states at normal learning rate. In comparison, two-scales HGMM is used here. While sharp illumination change occurs after frame 131 (Figure 9 a)), the view stabilizes at frame 140 (Figure 9 b)), and HGMM detector recovers bright state (Figure 9 c)). Similarly, illumination changes again after frame 227 (Figure 9 d)), and HGMM detector switches to dark state soon at frame 231 (Figure 9 e)–f)). So it can be concluded that HGMM responds quickly to sharp changes and to beneficial for continuous tracking.

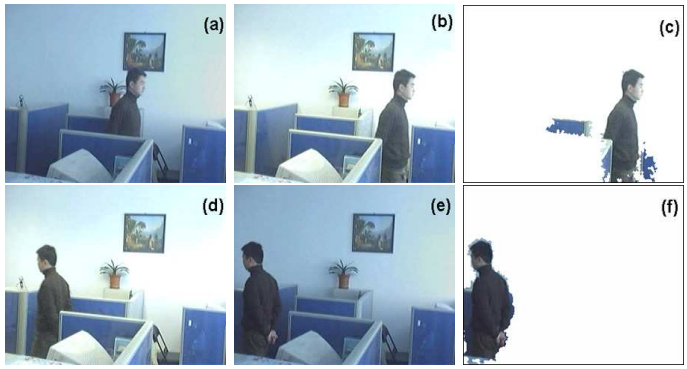

Fig. 9. Results of HGMM to handle sharp illumination changes without shadow elimination (Default parameters: $L = 30$, $\alpha = 0.005$, $T = 0.4$, $T_h = T_m = 2\,\mathrm{s}$)

Experiment F: The last experiment shows our good result on shadow elimination. We adopt the evaluation metrics in [20], the shadow detection rate η and shadow discrimination rate ξ:

$$\eta = \frac{TP_S}{TP_S + FN_S} \qquad \xi = \frac{\overline{TP}_F}{TP_F + FN_F} \tag{22}$$

where the subscript S stands for shadow and F for foreground, the $\overline{TP}_F$ is the number of ground-truth points of the foreground objects minus the number of points detected as shadows, but belonging to foreground pixels.

An accurate detection result is given in Figure 10, where $T_{lo} = 0.6$, $T_{hi} = 1.1$ and $T_\theta = 3$. Table 3 is further quantitative comparison of 5 different methods made on "Intelligent Room". All algorithms do not implement any background updating process since each proposes a different approach. Refer to [20] for relevant data. It can be seen that our method of shadow detection achieves good detection rate as well as good discrimination rate. Moreover, as a deterministic nonmodel-based approach, it may be considered as the most convenient algorithm in this field. As no perfect choice exists in shadow detection, we find that our method shows satisfying performance in experimental system, especially indoor scenes.

Shadow detection Method of	SNP	SP	DNM1	DNM2	HGMM
$\eta\,\%$	72.82 %	76.27 %	78.61 %	62.00 %	75.80 %
$\xi\,\%$	88.90 %	90.74 %	90.29 %	93.89 %	86.42 %

Table 3. The shadow detection rate and shadow discrimination rate of different methods (Default parameters: $L = 60$, $T = 0.4$, $T_h = T_m = 1\,\mathrm{s}$)

Fig. 10. Intelligent Room a)–c) are original image, detection result without and with shadow elimination (frame 300)

6 CONCLUSION AND OUTLOOK

This paper reviews the GMM class of methods in background subtraction, a developing approach in moving object detection. The recent history of each pixel is modeled as a mixture of Gaussian distributions, and all changes are tracked with on-line update equations. Based on the previous work, a new foreground detector is presented for our system, which differs from the existing methods in many aspects. It aims at achieving more accurate detection with both spatial and temporal efficiency. Firstly, an on-line algorithm is proposed to initialize new models, which runs seamlessly under the framework of conventional GMM and overcomes its weakness of slow learning successfully. Secondly, the best match with the maximum conditional probability is found in model matching rather than the first match, and a comparison is made among relevant methods to show the usefulness of our method. Thirdly, three types of models are distinguished to perform complete detection within a customized period of time. FSM facilitates the implementation. Thus foreground targets contain moving objects and still lives, and semantic information like moving and suspending can also be extracted from this bottom-level process. Fourthly, we describe the improvements to overcome some problems that we find in model updating. Fifthly, since how to handle sharp changes is always a challenging problem in moving object detection, the idea of HGMM is introduced in our work, which uses hierarchical state models without temporal correlation to detect global, local or pixel-wise changes. Fast adaptation to sharp illumination changes can be seen from relevant results. Lastly, we propose a convenient method to detect moving shadows, where luminance distortion and chrominance distortion are tested on foreground pixels. This algorithm significantly reduces additional computational burdens, and performs well in our experimental system.

Currently, our foreground detector is running in real time with all of the above characteristics (i.e. fps = 30), and we still have some further work to do for this promising approach:

- Background subtraction is sensitive to noise or dynamic lighting change, without the exception of the GMM. Post-processing is generally required to eliminate

small noise or false targets. In our opinion, current methods need to be further studied to enhance the ability of modeling the range of variation, especially the update of variance parameters, and may well become the dominant direction of future research.

- In GMM, each pixel is always modeled as an independent one, which is clearly a simplification, because two neighboring pixels corresponding to the same object are highly correlated. How to employ such spatial constraints at lower levels to improve moving object detection is still an open issue.
- HGMM currently updates with the whole incoming frame. In practice, the moving object and moving shadows had better not participate in model updating, because they never belong to any parts of background. So similar method as [7] may be introduced to discriminate foreground targets into moving visual object (MVO), ghost (i.e. the detection of false objects), MVO shadow and ghost shadow, where selective update is used to protect the background model from destruction.
- HGMM is effective to handle sharp changes, but different versions of GMMs at the same scale are updated separately. Thus not all of illumination states could reflect casual changes due to extraneous events, such as removing an object before switching the light, which leaves a hole to the new illumination state. Further research is conducted on eliminating such false positives.

Acknowledgements

We would like to thank A. Prati, I. Mikic, M. M. Trivedi, R. Cucchiara, and CVRR in University of California at San Diego, for sharing their testbed data. This work is supported by the National Natural Science Foundation of China (No. 60441002) and the University Key Research Project (No. 2003SZ002, No. JSJ04010).

REFERENCES

[1] Collins, R. T.—LIPTON, A. J.—Fujiyoshi, H.—Kanade, T.: Algorithms for Cooperative Multisensor Surveillance. Proceedings of the IEEE, Vol. 89, 2001, No. 10, pp. 1456–1477.

[2] Rosin, P. L.—Ellis, T.: Image Difference Threshold Strategies and Shadow Detection. Proc. British Machine Vision Conf., 1995, pp. 347–356.

[3] Barron, J.—Fleet, D.—Beauchemin, S.: Performance of Optical Flow Techniques. International Journal of Computer Vision, Vol. 12, 1994, No. 1, pp. 42–77.

[4] Haritaoglu, I.—Davis, L. S.—Harwood, D.: W4 Who? When? Where? What? A Real Time System for Detecting and Tracking People. In FGR98, 1998.

[5] Wren, C.—Azarbayejani, A.—Darrell, T.—Pentland, A.: Pfinder: Real-Time Tracking of The Human Body. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, 1997, No. 7, pp. 780–785.

[6] Toyama, K.—Krumm, J.—Brumitt, B.—Meyers, B.: Wallflower: Principles and Practice of Background Maintenance. In Proc. Int. Conf. Computer Vision, Corfu, Greece, 1999, pp. 255–261.

[7] Cucchiara, R.—Grana, C.—Piccardi, M.—Prati, A.: Detecting Moving Objects, Ghosts, and Shadows in Video Streams. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, 2003, No. 10, pp. 1337–1342.

[8] Lo, B. P. L.—Velastin, S. A.: Automatic Congestion Detection System for Underground Platforms. Proc. Int'l Symp. Intelligent Multimedia, Video, and Speech Processing, 2000, pp. 158–161.

[9] Gloyer, B.—Aghajan, H. K.—Siu, K. Y.—Kailath, T.: Video-Based Freeway Monitoring System Using Recursive Vehicle Tracking. Proc. SPIE Symp. Electronic Imaging: Image and Video Processing, 1995.

[10] Mckenna, S. J.—Jabri, S.—Duric, Z.—Rosenfeld, A.—Wechsler, H.: Tracking Groups of People. Computer Vision and Image Understanding, Vol. 80, 2000, No. 1, pp. 42–56.

[11] Jabri, S.—Duric, Z.—Wechsler, H.—Rosenfeld, A.: Detection and Location of People in Video Images Using Adaptive Fusion of Color and Edge Information. Proc. Int'l Conf. Pattern Recognition, 2000, pp. 627–630.

[12] Stauffer, C.—Grimson, W. E. L.: Learning Patterns of Activity Using Real-Time Tracking. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 22, 2000, No. 8, pp. 747–757.

[13] Stauffer, C.—Grimson, W. E. L.: Adaptive Background Mixture Models for Real-Time Tracking. Proc. Computer Vision and Pattern Recognition 1999 (CVPR'99), June 1999.

[14] Pavlidis, I.—Morellas, V.—Tsiamyrtzis, P.—Harp, S.: Urban Surveillance Systems: From the Laboratory to the Commercial World. Proceedings of the IEEE, Vol. 89, 2001, No. 10, pp. 1478–1497.

[15] Kaewtrakulpong, P.—Bowden, R.: An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection. Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems (AVBS01), 2001.

[16] Lin, J.: Divergence Measures Based on the Shannon Entropy. IEEE Trans. Inform. Theory, Vol. 37, 1991, pp. 145–151.

[17] Horprasert, T.—Harwood, D.—Davis, L. S.: A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection. Proc. IEEE Int'l Conf. Computer Vision '99 FRAME-RATE Workshop, 1999.

[18] Stenger, B.—Ramesh, V.—Paragios, N.—Coetzee, F.—Buhmann, J. M.: Topology Free Hidden Markov Models: Application to Background Modeling', ICCV 2001, July 2001, pp. 294–301.

[19] Adaptive Background Estimation Using a Hidden Markov Model. `http://www-dbv.cs.uni-bonn.de/Video/`.

[20] Prati, A.—Mikic, I.—Trivedi, M. M.—Cucchiara, R.: Detecting Moving Shadows: Algorithms and Evaluation. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, 2003, No. 7, pp. 918–923.

[21] The ATON project. `http://cvrr.ucsd.edu/aton/shadow`.

Yunda Sun received his Bachelor degree in mechanical engineering and automation in 2000 from Beijing Jiaotong University, Beijing, China. Since then he has been working towards the Ph.D. degree in information and communication engineering in the Institute of Information Science, Beijing Jiaotong University. His current research interests include intelligent vision surveillance, computer vision and 3D reconstruction.

Baozong Yuan received the Ph.D. degree in electrical engineering from USSR in 1960. He has joined the Beijing Jiaotong University in 1953. He was a visiting professor at the University of Pittsburgh, USA, and the University of Wales, UK in 1982, 1983, and 1988, respectively. His research interests include computer vision, virtual reality, image processing, computer graphics, speech signal processing, and multimedia information processing and data communication.

Zhenjiang Miao received his Bachelor degree from Tsinghua University in 1987, Master and Ph.D. degree from Northern Jiaotong University in 1990 and 1994. Between 1995 and 1998, he worked in France as a post-doctoral fellow in ENSEEIHT, Institut National Polytechnique de Toulouse, and as a researcher in the Institut National de la Recherche Agronomique at Sophia Antipolis. Then he worked in Canada between 1998 and 2004 in the Institute of Information Technology, National Research Council Canada, Nortel Networks, etc. He has joined Beijing Jiaotong University in April 2004. He published more than 50 papers in pattern recognition, multimedia information processing, wireless telecommunications, etc. His current research interests include pervasive & mobile computing, intelligent human-machine interaction, biometrics, etc.

Wei Wu received her Bachelor degree in information and communication engineering from Beijing Jiaotong University, Beijing, China, in 2002. Now she is pursuing the M.E. degree in information and communication engineering in Institute of Information Science, Beijing Jiaotong University. Her current search interests include image processing, pattern recognition and artificial intelligence.