Regularized Background Adaptation: A Novel Learning Rate Control Scheme for Gaussian Mixture Modeling

Horng-Horng Lin, *Student Member, IEEE*, Jen-Hui Chuang, *Senior Member, IEEE*, and Tyng-Luh Liu, *Member, IEEE*

***Abstract*—To model a scene for background subtraction, Gaussian mixture modeling (GMM) is a popular choice for its capability of adaptation to background variations. However, GMM often suffers from a tradeoff between robustness to background changes and sensitivity to foreground abnormalities and is inefficient in managing the tradeoff for various surveillance scenarios. By reviewing the formulations of GMM, we identify that such a tradeoff can be easily controlled by adaptive adjustments of the GMM's *learning rates* for image pixels at different locations and of distinct properties. A new rate control scheme based on high-level feedback is then developed to provide better regularization of background adaptation for GMM and to help resolving the tradeoff. Additionally, to handle lighting variations that change too fast to be caught by GMM, a heuristic rooting in frame difference is proposed to assist the proposed rate control scheme for reducing false foreground alarms. Experiments show the proposed learning rate control scheme, together with the heuristic for adaptation of *over-quick* lighting change, gives better performance than conventional GMM approaches.**

***Index Terms*—Background subtraction, Gaussian mixture modeling, learning rate control, surveillance.**

I. Introduction

FOR video surveillance using static cameras, background subtraction is often regarded as an effective and efficient method for differentiating foreground objects from a background scene. The performance of background subtraction highly depends on how a background scene is modeled. Ideally, a perfect design of background modeling should be able to tolerate various background variations without losing the sensitivity in detecting abnormal foreground objects. However, the tradeoff between model robustness and model sensitivity is commonly encountered in practice and is hard to balance within a single background modeling framework.

Among various background modeling approaches, e.g., [1], [3]–[5], [7], [8], [12], [15], [16], [18]–[23], [25], [26], [28], [29], the Gaussian mixture modeling (GMM) [5], [7], [23] is known to be effective in sustaining background variations, e.g., waving trees, due to its use of multiple buffers to memorize scene states. It is hence widely adopted as a base framework in many later developments [9]–[11], [13], [17], [24], [30]. However, GMM often suffers from the tradeoff between model robustness to background changes and model sensitivity to foreground abnormalities, abbreviated as *R-S tradeoff* in later discussions. For instance, a Gaussian mixture model being tuned to tolerate quick changes in background may also adapt itself to stationary objects, e.g., unattended bags left by passengers, too quickly to issue reliable alarms. The lack of a simple and flexible way to manage the R-S tradeoff for various scenarios motivated this research to reexamine the formulations of the GMM.

In the original formulations of GMM, every image pixel, regardless of its intensity being changed or not, is given the same setting of *learning rates* in background model estimation, which is inefficient in managing the R-S tradeoff. Considering a pixel of background that was just uncovered from occlusion of a moving object, the corresponding Gaussian mixture model for this pixel should be updated in a slower pace than that for a stable background pixel, to prevent false inclusion of moving shadows or motion blurs into background. Nonetheless, in the original GMM formulations, an identical learning rate setting is applied to all image pixels, leaving no space for tuning the background adaptation speeds for this case. We therefore highlight the importance of adaptive learning rates control in space and in time and develop a new rate control scheme based on the high-level feedback of pixel properties for the GMM.

There are several features of the proposed scheme of learning rate control for the GMM. First, two types of learning needs are identified for a Gaussian mixture model (for an image pixel), one for controlling the model estimation accuracy and the other for regularizing the R-S tradeoff. Different from previous works, e.g., [9] and [23], that use a single learning rate setting for both learning needs, the proposed rate control scheme distinguishes two different types of learning rates and manipulates them independently. Second, the background adaptation rates for image pixels are set individually in space. Image pixels at different locations may thus exhibit distinct behaviors in background adaptation for accommodating local scene changes. Third, for every image pixel, its learning rate for regularizing the R-S tradeoff is computed based on the high-level feedback of its latest pixel type, i.e., as background, stationary foreground, or moving foreground. Under this feedback control, the learning rate setting

Manuscript received December 21, 2009; revised June 16, 2010; accepted August 22, 2010. Date of publication September 13, 2010; date of current version February 18, 2011. This work was supported by the QNAP System Inc. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sharathchandra Pankanti.

H.-H. Lin is with the Department of Computer Science, National Chiao Tung University, Hsinchu 30010, Taiwan, and also with QNAP System Inc., New Taipei City 221, Taiwan.

J.-H. Chuang is with the Department of Computer Science, National Chiao Tung University, Hsinchu 20010, Taiwan.

T.-L. Liu is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan.

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

Digital Object Identifier 10.1109/TIP.2010.2075938

for an image pixel can be dynamically adjusted in time, according to its type, and with respect to different application scenarios.[1] The more pixel types are allowed, the higher flexibility in background adaptation can be attained. Fourth, a heuristic for *adaptation of over-quick lighting change* is suggested to assist the learning rate control to adapt very rapid lighting changes in background. This heuristic enhances the model robustness to speedy lighting variations without sacrificing the sensitivity in detection of significant foreground motions. To sum up, we maintain that, via a careful design of learning rate control for the GMM, the R-S tradeoff can be effectively and efficiently regularized in fulfilling various needs in video surveillance.

A. Related Work

Balancing the R-S tradeoff has long been an important task in background modeling. In [25], Toyama *et al.* explore several scenarios that are hard to be handled by background modeling and propose a hybrid approach to maintain background models at different spatial scales. In [1], Boult *et al.* apply different learning rates to foreground and background pixels to increase the model sensitivity for single Gaussian formulation and develop cleaning algorithms to reduce false alarms. In [6], Gao *et al.* use statistical analysis to tune parameters in background modeling, including the number of Gaussian components and the learning rate, for controlling the tradeoff. In [14], Li *et al.* utilize spatio-temporal features to model complex backgrounds and develop a criterion to select the learning rate for the adaptation of *once-off* background change. In [9], Harville discusses some tradeoffs frequently encountered by GMM and adopts high-level feedback as a remedy. Also based on GMM, Tian *et al.* propose a weight exchange scheme based on object-level feedback to prevent foreground fragmentation in the detection of static object [24]. In [30], Zivkovic analyzes the appropriate number of mixture components for GMM and dynamically removes some mixture components for computational efficiency. In [13], Lee proposes a new rate control formulation for the learning of Gaussian parameters to enhance the accuracy and *convergence speed* of background model estimation. Model robustness to background changes is improved by Lee's learning rate control without obvious side-effects on model sensitivity. In [27], a two-layer GMM is proposed by Yang *et al.* to learn foreground and background models at different learning rates and to achieve better foreground segmentation results. Beyond Gaussian-based formulations, Elgammal *et al.* adopt kernel density estimation to compute background models, and combine short-term and long-term models to balance the R-S tradeoff [4]. Despite the effectiveness in background modeling for all the approaches mentioned above, no comprehensive investigation into the relationship between model learning rates and the tradeoff control for different surveillance scenarios within a single background modeling framework has been conducted.

Note also that the idea of adopting high-level feedbacks, e.g., using foreground pixel type, in background modeling is not new [1], [9], [24], [27]. However, the proposed feedback control over learning rates has several novel features. First, to the best of our knowledge, the proposed work is the first to apply independent controls over two types of learning rates for simultaneously enhancing the model estimation accuracy and regularizing the R-S tradeoff. High-level feedbacks are applied only to the learning rate control related to the R-S tradeoff. Based on our study, this independent control of two-type learning rates is a key to derive a robust background modeling system. Second, a new rate control framework capable of managing multiple pixel types as feedbacks is demonstrated to be practical and feasible. Third, the need of dynamically adjusting the learning rates for pixels of background type is firstly identified in this study. This particular learning rate control for background pixels can increase model sensitivity to hovering objects with little side effects to model robustness.

B. Model Accuracy, Robustness, and Sensitivity

To estimate a density distribution from a sequence of intensities $I_{0,\mathbf{x}}, \ldots, I_{t,\mathbf{x}}$[2] for a pixel at a position $\mathbf{x}$ via GMM, three issues regarding model accuracy, robustness, and sensitivity need to be addressed. Specifically, a mixture model consisting of N Gaussian distributions at time instance t can be denoted by

$$P(I_{t,\mathbf{x}}) = \sum_{n=1}^{N} w_{t-1,\mathbf{x},n} \mathcal{N}\left(I_{t,\mathbf{x}}; \mu_{t-1,\mathbf{x},n}, \sigma^2_{t-1,\mathbf{x},n}\right)$$

where $\mathcal{N}$ symbolizes a Gaussian probability density function

$$\mathcal{N}(I; \mu, \sigma^2) \doteq \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(I-\mu)^2}{2\sigma^2}\right)$$

and $\mu_{t-1,\mathbf{x},n}$ and $\sigma^2_{t-1,\mathbf{x},n}$ are the Gaussian parameters of the nth model, and $w_{t-1,\mathbf{x},n}$ is the respective mixture weight. For maintaining this mixture model, the parameters μ_{t-1}, σ^2_{t-1} and w_{t-1} need to be updated based on a new observation $I_{t,\mathbf{x}}$. In the GMM, the update rule for μ, for the case that $I_{t,\mathbf{x}}$ matches the nth Gaussian model, is

$$\mu_{t,\mathbf{x},n} = (1-\rho)\mu_{t-1,\mathbf{x},n} + \rho I_{t,\mathbf{x}}$$

where $0 \le \rho \le 1$ is a *learning rate*[3] that controls how fast the estimate μ converges to new observations. Likewise, similar update rules can be applied to renewing σ^2 and w, given corresponding learning rates.

In updating the Gaussian parameters μ and σ^2, their values should reflect the up-to-date statistics of a scene as accurately as possible. It is thus preferable to set their learning rates to large values to quickly derive Gaussian distributions that fit new observations. Also, as noted in [13], setting higher learning rates for μ and σ^2 improves model convergency and accuracy and brings few side effects in model stability.

While the model estimation accuracy depends on the learning rates for μ and σ, one can see that the R-S tradeoff is affected by the learning rate for the mixture weight w. In the original GMM for background model estimation, the classification of

[1] For example, while pixels of stationary objects may need to be quickly adapted into the background for the application of moving object detection, they should be stably identified as foreground for the application of unattended object detection.

[2] Here, $I_{t,\mathbf{x}} \in \mathbb{R}$ denotes the 1-D pixel intensity only. However, all of our formulations can be easily extended to multidimensional color image processing, e.g., $\mathbf{I}_{t,\mathbf{x}} \in \mathbb{R}^3$.

[3] The definition of learning rate is inherited from [23].

```
Algorithm 1: Background model maintenance
Parameters: $T_\sigma (= 2.5)$, $\sigma_0^2 (= 10^2)$, $w_0 (= 0.01)$
// Model matching
$M_{t,\mathbf{x},n} = 0, \forall n = 1, \ldots, N$
$d_{t,\mathbf{x},n} = \inf, \forall n = 1, \ldots, N$
for $n = 1, \ldots, N$ do
    if $|I_{t,\mathbf{x}} - \mu_{t-1,\mathbf{x},n}| \leq T_\sigma \, \sigma_{t-1,\mathbf{x},n}$ then $d_{t,\mathbf{x},n} = -w_{t-1,\mathbf{x},n}$
$l(t,\mathbf{x}) = \arg\min_{n=1,\ldots,N} \; d_{t,\mathbf{x},n}$
if $d_{t,\mathbf{x},l(t,\mathbf{x})} \neq \inf$ then $M_{t,\mathbf{x},l(t,\mathbf{x})} = 1$ else $l(t,\mathbf{x}) = 0$
// Model renewing
$w_{t,\mathbf{x},n} = (1 - \eta_{t,\mathbf{x}}(\beta))\, w_{t-1,\mathbf{x},n} + \eta_{t,\mathbf{x}}(\beta)\, M_{t,\mathbf{x},n}, \forall n$
if $M_{t,\mathbf{x},l(t,\mathbf{x})} = 1$ then
    // Update phase
    $\rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) = \alpha \mathcal{N}\left(I_{t,\mathbf{x}};\, \mu_{t-1,\mathbf{x},l(t,\mathbf{x})}, \sigma^2_{t-1,\mathbf{x},l(t,\mathbf{x})}\right)$
    $\mu_{t,\mathbf{x},l(t,\mathbf{x})} = \left(1 - \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\right) \mu_{t-1,\mathbf{x},l(t,\mathbf{x})} + \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\, I_{t,\mathbf{x}}$
    $\sigma^2_{t,\mathbf{x},l(t,\mathbf{x})} = \left(1 - \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\right) \sigma^2_{t-1,\mathbf{x},l(t,\mathbf{x})} + \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\, (I_{t,\mathbf{x}} - \mu_{t,\mathbf{x},l})^2$
else
    // Replacement phase
    $k = \arg\min_{n=1,\ldots,N} \; w_{t-1,\mathbf{x},n}$
    $\mu_{t,\mathbf{x},k} = I_{t,\mathbf{x}}$
    $\sigma^2_{t,\mathbf{x},k} = \sigma_0^2$
    $w_{t,\mathbf{x},k} = w_0$
$w_{t,\mathbf{x},n} = w_{t,\mathbf{x},n} / \sum_{n=1}^{N} w_{t,\mathbf{x},n}, \forall n$
```

Fig. 1. Algorithm 1.

Gaussian models into foreground and background is done by evaluating their mixture weights through thresholding. The Gaussian models that appear more often will receive larger weights in the model updating process and will possibly be labeled as background [23]. However, the frequency of model occurrence should not be the only factor that guides the changes of mixture weights. For example, one may prefer to give large weights to the Gaussian models of tree shadows (for background adaptation) while to keep small weights to those of parked cars (for foreground detection), despite the similar frequencies of occurrence of these two objects. By incorporating the high-level information of pixel types, e.g., of shadow or car, into the weight updating process, flexible background modeling can then be carried out. As more pixel types are designated by a surveillance system, more appropriate controls on weight changes can be advised accordingly, which will help resolving the R-S tradeoff in background modeling. Based on this observation, we propose a feedback scheme for the learning rate control of the GMM.

The remainder of this paper is organized as follows. In Section II, the formulations of the proposed learning rate control based on high-level feedback are detailed. The adopted heuristic for adaptation of over-quick lighting change is introduced as well. In Section III, experimental comparisons of foreground detection results among [13] and [23] and our approach are presented. Finally, brief discussions of the proposed rate control scheme and future work are given in Section IV.

II. Learning Rate Control Via High-Level Feedback

Our presentations of the learning rate control scheme are divided into three parts. First, an algorithm of *background model maintenance* using the GMM is proposed, wherein two types of learning rates are formally defined. We highlight the importance of the learning rate control for mixture weights and elaborate its relationship to foreground pixel labeling. Second, a feedback scheme that controls the learning rates for mixture weights is detailed. Under this feedback control, different learning rates can be applied to different image locations and scene types, which makes dynamic background adaptation possible. Third, a heuristic based on frame difference is introduced to assist the learning rate control for the adaptation of over-quick lighting changes. False alarms caused by, e.g., sudden sunshine changes in the background can hence be suppressed by this heuristic while significant object motions can still be captured.

A. Background Model Maintenance

Given a new observation of pixel intensity $I_{t,\mathbf{x}}$, the task of background model maintenance is to match this new observation to existing Gaussian distributions, if possible, and to renew all the parameters of the Gaussian mixture model for this pixel. The detailed steps of the proposed background model maintenance using the GMM is shown in Algorithm 1 (see Fig. 1).

For the model matching in Algorithm 1, $l(t,\mathbf{x})$ is utilized to index the best matched Gaussian model of $I_{t,\mathbf{x}}$, if existing. Otherwise, $l(t,\mathbf{x}) = 0$ will be set to indicate $I_{t,\mathbf{x}}$ is a brand new observation and should be modeled by a new Gaussian distribution. The matching results of $I_{t,\mathbf{x}}$ can be recorded by model matching indicators, i.e.,

$$M_{t,\mathbf{x},n} = \begin{cases} 1, & \text{if } n = l(t,\mathbf{x}) \\ 0, & \text{otherwise} \end{cases} \quad \text{for } n = 1, \ldots, N$$

and will be used in the later model update. Unlike [23], which adopts a more complex formulation in model matching, i.e.,

$$l(t,\mathbf{x}) = \arg\min_{n=1,\ldots,N} \frac{|I_{t,\mathbf{x}} - \mu_{t-1,\mathbf{x},n}|}{\sigma_{t-1,\mathbf{x},n}} \tag{1}$$

a simple rule that selects the model of higher weight as the best match is used in Algorithm 1. The proposed *weight-based matching rule* prefers matching a pixel observation to the Gaussian model of background (with higher weight) other than those of foreground, if this observation falls in the scopes of multiple models. Using this rule not only saves computational costs but also fits the proposed rate control scheme better, as will be discussed in more detail later.

After model matching, we check if $M_{t,\mathbf{x},l(t,\mathbf{x})}$ is equal to 0, which implies no model matched. If so, a model replacement is performed to incorporate $I_{t,\mathbf{x}}$ into the GMM; otherwise, a model update is executed. In the replacement phase, the least weighted Gaussian model is replaced by the current intensity observation. In the update phase, the following three rules are applied:

$$\begin{aligned} \mu_{t,\mathbf{x},l(t,\mathbf{x})} &= \left(1 - \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\right) \mu_{t-1,\mathbf{x},l(t,\mathbf{x})} \\ &\quad + \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) I_{t,\mathbf{x}} && (2) \\ \sigma^2_{t,\mathbf{x},l(t,\mathbf{x})} &= \left(1 - \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha)\right) \sigma^2_{t-1,\mathbf{x},l(t,\mathbf{x})} \\ &\quad + \rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) \left(I_{t,\mathbf{x}} - \mu_{t,\mathbf{x},l(t,\mathbf{x})}\right)^2 && (3) \\ w_{t,\mathbf{x},n} &= \left(1 - \eta_{t,\mathbf{x}}(\beta)\right) w_{t-1,\mathbf{x},n} + \eta_{t,\mathbf{x}}(\beta) M_{t,\mathbf{x},n} && (4) \end{aligned}$$

where $\rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) \in \mathbb{R}$ denotes the learning rate for the Gaussian parameters μ and σ^2, and $\eta_{t,\mathbf{x}}(\beta) \in \mathbb{R}$ is a new learning rate introduced in this research for controlling the updating speed of the mixture weight w. Here, the two scalars α and β can be viewed as hyper-parameters over ρ and η for tuning their values. In [23], the learning rate ρ is defined as

$$\rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) \doteq \alpha \mathcal{N}\left(I_{t,\mathbf{x}}; \mu_{t-1,\mathbf{x},l(t,\mathbf{x})}, \sigma^2_{t-1,\mathbf{x},l(t,\mathbf{x})}\right) \tag{5}$$

while in [13] it is given by

$$\rho_{t,\mathbf{x},l(t,\mathbf{x})}(\alpha) \doteq \left(\frac{1-\alpha}{c_{t,\mathbf{x},l(t,\mathbf{x})}} + \alpha\right) \tag{6}$$

where $c_{t,\mathbf{x},n} = c_{t-1,\mathbf{x},n} + M_{t,\mathbf{x},n}$ and $c_{t=0,\mathbf{x},n} = 0, \forall \mathbf{x}, n$.[4] Although (6) may result in quicker convergence in Gaussian parameter learning [13], we still choose (5) in our implementation for experimental comparisons and put our emphasis on the control of the learning rate η for the mixture weight. In later experiments we will show that better performance can be achieved by controlling the learning rate η than by tuning the rate ρ. Also, as noted in [13], typical values of α are in $[0.1, 0.001]$ for both (5) and (6), yielding a wide range of convergence rates in Gaussian parameter estimation. Here we set $\alpha = 0.025$ as a default value for quick model learning.

In previous background modeling researches, e.g., [9], [13], [23], a naive setting for mixture weight update, i.e.,

$$w_{t,\mathbf{x},n} = (1-\alpha)\, w_{t-1,\mathbf{x},n} + \alpha M_{t,\mathbf{x},n}, \tag{7}$$

is adopted. The rule (7) can be viewed as a special case of the proposed weight update of (4) with $\eta_{t,\mathbf{x}} = \alpha$. In (7), all image pixels are confined to having an identical rate setting in mixture weight learning that scene changes can not be properly handled with respect to space and time. Instead, with our generalization that assigns individual learning rates for mixture weights to image pixels and adapts them over time, higher flexibility in regularizing background adaptation can be obtained. Note that the index n is not attached to $\eta_{t,\mathbf{x}}$ because the changing rates for the weights $w_{t,\mathbf{x},n}$s, $\forall n$, are designed to be consistent among the N Gaussian models of the same image pixel. Regarding the computation of $\eta_{t,\mathbf{x}}$, we link it to the high-level feedback of pixel types and describe the feedback control in Section II-B.

In the GMM, all of the scene changes, regardless of being foreground or background, are modeled by Gaussian distributions. To further distinguish these two classes, a foreground indicator $F_{t,\mathbf{x},n}$ for each Gaussian model is defined using the corresponding mixture weight as

$$F_{t,\mathbf{x},n} = \begin{cases} 0, & \text{if } w_{t,\mathbf{x},n} \geq T_w \\ 1, & \text{otherwise} \end{cases} \tag{8}$$

where $T_w \in \mathbb{R}$ is a preset parameter.[5] A binary foreground map can then be defined as a set $\mathcal{F}_t = \{F_{t,\mathbf{x},l(t,\mathbf{x})} \,|\, \forall \mathbf{x}\}$. In the original GMM formulations applying (7), more frequently matched Gaussian models will have larger weights and will be labeled as background. Nevertheless, stationary objects, e.g., abandoned packages or standing persons, that appear constantly within a restricted area should not always be absorbed into background for some applications. Rather, these objects may need to be stably highlighted as foreground and alarms should be triggered if necessary. By adaptively adjusting $\eta_{t,\mathbf{x}}$ in (4) based on object types, as will be discussed next, such demands may be fulfilled without resorting to complex versions of (8) for foreground and background separation.

B. Feedback Control

A flowchart of a general-purpose surveillance system is illustrated in Fig. 2, where five processing modules are presented in a sequential manner. In order to address the above issue associated with object types, the final results derived from the last module of *object type classification* is fed back to the first one of *background model maintenance* for further control of the learning rates. Rather than digging into the details of each module wherein different implementations can be accommodated by the proposed feedback scheme, we place the focus on the learning rate control for mixture weights in the following discussions.

[4] Interested readers can find the details of (6) in [13].

[5] The procedure of model sorting by the values of w/σ, as suggested in [23], is not applied here since it is more complex and may cause complications in foreground pixel labeling.

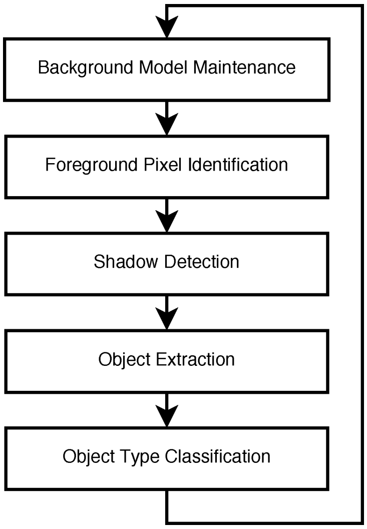

Background Model Maintenance
Foreground Pixel Identification
Shadow Detection
Object Extraction
Object Type Classification

Fig. 2. Flowchart of a general-purpose surveillance system. The first module of background model maintenance corresponds to the Algorithm 1. The second one of foreground pixel identification is implemented by using the mixture weight thresholding discussed in Section II-A. The third module can be realized by using an shadow detection algorithm described in [2]. For object extraction, we mark small (<4 × 4 pixels), isolated foreground regions as noises via morphological processing and group the rest foreground pixels into objects by connected component analysis. Regarding the object type classification and the feedback control on learning rates, they are presented in Section II-B.

In the proposed approach, we adopt different learning rate settings for four object types of background, shadow, still foreground and moving foreground, respectively. Based on the processing flow of Fig. 2, the object types of background, shadow, and foreground can be easily discriminated. To further classify the foreground type into still and moving ones, the object tracking algorithm presented in [2] is adopted to find the temporal associations among objects of time instances t and $t-1$. Then, the position displacements of tracked objects are thresholded for discrimination of still and moving types. Thus, an object type indicator for every pixel at time instance t can be defined as

$$O_{t,\mathbf{x}} = \begin{cases} 0, & \text{if } F_{t,\mathbf{x},l(t,\mathbf{x})} = 0, \text{ (Background)} \\ 1, & \text{if } F_{t,\mathbf{x},l(t,\mathbf{x})} = 1 \text{ and Type}(I_{t,\mathbf{x}}) = \text{Shadow} \\ 2, & \text{if } F_{t,\mathbf{x},l(t,\mathbf{x})} = 1 \text{ and Type}(I_{t,\mathbf{x}}) = \text{Still foreground} \\ 3, & \text{Otherwise(Moving foreground)} \end{cases}$$

and an object map can be denoted by $\mathcal{O}_t = \{O_{t,\mathbf{x}} \mid \forall \mathbf{x}\}$. Subsequently, the object map $\mathcal{O}_t$ is sent to the background maintenance module for the learning rate control for the next time instance. Obviously this is a delayed feedback control since the current learning rates are actually calculated from the previous classification results of pixel types. Yet, this kind of one-frame delay is acceptable in practice. In addition, for the feedback control being applied to the learning rate $\eta_{t,\mathbf{x}}$, but not to the mixture weights $w_{t,\mathbf{x},n}$ directly, dramatic changes in mixture weights as pixel type varies can be avoided. Stable foreground and background separation (via weight thresholding) can therefore be obtained.

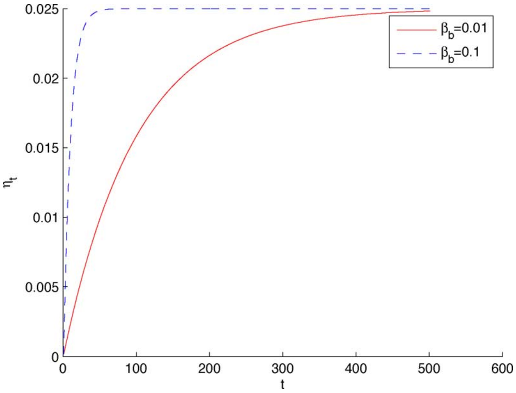


Fig. 3. Simulated changes of the learning rate η_t for a pixel being persistent background, given $\beta_b = 0.01$ (solid line) and $\beta_b = 0.1$ (dotted line), respectively. The initial learning rate $\eta_{t=0}$ is set to $1/6000$ and η_b is set to 0.025.

With the above notations, the learning rate $\eta_{t,\mathbf{x}}$ can now be specified by

$$\eta_{t,\mathbf{x}}(\boldsymbol{\beta}) = \begin{cases} (1-\beta_b)\eta_{t-1,\mathbf{x}} + \eta_b\beta_b, \\ \quad \text{if } O_{t-1,\mathbf{x}} = 0 \\ \beta_d \mathcal{N}\left(I_{t,\mathbf{x}}; \mu_{t-1,\mathbf{x},b(t,\mathbf{x})}, \sigma^2_{t-1,\mathbf{x},b(t,\mathbf{x})}\right), \\ \quad \text{if } O_{t-1,\mathbf{x}} = 1 \\ \beta_s, \quad \text{if } O_{t-1,\mathbf{x}} = 2 \\ \beta_m, \quad \text{if } O_{t-1,\mathbf{x}} = 3 \end{cases} \tag{9}$$

where η_b is a preset constant, the hyper-parameter $\boldsymbol{\beta} = [\beta_b\ \beta_d\ \beta_s\ \beta_m]^T \in \mathbb{R}^4$ is extended to a vector for regularizing the learning rate with respect to different pixel types, and the index of the most probable background model, $b(t,\mathbf{x})$, is defined by

$$b(t,\mathbf{x}) = \arg\max_{n=1,\ldots,N} w_{t,\mathbf{x},n}.$$

For a pixel of moving foreground ($O_{t-1,\mathbf{x}} = 3$), one may set $\beta_m \sim 0$ to suppress the adaptation of all moving objects into background, resulting in a very sensitive system to motions. In contrast, by setting β_m to a large value, which results in a quick increase of the weight of a Gaussian model for, say, a waving tree, a system will be more capable of tolerating background variations. On the other hand, for the type of still foreground, the larger the β_s is set, the quicker a stationary object will be merged into background. For the application of abandoned and missing object detection, a small β_s is preferred. Regarding the case of shadow type, we favor faster adaptation of fainter shadows into background, so $\mathcal{N}(\cdot)$ is used to estimate the similarity between the shadow intensity and the Gaussian model of the most probable background (indexed by $b_{t,\mathbf{x}}$). The corresponding learning rate is then set to the similarity measure multiplied by a regularization scalar β_d.

For a pixel of background type, i.e., $O_{t-1,\mathbf{x}} = 0$, its learning rate is designed to be gradually increased at a rate regularized by β_b, as formulated in (9). The learning rate for an image pixel being persistently identified as background will asymptotically approach η_b, as shown in Fig. 3. However, once this pixel position being occluded by shadows or moving objects, the respective learning rate will be reset to other value, e.g., β_m, that is

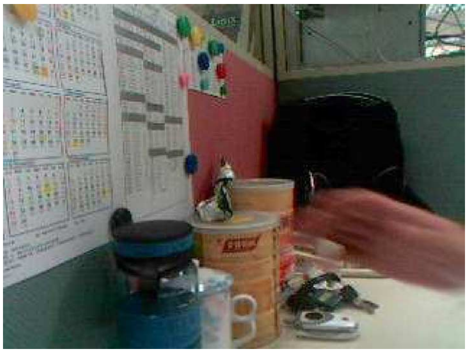


Fig. 4. Example of motion blur. The foreground and background boundaries of a moving hand may not be clearly distinguished, even by human visual inspection.

much smaller than it was used to be. This design helps preventing false inclusion of *afterimages* left by moving objects into background. When taking pictures of moving objects, their boundaries are often in blur. See Fig. 4 for an example. Some motion-blurred regions near object boundaries may be misclassified as background, resulting in afterimages. For an object hovering over a region, its afterimages appear frequently and will be quickly included into a background model. To alleviate such a problem, instead of setting the learning rate to a constant, i.e.,

$$\eta_{t,\mathbf{x}} = \eta_b, \qquad \text{if } O_{t-1,\mathbf{x}} = 0 \tag{10}$$

it is increased gradually for a pixel of background in the proposed approach. In Section III-A, benefits of adopting this *background-type rate control* will be demonstrated. Note that, in all our experiments, we set $\eta_b = \alpha$, $\beta_b = 0.01$, $\beta_d = 1/100$, $\beta_s = 1/900$ and $\beta_m = 1/6000$.

As discussed in [1] and [4], a major *problem with feedback control* for background modeling is that misclassifications of pixel type in the current frame will propagate to subsequent frames as the learning rates are determined by classification results. For instance, if a background pixel is misclassified as foreground, a false positive will persist at this pixel location for a long time due to the low learning rate setting for foreground pixel. Fortunately, this problem can be treated, if not cured, by the proposed framework of learning rate control.

Based on our observations, the *problem with feedback control* for background modeling can be effectively treated if the following two criteria fulfilled: 1) accurate estimation of a background model and 2) prevention of background adaptation to pixels of misclassified types. In the proposed approach, giving separate controls to the learning rates ρ and η meets the criterion 1). Up-to-date model estimations can hence be delivered by setting a large ρ, regardless of foreground classification results controlled by η. Even for the pixels of misclassified types, their Gaussian models can still be accurately estimated. Our experiments in Section III-E show that the accurate estimation of background models will help reducing persistent false positives of misclassified pixels.

Regarding the criterion 2), the background-type rate control in (9) is designed for it. With this control, false background adaptation to foreground motion blurs (a.k.a. afterimages) can be largely reduced, as will be shown in Section III-A. In addition, the weight-based matching rule is utilized in our approach to eliminate false positives even more. Although the matching rule seems to prefer the most-weighted Gaussian models of background for new pixel observations, its matching results are still trustworthy owing to our capability of deriving accurate Gaussian models. Advantages of adopting this weight-based matching rule will be further demonstrated in Section III-E.

C. Heuristic for Adaptation of Over-Quick Lighting Change

Surveillance systems often encounter challenges from lighting changes, especially for systems used in outdoor environments. While gradual and quick lighting variations can often be adapted by the GMM, some *over-quick* changes can not be caught via background model learning at reasonable learning rates. For instance, two examples of quick and over-quick lighting changes are given in Fig. 5. The image sequence $\mathbf{A}$ shown in Fig. 5(a)–(c) records a laboratory with a monitor displaying rolling interferences. In this indoor sequence, it takes about 3 seconds to increase the average intensity by 20%. This quick variation in image brightness can still be learned by the GMM, as will be demonstrated in Section III-A. In contrast, for a over-quick lighting change shown in Fig. 5(d)–(f), similar increases of image intensity are observed in less than one second for an outdoor environment. As will be shown in Section III-C, many false alarms in foreground detection are issued under such condition. Consequently, a heuristic based on frame difference is also developed to assist the GMM to cope with over-quick lighting changes.

The idea behind the heuristic is simple yet effective. While image intensity variation of over-quick lighting change may seem to be large among temporally distant image frames, it may be small between two consecutive frames if the frame rate of recording is high enough. The small and smooth change of image brightness between consecutive image frames provides a cue for eliminating false alarms in foreground detection for over-quick, but not *abrupt*,[6] lighting changes. For example, by thresholding the differences between corresponding pair of pixels, each from two consecutive frames, at a proper level, such false alarms can often be reduced.

Accordingly, the proposed heuristic consists the following formulations. First, the thresholding of intensity difference for every pixel pair is performed by

$$D_{t,\mathbf{x}} = \begin{cases} 1, & |I_{t,\mathbf{x}} - I_{t-1,\mathbf{x}}| > T_d \\ 0, & \text{otherwise} \end{cases}$$

where $T_d(= 10)$ is a given threshold. Thus, a frame difference map $\mathcal{D}_t = \{D_{t,\mathbf{x}} \,|\, \forall \mathbf{x}\}$ can be derived. By combining both the frame difference map $\mathcal{D}_t$ and the foreground map $\mathcal{F}_t$ via

$$\mathcal{F}'_t = \mathcal{F}_t \quad \texttt{AND} \quad (\mathcal{F}'_{t-1}\, \texttt{OR}\, \mathcal{D}_t) \tag{11}$$

a new foreground map $\mathcal{F}'_t$ being less affected by lighting changes can now be obtained. Note that the OR operation in (11) is utilized for temporal accumulation of foreground regions, which is useful for detecting objects in slow motion. The map $\mathcal{F}'_t$ is then used to replace $\mathcal{F}_t$ as a new output of the second module in Fig. 1. Regarding the lighting change areas where

[6] Abrupt changes in background are regarded as salient deviations between two consecutive image frames, due to, e.g., light on/off.

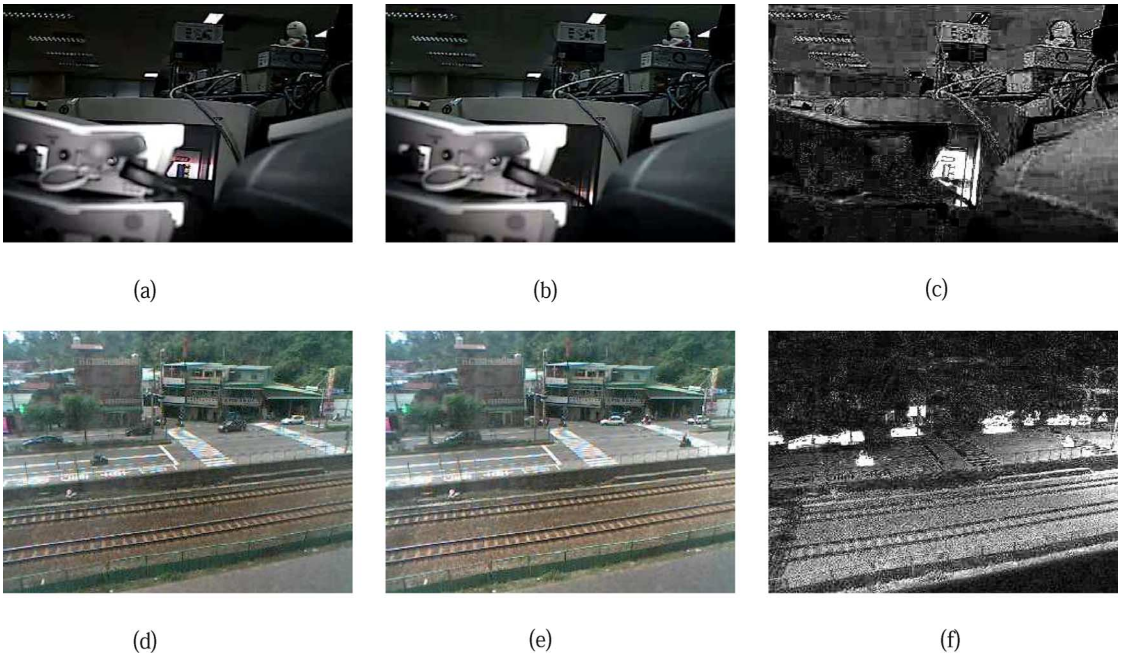


Fig. 5. Examples of (i) quick and (ii) over-quick lighting changes. (a)–(c) Two images in Seq. $\mathbf{A}$ (recorded at 20 fps) and their difference for (i). (d)–(f) Two images in Seq. $\mathbf{B}$ (recorded at 15 fps) and their difference for (ii). (a) $I^{\mathbf{A}}_{t-60}$, (b) $I^{\mathbf{A}}_{t}(t = 183)$, (c) $|I^{\mathbf{A}}_{t} - I^{\mathbf{A}}_{t-60}| * 6$, (d) $I^{\mathbf{B}}_{t-12}$, (e) $I^{\mathbf{B}}_{t}(t = 548)$, (f) $|I^{\mathbf{B}}_{t} - I^{\mathbf{B}}_{t-12}| * 6$.

$\mathcal{F}'_t - \mathcal{F}_t \neq 0$, they are relabelled as background and will be quickly learned by the GMM via (9). False alarms caused by over-quick lighting changes will hence be reduced. Based on our experiments shown in Section III-C, the system robustness to lighting changes will be increased without losing the sensitivity in detecting significant foreground motions.

Because this heuristic is developed to improve the tolerance of our model to speedy lighting changes without altering the background estimation results much, the threshold value is usually limited by $10 \leq T_d \leq 20$. Image differences larger than 20 between two consecutive image frames, which might be perceived by sensitive human eyes, are considered as abrupt changes. Owing to the accumulating formulation in (11), large lighting changes between two distant image frames can still be handled using small T_d for most cases.

III. Experimental Results

Several real videos are used to test the effectiveness of the proposed rate control scheme. In Section III-A, comparisons among different learning rate controls proposed by the original GMM [23], its variant [13] and this research are presented[7] using two image sequences with lighting changes, missing objects and waving hands. While the first scenario of lighting changes should be quickly adapted into background, the other two should not. All these scenarios can be properly handled by the proposed approach but not by those of [23] and [13]. In Section III-B, the effects of tuning the parameter $\boldsymbol{\beta}$ are discussed. Next in Section III-C, by using a third image sequence as a benchmark, the superiority of the proposed heuristic for adaptation of over-quick lighting change is demonstrated. In Section III-D, quantitative evaluations of selected approaches with respect to different α values are presented. In Section III-E, an example of fountain spurt is used to demonstrate our treatments of the *problem with feedback control* for background modeling. Finally, additional experimental results are given to show the effectiveness of the proposed approach for the scenes of *waving water* and *crowded entrance*.

A. Regularized Background Adaptation

In the first experiment for the adaptation of quick lighting changes, we use Seq. $\mathbf{A}$ previously illustrated in Fig. 5 as a benchmark. The foreground detection results and the learned background models, up to the image frame $I^{\mathbf{A}}_t$, obtained from different approaches are shown in Fig. 6 for two different learning rates. For visual comparisons of the learned background models, a definition of background map $\mathcal{B}_t = \{\mu_{t,\mathbf{x},b(t,\mathbf{x})} \,|\, \forall \mathbf{x}\}$ is adopted, and the derived background maps are drawn in the middle row of Fig. 6. In Fig. 6(a) and (b), false positives of foreground detection are observed by using $\alpha = 0.01$ for the approaches of [23] and [13]. As shown in Fig. 6(d) and (e), all of the false positives can be eliminated by giving a higher learning rate with $\alpha = 0.025$ while only the rolling interferences on a monitor are marked as foreground. On the other hand, correct foreground detection results are obtained in Fig. 6(c) and (f) by the proposed approach (with the heuristic of (11) applied) for both rate settings.

In the previous experiment, $\alpha = 0.025$ can be regarded as a proper setting for adaptation of quick lighting change. However, if the same setting is used for Seq. $\mathbf{C}$, defects of foreground detection will appear for approaches of [23] and [13]. (Because the foreground detection results of [23] and [13] in this experiment are almost the same, only those of [13] are shown in Fig. 7 for brevity.) As shown in Fig. 7(a), a cellular phone on a desk is taken away. Usually, a missing personal property should be marked as foreground and trigger an alarm. However, such an

[7] For experimental evaluations, we apply the conventional matching rule (1) to [13] and [23], and use the same labeling rule (8) with $T_w = 0.24$ to all the methods to segment foreground regions.

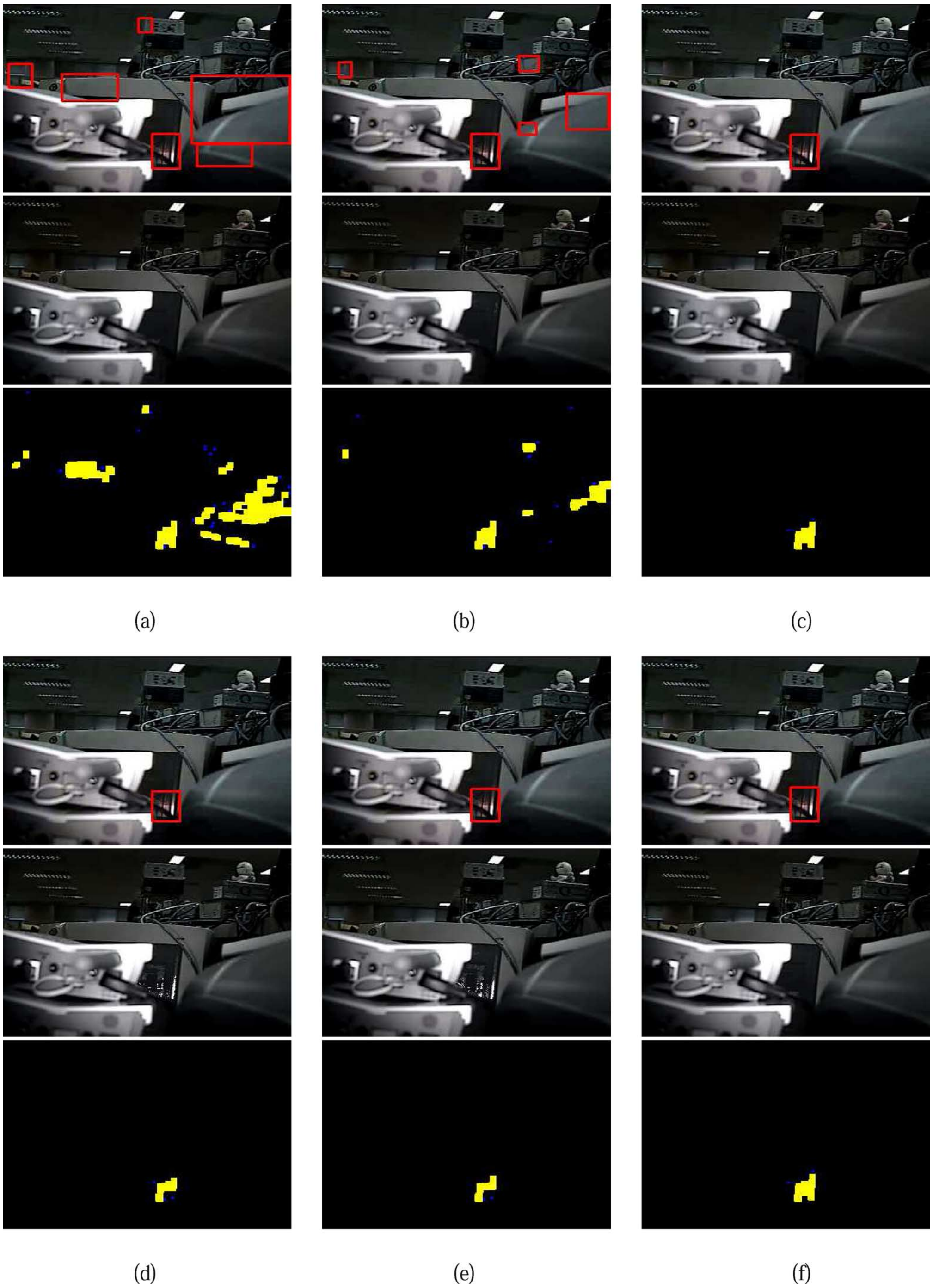


Fig. 6. Comparisons of background adaptation to quick lighting changes using Seq. $\mathbf{A}$. Top row: foreground detection results for $I_t^{\mathbf{A}}$. Middle row: computed background maps $\mathcal{B}_t$s. Bottom row: derived foreground maps. In the foreground maps, the regions in blue denote shadows and noises. (a), (b), (c) The results of [23], [13], and our approach, respectively, with $\alpha = 0.010$. (d), (e), (f) The results of [23], [13], and our approach, respectively, with $\alpha = 0.025$. (a) Results of [23] $(\alpha = 0.010)$. (b) Results of [13] $(\alpha = 0.010)$. (c) Our results $(\alpha = 0.010)$, (d) Results of [23] $(\alpha = 0.025)$. (e) Results of [13] $(\alpha = 0.025)$. (f) Our results $(\alpha = 0.025)$.

abnormal event can not be stably detected with $\alpha = 0.025$. The quick adaption of the uncovered region into background happens in about one second, as shown in Fig. 7(b), leaving no evidence of the missing cellular phone. Similarly, hand waving in front of the camera is soon adapted into background as well, as shown in Fig. 7(c), causing the hand regions only partially detected. In contrast, the above two scenarios can be properly handled by the proposed approach with the same parameter setting $(\alpha = 0.025)$, as shown in Fig. 7(d)–(f). Thank to the regularization of the learning rate η, quick lighting changes, missing objects and periodic motions can all be modeled decently in an unified framework.

Advantages of the proposed background-type rate control are also demonstrated, using Seq. $\mathbf{C}$, in Fig. 8 wherein background modeling results are obtained with and without the background-type rate control. By replacing the gradual increase of back-

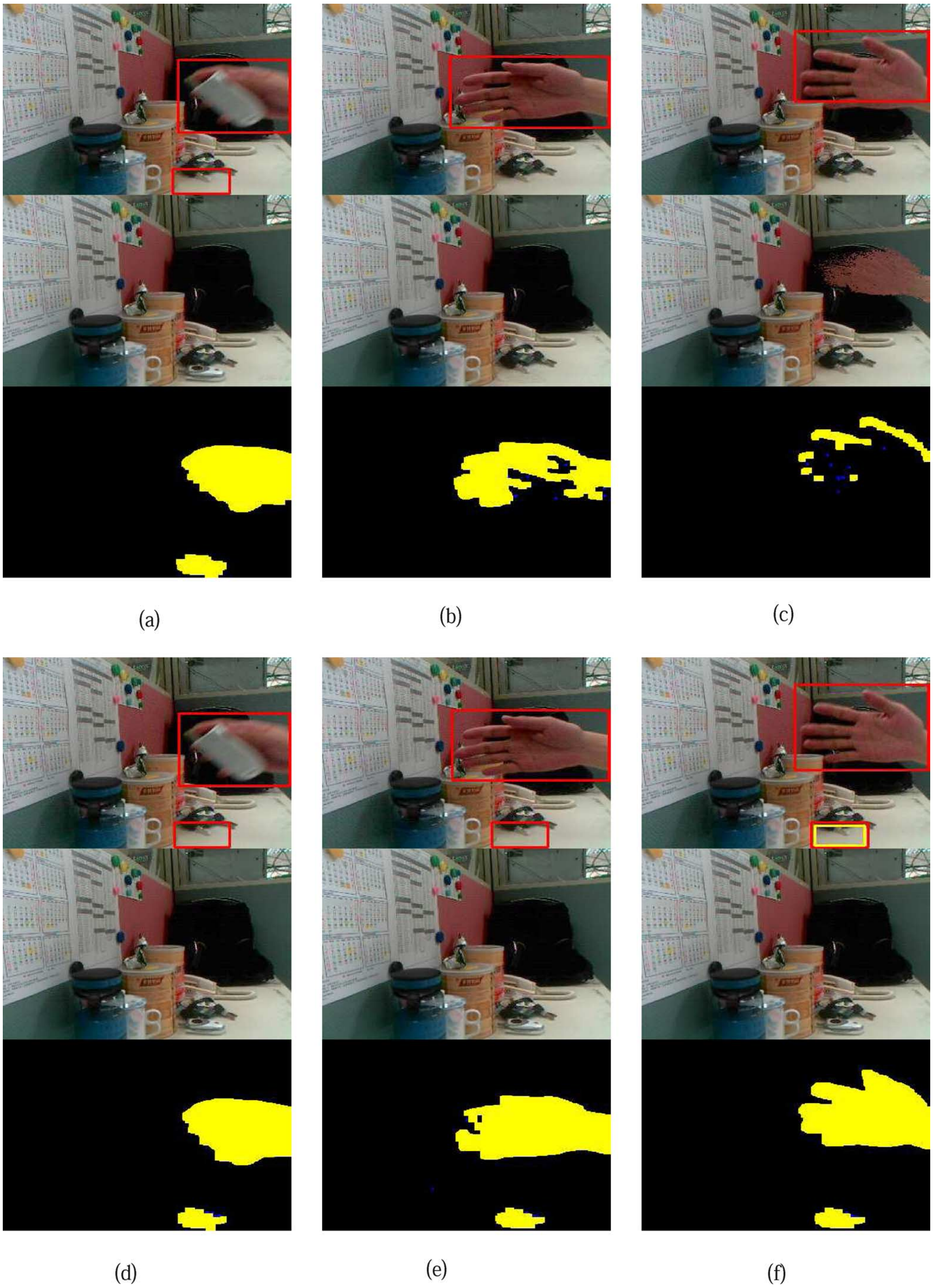


Fig. 7. Comparisons of background modeling for missing object and waving hand using Seq. $\mathbf{C}$. Top row: foreground detection results. Middle row: computed background maps. Bottom row: derived foreground maps. (a), (b), (c) The results for $I_{85}^{\mathbf{C}}$, $I_{110}^{\mathbf{C}}$, and $I_{150}^{\mathbf{C}}$, respectively, using [13] with $\alpha = 0.025$. (d), (e), (f) The results for $I_{85}^{\mathbf{C}}$, $I_{110}^{\mathbf{C}}$, and $I_{150}^{\mathbf{C}}$, respectively, using our approach with $\alpha = 0.025$. In (f), the cellular phone taken away is identified as a missing object and highlighted by a yellow box. (a) Results of [13] for $I_{85}^{\mathbf{C}}$. (b) Results of [13] for $I_{110}^{\mathbf{C}}$. (c) Results of [13] for $I_{150}^{\mathbf{C}}$. (d) Our results for $I_{85}^{\mathbf{C}}$. (e) Our results for $I_{110}^{\mathbf{C}}$. (f) Our results for $I_{150}^{\mathbf{C}}$.

ground learning rate in (9) with a constant setting of (10), as can be seen in Fig. 8(a), the afterimages induced by the waving hand are included into the background model (the second row) and the resulted segmentation of foreground regions is incomplete (the third row). In Fig. 8(b), as the hand moving out of the scene, the incorrect background model continues to gives false positives in foreground detection for a period of time. On the other hand, such defects can be effectively reduced by using the proposed rate control for background pixels, as shown in Fig. 8(c) and (d).

B. Parameter Tuning of $\boldsymbol{\beta}$

As tuning the hyper-parameter $\boldsymbol{\beta} = [\beta_b \ \beta_d \ \beta_s \ \beta_m]^T$, the effect is more on time span of background adaptation than on the accuracy of background modeling. Specifically, varying β_s

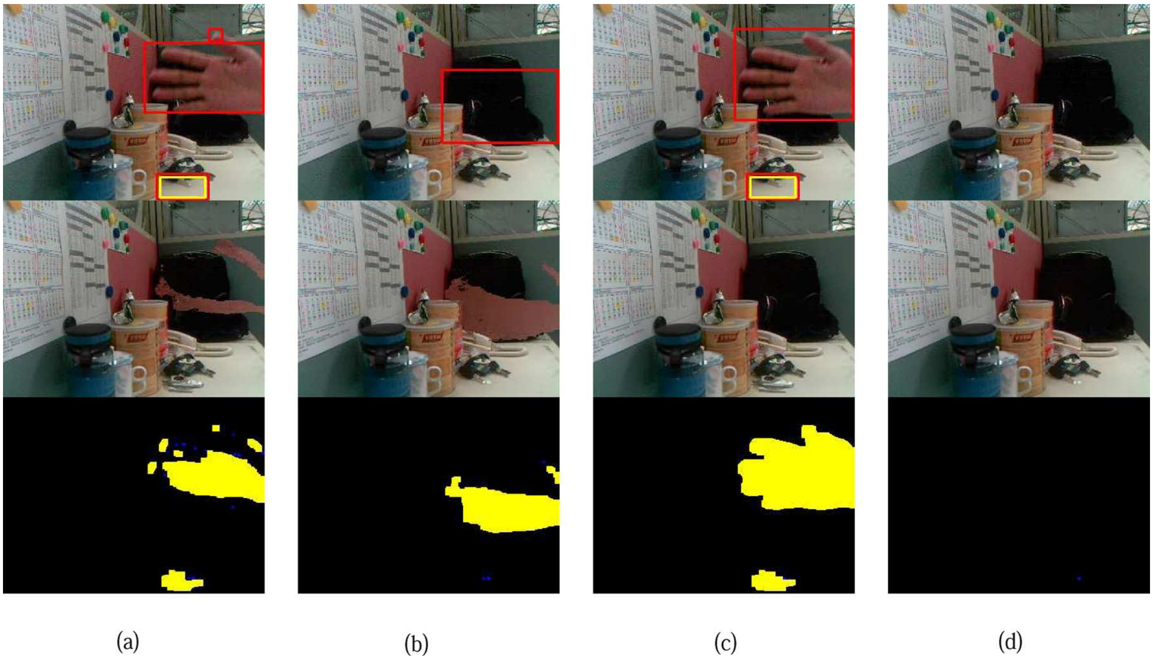


Fig. 8. Comparisons of background modeling results obtained without and with using the background-type rate control (BTRC). Top row: foreground detection results. Middle row: computed background maps. Bottom row: derived foreground maps. (a), (b) The results derived by replacing the first equation of (9) with (10). (c), (d) The results derived by (9). (a) Without BTRC for $I^{\mathbf{C}}_{365}$. (b) Without BTRC for $I^{\mathbf{C}}_{810}$. (c) With BTRC for $I^{\mathbf{C}}_{365}$. (d) With BTRC for $I^{\mathbf{C}}_{810}$.

changes the time span for a still object to be merged into a background model, if no interrupt occurs. The number of required image frames to adapt a pixel of still type into background can be estimated by

$$\arg\min_{t} w_t > T_w \quad \text{subject to } w_t = (1-\beta_s)w_{t-1} + \beta_s.$$

For example, given $\beta_s = 1/900$ and $w_0 = 0.01$ as an initial value, at least $t = 239$ image frames are required to complete the background adaptation of a still-type pixel. For Seq. $\mathbf{C}$ shown in Fig. 8, it takes about 288 frames to replace regions of the missing cellular phone with newly revealed scenes in the background model, just a little longer than predicted. Regarding the default setting of $\beta_m = 1/6000$, at least $t = 1588$ image frames are needed for a pixel being continuously occupied by the same hovering object to be adapted into background. This number of image frames roughly matches the testing example shown in Section III-E where all the regions of a fountain spurt are adapted into background in about 2000 image frames.

Similarly, tuning β_b alters the time span of avoiding afterimages to be incorporated into a background model. Taking Seq. $\mathbf{C}$ as an benchmark, the numbers of image frames having no afterimage in background models under different β_b's are summarized in Table I. Here, setting β_b to 0.05 or less gives no obvious defects in the estimated background models throughout the sequence. On the other hand, increasing β_b and β_m may be needed for scenarios with large periodic motions, e.g., shaking tree branches and moving tides.

Regarding β_d, it is tuned to slightly defer the adaptation of shadows that are usually casted by foreground objects into a background model. Thus, the product $\beta_d \mathcal{N}(I_{t,\mathbf{x}}; \mu_{t-1,\mathbf{x},b(t,\mathbf{x})}, \sigma^2_{t-1,\mathbf{x},b(t,\mathbf{x})})$ should be kept below β_b. In addition, if the product is less than β_s, it will be reset to β_s in our implementation instead, to adapt static and frequently seen shadows into background. To sum up, via proper tuning of $\boldsymbol{\beta}$, the required time spans for adapting pixel of different types into background can be easily and accurately controlled for various applications.

TABLE I
THE # OF IMAGE FRAMES RESISTING BACKGROUND ADAPTATION TO AFTERIMAGES W.R.T. β_b'S

β_b	0.01	0.05	0.1	0.5	1
# of Frames	> 700	> 700	~ 430	~ 175	~ 165

C. Adaptation of Over-Quick Lighting Change

Fig. 9 shows a scene experiencing very quick sunshine changes. The resultant over-quick changes in background can not be adapted in time by the GMM framework, even by setting high learning rates, as shown in Fig. 9. By utilizing the proposed heuristic, with T_d set to 10, for adaptation of over-quick lighting change, almost all the false positives resulted from sunshine changes are eliminated in the entire testing sequence. Nevertheless, a few sides-effects are also observed. Fig. 9(d) gives an example that a small motorcycle whose colors are similar to the background scene is misidentified as noises (marked in blue), for some parts of this object are deleted by frame difference. Through examination of these results, one can easily see that, overall, adopting such a heuristic actually brings in more benefits than drawbacks. Further quantitative evaluations, as will be presented later, also support this observation. Many false positives in foreground detection can thus be reduced while only limited false negatives are induced. Besides, large, significant motions will not be ignored by using

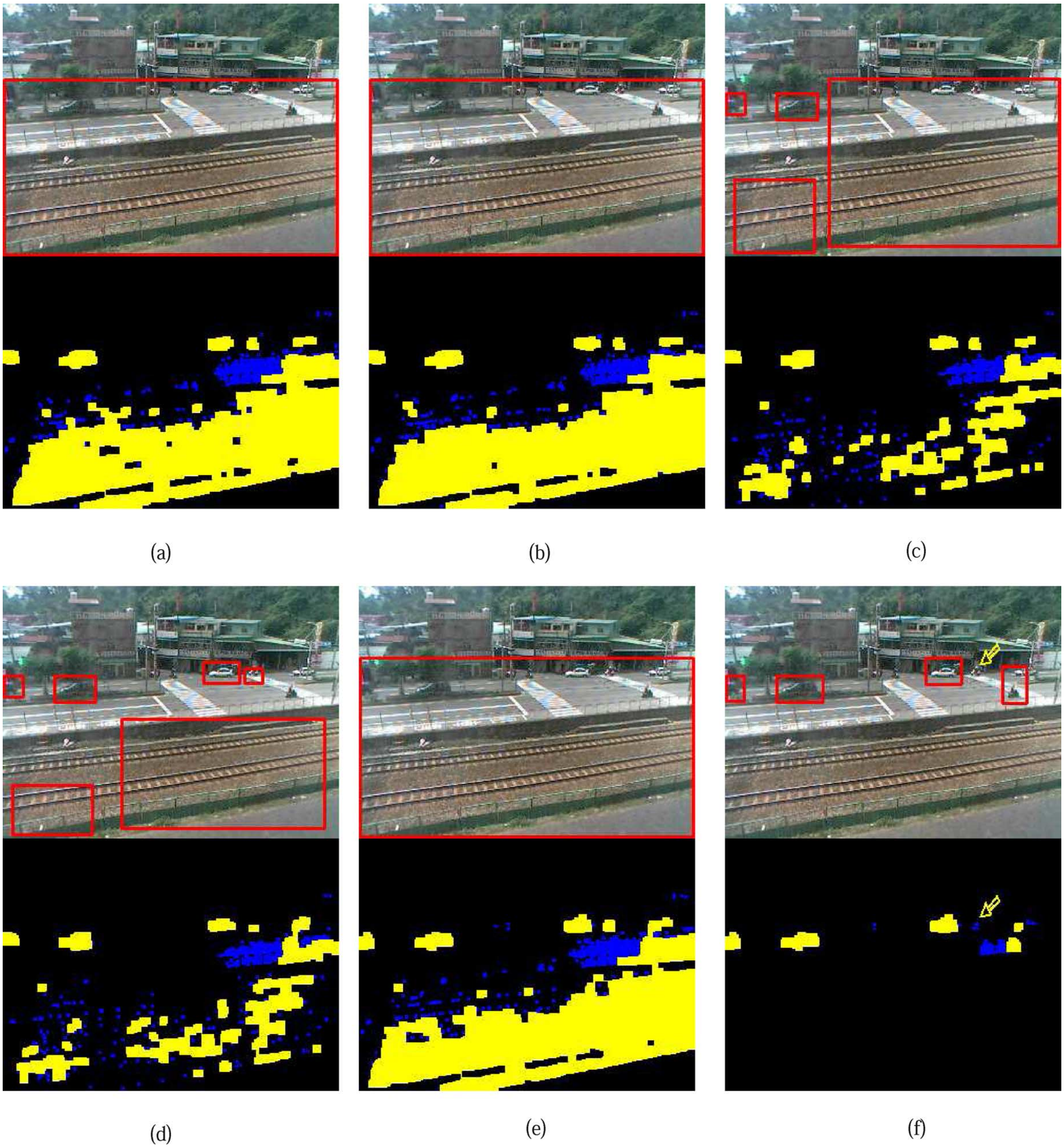


Fig. 9. Comparisons of background adaption to over-quick lighting change using Seq. $\mathbf{B}$. The foreground detection results for $I_t^{\mathbf{B}}$ are illustrated. (a), (b) The results of [23] and [13], respectively, with $\alpha = 0.025$. (c), (d) The results of [23] and [13], respectively, with $\alpha = 0.050$. (e) The results of the proposed approach without using the heuristic for adaptation of over-quick lighting change. (f) The results of the proposed approach. The yellow arrows mark the undetected foreground regions of a small motorcycle. (a) Results of [23] $(\alpha = 0.025)$. (b) Results of [13] $(\alpha = 0.025)$. (c) Results of [23] $(\alpha = 0.050)$. (d) Results of [13] $(\alpha = 0.050)$. (e) Our results w/o the heuristic $(\alpha = 0.025)$. (f) Our results $(\alpha = 0.025)$.

this heuristic due to its design of foreground map accumulation via the OR operation in (11).

D. Quantitative Evaluations

In the quantitative comparisons among [13], [23], and our approach without/with the heuristic of (11), Seq. $\mathbf{B}$ is used as a benchmark for it is a real and challenging sequence. To construct the ground-truth data, we write a program to segment possible foreground regions of Seq. $\mathbf{B}$ with high sensitivity. Subsequently, 32 representative image frames are selected by visual inspection, and with their segmentation results refined manually. Note that all the vehicles in the scene, no matter in motion or resting, are marked as foreground in this evaluation. Snapshots of the ground-truth images are given in Fig. 10.

The statistical plots in Fig. 11 are generated by applying different α values to all the compared methods. Also, two T_d settings for our approach are included in the comparison. Results in Fig. 11 show that, with $T_d = 10$, the proposed approach constantly achieves low false positive rates $(<1\%)$ while keeping high detection accuracy $(>90\%)$ for all αs. If the heuristic is not used, then $\alpha = 0.2$ can be chosen for our approach to both

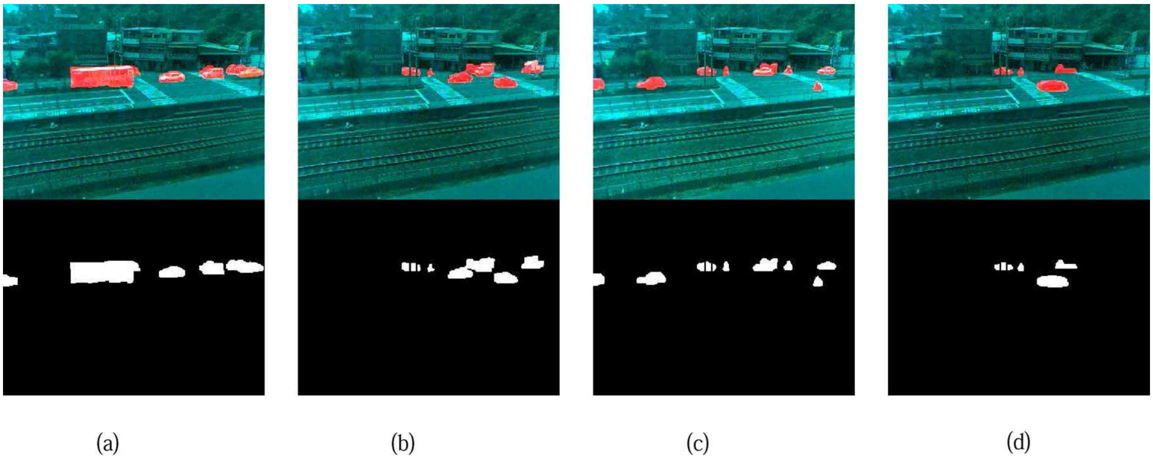

(a) (b) (c) (d)

Fig. 10. Snapshots of the ground-truth images for Seq. $\mathbf{B}$. (a) $I^{\mathbf{B}}_{450}$. (b) $I^{\mathbf{B}}_{500}$. (c) $I^{\mathbf{B}}_{548}$. (d) $I^{\mathbf{B}}_{600}$.

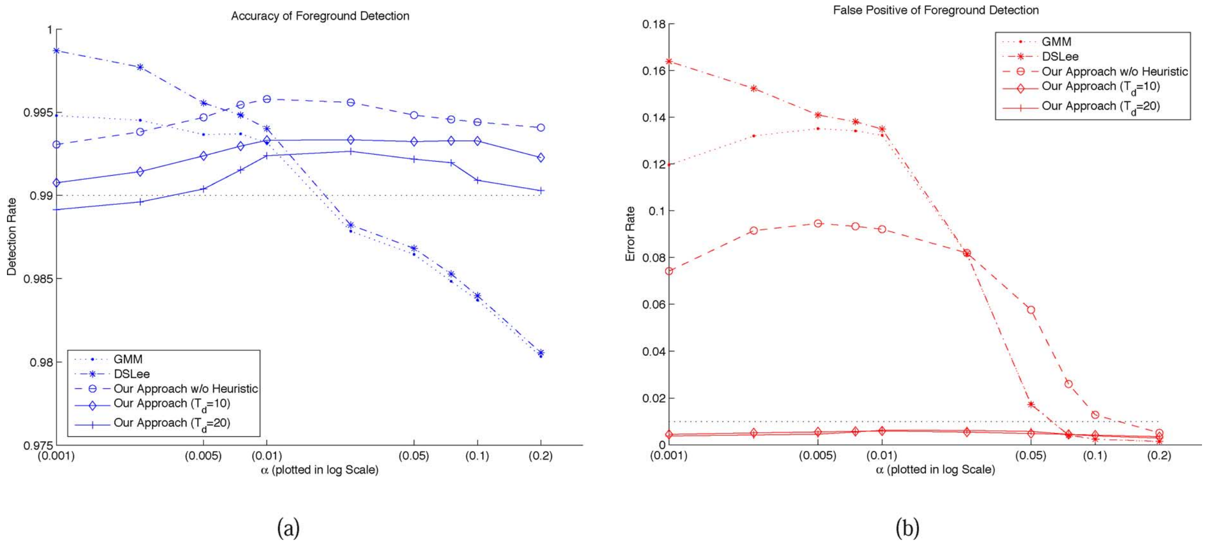


Fig. 11. Quantitative comparisons of [13] (DSLee), [23] (GMM), our approach without the heuristic, and our approach with $T_d = 10$ and $T_d = 20$ under different α settings using the 32 ground-truth images of Seq. $\mathbf{B}$. Here, the values of 0.0010, 0.0025, 0.0050, 0.0075, 0.0100, 0.0250, 0.0500, 0.0750, 0.1000, and 0.2000 are set to α to generate the curves. (a) Comparisons of foreground detection rates. The detection level of 99% is marked for reference. (b) Comparisons of false positives rates in foreground detection. The false positive level of 1% is marked for reference.

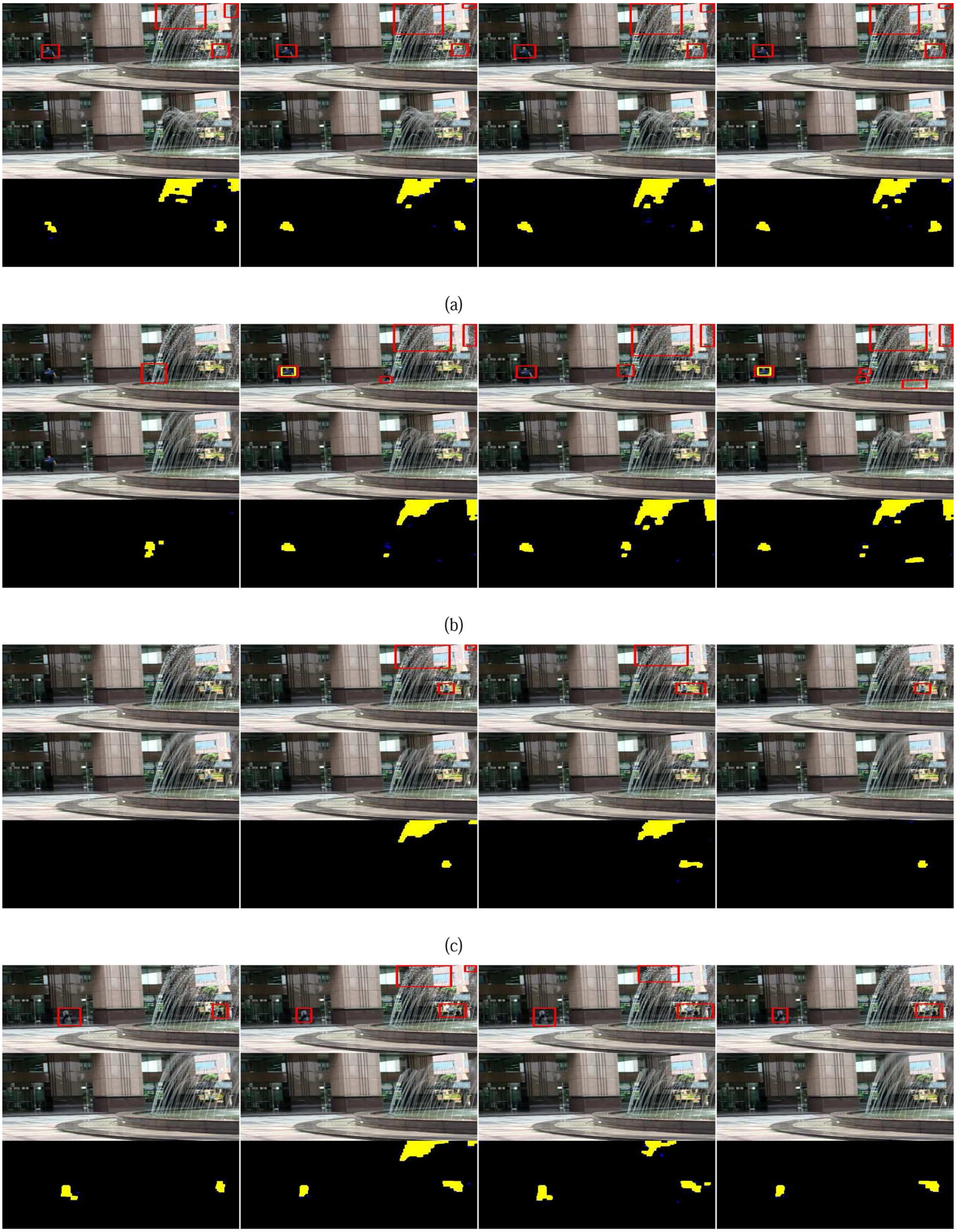
catch the over-quick lighting changes and maintain high detection accuracy. As for the methods of [13] and [23], finding a reasonably good parameter setting seems not possible for this case.

Although our approaches (with and without the heuristic) do not give the highest detection rate, they both have a feature of delivering stable detection results under various α settings, mainly owing to our independent controls of the two types of learning rates. Moreover, through examining the false positive rates with respect to different αs, we choose to bring the heuristic into our approach as a default practice since doing so will almost always give low false alarms. Note that, based on the evaluations, $\alpha = 0.01$ and $\alpha = 0.025$ can be suggested as default values for the heuristic because these values give slightly better detection accuracy.

To verify our argument that adjusting $\boldsymbol{\beta}$ does not affect the background modeling performance much, a quantitative evaluation is conducted by varying β_b to 0.001, 0.005, 0.010, 0.050, 0.100, and 0.500, with the other parameters fixed to the default values. While the detection and false positive rates for $\beta_b = 0.010$ (the default setting) are 99.3347% and 0.5420%, respectively, similar performance indexes for the other β_bs are all within $(99.3347 \pm 0.0274)\%$ and $(0.5420 \pm 0.0199)\%$, respectively, which supports our argument.

E. Adaptation of Scene Change

In Section II-B, the *problem with feedback control* and possible solutions are discussed. An example illustrating such a problem is given in Fig. 12, where a fountain suddenly spurting high causes a bunch of false positives in foreground detection. The first column of Fig. 12 shows such dramatic changes of background scene may be adapted too quickly (in about 100 image frames) by [13] if a high learning rate ($\alpha = 0.025$) is used. On the contrary, as shown in the second column, these false positives last for a very long time (>2440 image frames) if a naive feedback control by setting $\rho(\alpha) = \eta(\boldsymbol{\beta})$



(a)

(b)

(c)

(d)

Fig. 12. Comparisons of scene change adaptation among [13] (the first column), feedback control with $\rho(\alpha) = \eta(\boldsymbol{\beta})$ (the second column), our approach with (1) (the third column) and our approach with the weight-based model matching rule (the 4th column). (a) I_{2900}, (b) I_{3000}, (c) I_{4885}, and (d) I_{5340}.

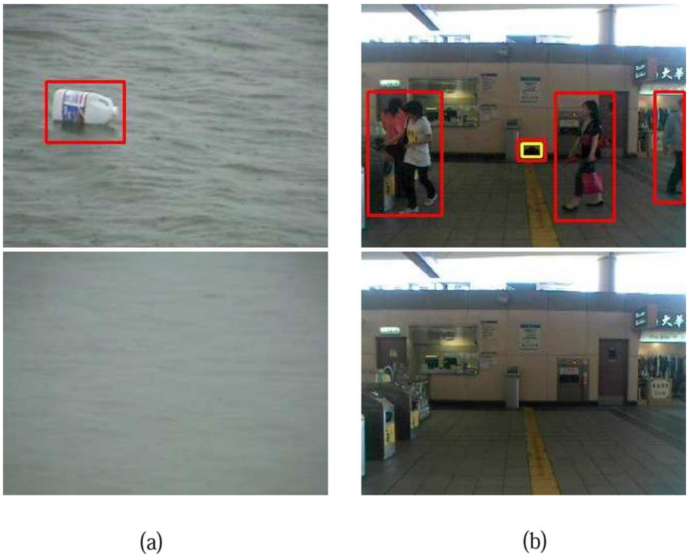


Fig. 13. Foreground detection results for the scenes of (a) waving water [28] and (b) crowded entrance. The yellow box in (b) marks an abandoned bag.

is used. The identical modeling of ρ and η, together with the feedback controls, makes a system behave as what the problem describes. However, as depicted in the third column of Fig. 12(c) and (d), quicker adaptation of false positives into background can be achieved by separating the control of ρ from that of η with the conventional model matching rule of (1). Finally, as shown in the forth column, the false positives resulted from scene changes can be completely eliminated in about 2000 image frames (equivalent to about 1.11 min for a 30-fps video) by combining the proposed rate control scheme with the weight-based model matching rule.

F. Other Results

Additional experiments for the scenes of waving water[8] and crowded entrance are demonstrated in Fig. 13. In Fig. 13(a), a floating bottle on the waving water can be successfully detected by the proposed approach. In the crowded entrance sequence shown in Fig. 13(b), a black bag left by a passenger is stably detected as a foreign object in a busy scene. These experiments show the effectiveness of the proposed scheme of learning rate control for the surveillance applications associated with complex scenes.

IV. Conclusion

In background model learning, maintaining a balance between robustness to background variations and sensitivity to foreground changes has long been regarded as a hard problem. In this work, via the clarification of the different roles of different learning rates for the GMM and by adopting the proposed rate control scheme, the tradeoff between model robustness and sensitivity can be effectively regularized. Experimental results show that, with careful tuning of the learning rates for mixture weights, robustness to quick variations in background as well as sensitivity to abnormal changes in foreground can be achieved simultaneously for several surveillance scenarios. In addition, a heuristic for adaptation of over-quick lighting change is proposed and verified in this work. With the help of this heuristic, large lighting changes occurring in very short time intervals, e.g., within one second, can be absorbed into background.

Our design of the learning rate control roots in the high-level feedback of pixel types identified by a surveillance system. Although, in our current setting, only a limited amount of pixel types are computed for rate control, noticeable improvements in foreground detection over conventional GMM approaches are already observable. Owing to the simplicity and scalability of the proposed scheme, more complex scenarios may be handled with more high-level information incorporated. For example, region-level classification results of skin/nonskin, face/nonface and human/nonhuman can be fed back to the pixel-level rate control of η in background modeling to increase model sensitivity to these objects. Also, proper learning rate settings of α and β for pixels of high spatio-temporal gradients may be worth an investigation. Another interesting direction is to apply biological cues, e.g., discriminant saliency [15] between center and surround, to increase the adaptation rates for background pixels of highly dynamic background scenes that are often misclassified as foreground one.

Acknowledgment

The authors would like to thank the anonymous reviewers for their fruitful comments. H.-H. Lin is grateful to C.-W. Huang of the QNAP System Inc. for his valuable suggestions throughout this work.

References

[1] T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, and A. Erkan, "Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets," in *Proc. 2nd IEEE Workshop on Visual Surveillance*, 1999, pp. 48–55.

[2] H.-T. Chen, H.-H. Lin, and T.-L. Liu, "Multi-object tracking using dynamical graph matching," in *Proc. IEEE Conf. Compute. Vis. Pattern Recogn.*, 2001, vol. 2, pp. 210–217.

[3] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.

[4] A. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric background model for background subtraction," in *Proc. Eur. Conf. Computer Vision*, 2000, vol. 2, pp. 751–767.

[5] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proc. Conf. Uncertainty in Artif. Intell.*, 1997, pp. 175–181.

[6] X. Gao, T. E. Boult, F. Coetzee, and V. Ramesh, "Error analysis of background adaption," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2000, vol. 1, pp. 503–510.

[7] W. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 1998, pp. 22–29.

[8] 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.

[9] M. Harville, "A framework for high-level feedback to adaptive, per-pixel, mixture-of-Gaussian background models," in *Proc. Eur. Conf. Computer Vision*, 2002, vol. 3, pp. 543–560.

[10] E. Hayman and J.-O. Eklundh, "Statistical background subtraction for a mobile observer," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, pp. 67–74.

[11] M. Heikkilä and M. Pietikäinen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 657–662, Apr. 2006.

[12] T. Ko, S. Soatto, and D. Estrin, "Background subtraction on distributions," in *Proc. Eur. Conf. Comput. Vis.*, 2008, vol. 3, pp. 276–289.

[8] The image sequence of waving water is from [28].

[13] D.-S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 827–832, May 2005.

[14] L. Li, W. Huang, I. Y.-H. Gu, and Q. Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1459–1472, Nov. 2004.

[15] V. Mahadevan and N. Vasconcelos, "Background subtraction in highly dynamic scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2008, pp. 1–6.

[16] S. J. McKenna, S. Jabri, Z. Duric, A. Rosenfeld, and H. Wechsler, "Tracking groups of people," *Comput. Vis. Image Understanding*, vol. 80, no. 1, pp. 42–56, Oct. 2000.

[17] A. Mittal and D. Huttenlocher, "Site modeling for wide area surveillance and image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2000, vol. 2, pp. 160–167.

[18] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh, "Background modeling and subtraction of dynamic scenes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, pp. 1305–1312.

[19] C. Ridder, O. Munkelt, and H. Kirchner, "Adaptive background estimation and foreground detection using Kalman-filtering," in *Proc. Int. Conf. Recent Advances in Mechatron.*, 1995, pp. 193–199.

[20] J. Rittscher, J. Kato, S. Joga, and A. Blake, "A probabilistic background model for tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2000, vol. 2, pp. 336–350.

[21] M. Seki, T. Wada, H. Fujiwara, and K. Sumi, "Background subtraction based on cooccurrence of image variations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2003, vol. 2, pp. 65–72.

[22] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1778–1792, Nov. 2005.

[23] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 1999, vol. 2, pp. 246–252.

[24] Y.-L. Tian, M. Lu, and A. Hampapur, "Robust and efficient foreground analysis for real-time video surveillance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, 2005, vol. 1, pp. 1182–1187.

[25] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, vol. 1, pp. 255–261.

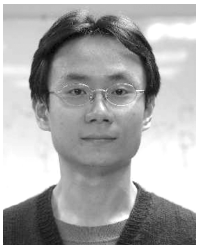
[26] C. R. Wren, A. Azarbayejani, T. J. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, Jul. 1997.

[27] H. Yang, Y. Tan, J. Tian, and J. Liu, "Accurate dynamic scene model for moving object detection," in *Proc. IEEE Int. Conf. Image Process.*, 2007, vol. 6, pp. 157–160.

[28] J. Zhong and S. Sclaroff, "Segmenting foreground objects from a dynamic textured background via a robust Kalman filter," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, vol. 1, pp. 44–50.

[29] Q. Zhu, S. Avidan, and K.-T. Cheng, "Learning a sparse, corner-based representation for time-varying background modeling," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, vol. 1, pp. 678–685.

[30] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proc. Int. Conf. Pattern Recogn.*, 2004, vol. 2, pp. 28–31.

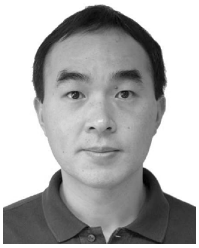


Horng-Horng Lin (S'10) received the B.S. degree in computer science and information engineering and M.S. degree in computer and information science from National Chiao Tung University, Hsinchu, Taiwan, in 1997 and 1999, respectively, where he is currently working toward the Ph.D. degree in computer science.

Between 1999 and 2004, he served the National Defense Substitute Service on Enterprise as a Research Assistant with the Institute of Information Science in Academia Sinica, Taipei, Taiwan. He joined QNAP System Inc., New Taipei City, Taiwan, in 2009. His research interests include computer vision and pattern recognition.

Jen-Hui Chuang (SM'06) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1980, the M.S. degree in electrical and computer engineering from the University of California, Santa Barbara, in 1983, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, in 1991.

Since 1991, he has been on the faculty of the Department of Computer Science at National Chiao Tung University, Hsinchu, Taiwan, where he is currently a Professor. His research interests include robotics, computer vision, 3-D modeling, and image processing.



Tyng-Luh Liu (M'99) received the B.S. degree in applied mathematics from National Chengchi University, Taipei, Taiwan, in 1986, and the Ph.D. degree in computer science from New York University, New York, in 1997.

He is a Research Fellow with the Institute of Information Science, Academia Sinica, Taipei, Taiwan. His research interests include computer vision, pattern recognition, and machine learning.

Dr. Liu was the recipient of the Junior Research Investigators Award from Academia Sinica in 2006.