Information Sciences 483 (2019) 65–81

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Background–foreground interaction for moving object detection in dynamic scenes

Zhe Chen[a], Ruili Wang[b,c,*], Zhen Zhang[a], Huibin Wang[a], Lizhong Xu[a]

[a] College of Computer and Information, Hohai University, Nanjing, China
[b] School of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou, China
[c] School of Natural and Computational Sciences, Massey University, Auckland, New Zealand

ARTICLE INFO

Article history:
Received 18 July 2018
Revised 17 November 2018
Accepted 20 December 2018
Available online 11 January 2019

Keywords:
Weighted kernel density estimation
Background-foreground interaction
Moving object detection
Dynamic scene

ABSTRACT

Both background subtraction and foreground extraction are the typical methods used to detect moving objects in video sequences. In order to flexibly represent the long-term state and the short-term changes in a scene, a new weighted Kernel Density Estimation (KDE) is proposed to build the long-term background (LTB) and short-term foreground (STF) models, respectively. A novel mechanism is proposed to support the interaction between the LTB and STF models. The interaction includes the weight transmission and the fusion between the LTB and STF models. In the weight transmission process between the LTB and STF models, the sample weight of one model (either the background model or the foreground model) in the current time step is updated under the guidance of the decision of the other model in the previous time step. In the background-foreground fusion stage, a unified Bayesian framework is proposed to detect objects and the detection result in any time step is given by the logarithm of the posterior ratio between the background and foreground models. This interactive approach proposed in this paper improves the robustness of moving object detection, preventing deadlocks and degeneration in the models. The experimental results demonstrate that our proposed approach outperforms previous ones.

1. Introduction

1.1. Moving object detection

Detecting moving objects in dynamic scenes is the first and crucial step in many outdoor surveillance systems [1,2]. Foreground extraction and background subtraction are the typical methods for moving object detection. Foreground extraction is a motion detector that classifies pixels according to the changes in the incoming frames, while background subtraction commonly works like a subtracter which suppresses the background by comparing an incoming frame to the background template. These models can also be categorized into two groups: parametric and nonparametric models [3]. The parametric models are established by a limited number of parameters which identify the distribution within interest regions (either background or foreground) [4]. Theoretically, any stable patterns or slow changes can be held with a series of predefined distributions and parameters. However, in a dynamic scene where its background has very high-frequency variations, the

* Corresponding author.
E-mail address: Ruili.wang@massey.ac.nz (R. Wang).

https://doi.org/10.1016/j.ins.2018.12.047
0020-0255/

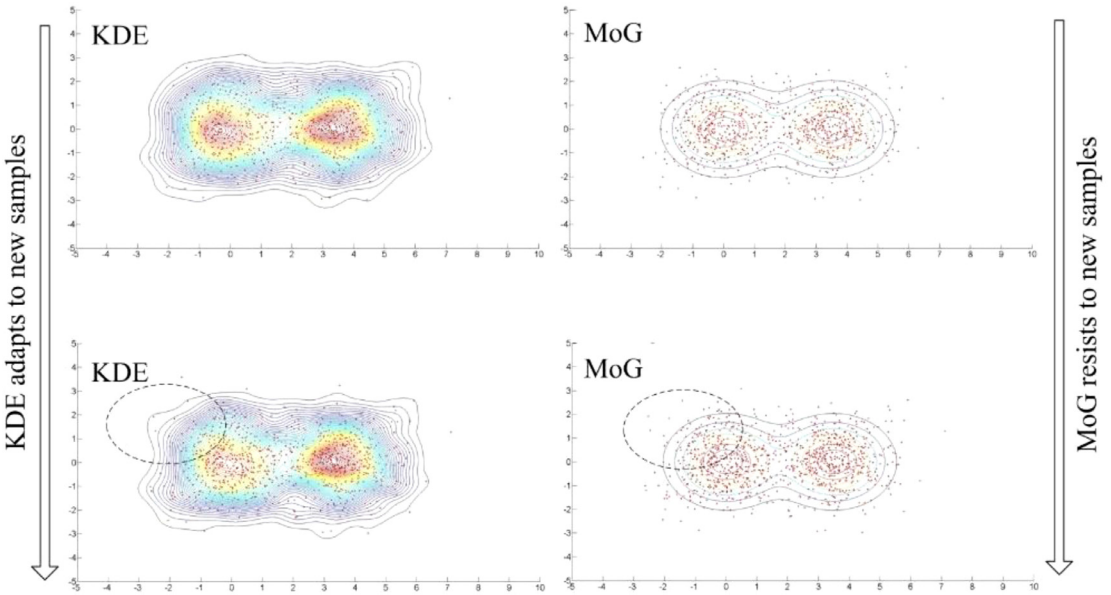

Fig. 1. Model sensitivity: the results on the original samples (the 1st row), and the results after adding white noises samples (the 2nd row).

scene distribution is of a multi-modal pattern which cannot be closely modelled by limited parametric distributions. As a result, parametric models sometimes fail to obtain sensitive detection results in dynamic scenes. Alternatively, nonparametric models are independent of parameters, in which the knowledge about a scene is indicated by selected samples [5–7]. In contrast to parametric models, nonparametric models are free from any assumption or prior knowledge but totally depend on their selected samples. They have a better ability to hold multi-modal distributions and have more opportunities to adapt to scene changes. In order to illustrate this problem, we model the distribution of 400 samples which are of a 2D Gaussian distribution by the Kernel Density Estimation (KDE) and Mixture of Gaussian (MoG) models, respectively (the first row in Fig. 1). We then add a 2D white noise signal (the ellipse) and model the combined samples again (the second row in Fig. 1). These results clearly show that the white noise significantly changes the pattern of the nonparametric model (i.e., KDE) while having little effect on the parametric model (i.e., the MoG). These results demonstrate that the nonparametric model has a better adaptive performance to scene changes, but its performance may be more susceptible if any noise is introduced into sample sets.

1.2. Motivations and distinctive features

Both of the aforementioned factors (i.e., scene changes and dynamic noises) exist in dynamic scenes. Therefore, it is desirable that background and foreground models are adaptive to scene changes and robust against dynamic noises. This largely depends on the samples selection and updating strategy, which include the following concerns:

- How to select samples to represent the background or foreground patterns and adapt to various scene changes?
- How to estimate and quantify the accuracy and representativeness of the selected samples?
- How to introduce these factors (i.e., the accuracy and representativeness of selected samples) into the modelling process?
- How to update model samples properly?

Theoretically, these issues affect the performance of a model, either adaptability or stability. Moreover, dynamic scenes also bring many other practical challenges to moving object detection such as illumination changes, cluttered scenes and camouflage appearances. To handle these theoretical tradeoff and practical issues, any single background or foreground model is not sufficient. When a background or foreground model focuses on its long-term pattern, its short-term pattern is inevitably missed, vice versa. If we jointly use the background and foreground models in a united framework, we can establish a stable and adaptive object detection approach which takes both LTB and STF patterns into account. Another motivation for jointly using the LTB and STF models is that decisions given by an individual LTB or STF model can detect objects using different cues (i.e., background or foreground features) and a more robust result can be obtained by fusing these decisions.

In this paper, we focus on the aforementioned theoretical and practical issues, proposing a novel background-foreground interactive mechanism. In this mechanism the background model aims to extract long-term features and suppress the stable background, while the short-term variations are modelled by the foreground model which extracts the variations caused by objects and removes noises in a dynamic scene. Moreover, a novel weighted KDE method is proposed in this paper. In our background model, its sample weights are updated according to the decision given by the foreground model, while our

foreground samples are updated by the decision of the background model. Through this interaction, instantaneous intensity changes caused by object motions can be detected, while small disturbances which happen periodically can be removed. Consequently, our approach is more robust and leads to moving object detection results high in accuracy and low in false alarms. Some of the distinctive features of this approach include:

- A novel background-foreground interactive mechanism is proposed to solve the trade-off problem between stability and adaptability for nonparametric models.
- A sample-weight factor is designed to enhance the robustness of the nonparametric models, establishing a weighted KDE for modelling the background and foreground patterns.
- A novel foreground model is proposed by combining the weighted KDE with the motion saliency.

2. Related works

Previous methods for object detection are wide-ranging such as foreground or background modelling, feature point detection, and image segmentation. Our approach aims to detect moving objects based on the interaction between foreground and background models. Thus, foreground and background modelling are the topics most related to this paper.

Background models assume that a video is captured by a static camera, thus the stable state of the background can be modelled by learning from historical frames. By comparing an incoming frame to the learned stable state, moving objects can be detected. These models generally can be divided into two categories. One employs a parametric form while the other one employs a nonparametric form.

With respect to the parametric form, typical examples include the single Gaussian distribution [8], Mixture of Gaussian (MoG) [9], Hidden Markov [10], and the linear autoregressive models [11,12]. To remove random image noises, a single Gaussian distribution models the intensity value of each pixel over time by a uni-modal pattern. This model is effective in static scenes but it can hardly remove the temporal changes when the intensity distribution is multi-modal. One solution proposed to solve this problem is using the mixture Gaussian distribution [4]. By modelling the distribution of a pixel with the weighted mixture of Gaussian distributions, MoG has the ability to obtain good results when background is slightly dynamic. However, as mentioned above, a parametric model includes more than one parameter. To estimate these parameters is challenging and needs a tradeoff between the stability and adaptability. Several learning and optimizing methods are introduced to estimate the model parameters such as the recursive Bayesian learning [13] and variable-bandwidth mean shift methods [14]. Another solution to dynamic scenes appears by predicting the state of the background. Based on MoG, the Hidden Markov model further introduces the transition between frames as the cue for temporally updating the parameters. Unlike the traditional image-averaging approach [4], this update process is based on the likelihood of membership; hence slow-moving objects can be handled. In order to remove dynamic noises, the auto-regressive model [11,15] treats an image as a time series and establishes an autoregressive model to capture the most important variations. Although these state prediction methods have a better adaptability to scene changes, they nevertheless also suffer from the problem that a fixed number of distributions cannot suffice to model all sorts of background modals.

To address this issue, nonparametric forms are developed as an alternative way to model backgrounds. Typical examples include ViBe [16], codebook models [17], KDE approaches [18] and the robust principle component analysis (RPCA) method [19]. The ViBe model for each pixel records a set of values taken in the past at the same location or in the neighborhood. The value in the current frame is then compared with the sample set to determine whether that pixel belongs to the background. ViBe achieves superior performance because samples can flexibly represent background changes. However, the color feature used in the ViBe method is sensitive to illumination changes and noises, which degrades its performance in dynamic scenes. The codebook model quantizes samples into a codebook which is the compressed representation of a background [17]. In order to improve the adaptability to dynamic scenes, an improved codebook model [20] is proposed by fusing the spatial and temporal information of pixels. Moreover, the codebook background model is further upgraded to a multiple layer framework and the image blocks are taken as the component to establish the codebook [17].

By using dimensionality reduction, subspace methods have been widely proposed in literature, such as the Robust Principle Component Analysis (RPCA) [19]. These methods estimate a background by a sparse representation and rank minimization. They are further integrated with statistical features to improve the performance for background subtraction. According to the same idea, another subspace analysis method, i.e., Independent Component Analysis, is introduced for detecting foreground objects [21]. Recently, a sparsity-based representation has drawn a lot of attention, and several sparse background modelling methods have been developed [22]. These methods have a capability to present a scene as a sparse linear combination of training signals themselves and uncover semantic information in image data.

Within the group of nonparametric models, KDE [23] is almost the most popular one. This method directly selects samples within a temporal window to represent the intensity distribution of individual pixels. Objects are detected by comparing an incoming frame to these selected samples. This method is free from any assumption on the density and also has a good ability to adapt to complex dynamic scenes. There are two typical sample selection strategies for KED: selective sampling and blind sampling. The selective sampling process enhances the sensitivity of the KDE model, while the blind sampling mechanism improves the generalization In general, the improvement of the classic KDE is twofold: kernel design and sample selection. For example, in order to improve the temporal adaptability of the KDE model, time-adaptive conditional kernel density estimation (CKDE) is proposed, which has a forgetting factor that gradually reduces the importance of the old sam-

ples and enhances the importance of the new samples [24]. In this method, the conversion of the static Nadaraya Watson (NW) estimator is introduced as a recursive time-adaptive estimator. Moreover, this CKDE model is also characterized by its adaptability in kernel selection. For kernel design, a variable kernel density estimation (VKDE) method can stretch and rotate kernels to focus on given sampling points, which enhance the correctness of the model fitting [25]. However, in this model the bandwidth of the kernel needs to be carefully designed, and in practice it is difficult to select a proper bandwidth for kernel calculation, which can make the model unstable. Based on an optimization theory, a fully data-driven bandwidth selection method is proposed with the concept of minimal penalty [15]. Recently, a spherical penalized comparison to overfitting (SPCO) procedure is proposed for designing the kernel bandwidth. In this work, the rates of convergence for the MISE (Mean Integrated Square Error) are computed as a rule for bandwidth estimation [15].

To adaptively model dynamic backgrounds, deep learning methods have been used to establish a Background Extracting and Learning Network [26–30]. In contrast to traditional methods a deep network has a better performance in background learning. Also, the deep learning architectures are used as an encoder to model the background. The latent patterns in the network are more powerful representations of the background, which is learned in an unsupervised and bottom-up manner [31].

In general, a background model based object detection method requires an accurate estimation of the background. Its performance is good if the scene background does not change much in a certain temporal interval. However, this is not the case for the applications in dynamic scenes where the background has high-frequency variations. Moreover, there are many tradeoff issues unsolved for background models such as the option between the long-term and short-term model, and between blind and selective sampling.

In contrast to the background, foreground objects have very high frequency variations. In order to establish a foreground model, foreground features are desired to be temporally stable. For example, the principal component analysis (PCA) vector [19], lasso foreground model [32], spatial coherence [33] and histogram [34] have been successively introduced for establishing foreground models. The PCA vector is generated by dimension reduction of a temporal dataset, which recently has been used as the observation model in the Kalman filter [35]. In order to handle the challenges in dynamic scenes such as illumination changes and dynamic noises, an adaptive generalized fused lasso has been proposed as a flexible structural prior to modelling a foreground [32]. Recently, the binary descriptor based foreground model achieves a coarse-to-fine detection of foreground objects. For this descriptor, the first-in-first-out sample updating strategy is used to maintain the most recent observed instances. Besides the temporal vector, the spatial coherence assumes that spatial uniform features exist in foreground objects. In other words, if a pixel is detected as a foreground, its neighbourhoods belong to the foreground with a high probability, which can be modelled by the single Gaussian distribution [36]. Also, the histogram of sparse codes has been introduced for object detection [37].

3. Proposed method

According to the previous discussions, nonparametric models have a better ability to handle high-frequency variations in dynamic scenes. However, the sample selection is a tradeoff issue between stability and adaptability for nonparametric models. By using any one strategy, it is impossible to provide a united solution available to all situations. For this tradeoff issue, a solution can be given by the interaction between a pair of models, if we specialize one of a pair of models to one site of the tradeoff and keep both of them in a united framework. Moreover, the interaction between models also enables the detection to be robust against dynamic noises.

In this paper, we propose a model interaction method to solve the problems existing in the nonparametric models, enhancing the performance of the object detection in dynamic scenes. Fig. 2 shows the block diagram of the proposed object detection approach. Firstly, aiming to model foreground objects, a motion saliency detection method is combined with a novel weighted KDE model to initially identify the foreground object regions. In this foreground model, the motion saliency detection results are used as the preprocessor to initially label the samples belonging to foreground objects. The samples of foreground objects are given with large weights, while background counterparts outside are assigned with small weights. These weights are then integrated with the KDE model to establish a novel weighted KDE based foreground model. Secondly, the weighted KDE model is further used to establish the background model. Different from the foreground model, the initial sample weights in the background model are given with a default value since we can initially assume that each sample in a scene is static and belongs to the background. Here, we do not utilize the motion saliency to initialize the background model. This can prevent the deadlock situation when errors in the motion saliency may stubbornly transmit to the final results. Thirdly, we establish a weight transmission method, wherein the decisions given by the background model are introduced to update sample weights in the foreground model, vice versa. Finally, under a unified Bayesian framework [38] final detection results are given by the fusion between the background and foreground models. In this paper, the weight transmission and model fusion processes are jointly called the model interaction mechanism. This interaction mechanism can theoretically provide a solution to the tradeoff issue in both the background and foreground nonparametric models. Also, this model interaction mechanism has a capability to synchronously suppress the background and highlight the foreground. Moreover, the interaction and resampling processes in our approach have the capability to prevent the deadlock and sample degeneration problems. Different from previous methods, in our approach the updating process of one model is forced by the decision from the other model. This update decision of one model is not generated by itself, and newly-added samples can adapt the model to new distributions. We qualitatively measure the degeneracy of model samples. When a model

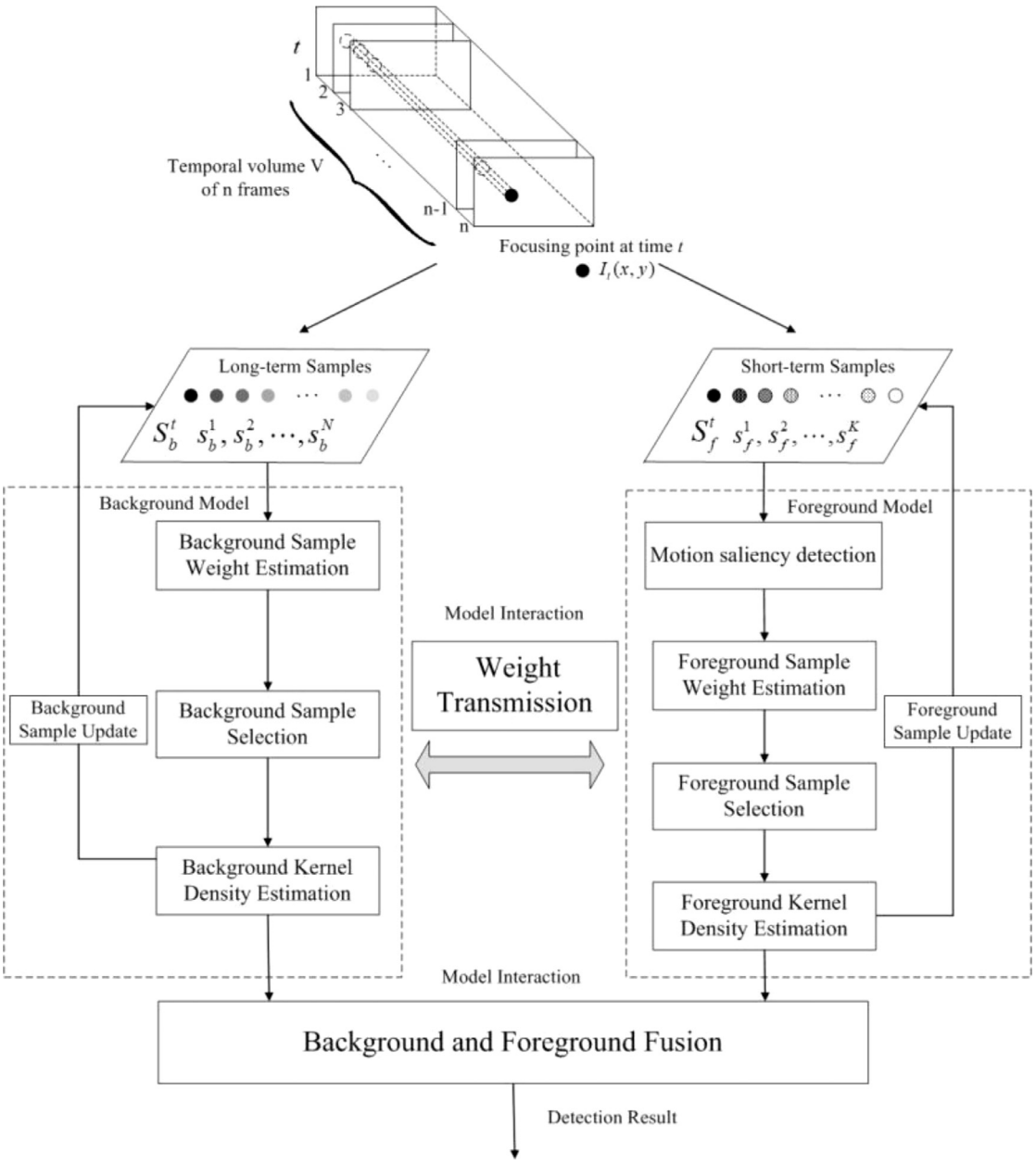

Fig. 2. Flowchart of our moving object detection approach.

degenerates, the weights of some samples (degenerated samples) will present extremely low values. In this work, the first-in-first-out manner is employed to eliminate these degenerated ones and add new samples.

4. Bayesian framework for background-foreground model fusion

Inspired by the work presented in [38], this paper presents a Bayesian framework to fuse the nonparametric background and foreground models into a uniform probability density model. The Bayesian framework is the combination of a pair of conditional probability functions. The first measures the possibility of pixels belonging to the background, while the second measures the possibility of pixels involved in the foreground.

Suppose we want to detect a moving object in a frame, each pixel can be classified into the background or foreground. Assume x_t is the pixel at time t, the decision on this point is D_i:

$$D_t = \begin{cases} B_t & x_t \in B_t \\ F_t & x_t \in F_t \end{cases}, \tag{1}$$

According to the Bayesian framework, the probability of the pixel x_i belonging to the background can be formulated as a conditional density:

$$P(B_t|X_t) = P(B_t|x_t, X_{t-1}) = \frac{P(x_t|B_t, X_{t-1})P(B_t|X_{t-1})}{P(x_t|X_{t-1})}, \tag{2}$$

where X_t, X_{t-1} are the temporal vectors; $X_t = \{x_1, x_2, \cdots, x_t\}$ and $X_{t-1} = \{x_1, x_2, \cdots, x_{t-1}\}$.

Since the background model B_t is explicitly estimated by the vector X_{t-1}, $P(x_t|B_t, X_{t-1}) = P(x_t|B_t)$ which is calculated by our background model, $P(B_t|X_{t-1})$ estimates the probability of the pixel x_{t-1} belonging to the background B_t at time step t, which can be calculated by the transmission of $P(B_{t-1}|X_{t-1})$ from time step *t-1* to time step t:

$$\begin{aligned} P(B_t|X_{t-1}) &= \sum_{D_{t-1}\in\{B_{t-1},F_{t-1}\}} P(B_t|D_{t-1}, X_{t-1})P(D_{t-1}|X_{t-1}) \\ &= \sum_{D_{t-1}\in\{B_{t-1},F_{t-1}\}} P(B_t|D_{t-1})P(D_{t-1}|X_{t-1}), \end{aligned} \tag{3}$$

where $P(B_t|B_{t-1})$ is the probability of keeping the state stable, while $P(F_t|B_{t-1})$ is the probability of the state change.

The normalization factor can be calculated as:

$$\begin{aligned} P(x_t|X_{t-1}) &= \sum_{D_t\in\{B_t,F_t\}} P(x_t|D_t, X_{t-1})P(D_t|X_{t-1}) \\ &= \sum_{D_t\in\{B_t,F_t\}} P(x_t|D_t)P(D_t|X_{t-1}), \end{aligned} \tag{4}$$

Using the same form in (2)–(4), the probability of the pixel x_t belonging to the foreground $P(F_t|x_t)$ can be calculated.

Finally, the moving object detection fusion between the background and foreground is obtained by the logarithm of the posterior ratio as:

$$\delta(x_t) = \begin{cases} 0 & \psi > \theta \\ 1 & \psi \le \theta \end{cases}, \tag{5}$$

where $\psi = -\ln \frac{P(B_t|X_t)}{P(F_t|X_t)}$ and θ is the threshold.

5. Weighted KDE model

5.1. KDE model

Assuming N samples x_i are selected, the probability that pixel x_t belongs to the background at time t is expressed by the KDE model as:

$$P(x_t|D_t) = \frac{1}{N}\sum_{i=1}^{N} K(x_t - x_i), \tag{6}$$

where $K(x)$ is the kernel function and satisfies the following conditions: $\int K(x)dx = 1$, $\int xK(x)dx = 0$ and $K(x) > 0$. Using this estimation, if $P(x_t|D_t) > \gamma$, and x_t is considered to belong to D_t.

5.2. Weighted density

We introduce quantified sample weights into the KDE model, establishing a weighted KDE model. The formal structure of the weighted KDE model can be described as:

$$P(x_i|B_t) = \frac{1}{N}\sum_{i=1}^{N} w_i K(x_t - x_i), \tag{7}$$

where w_i is the weight for the sample x_i, which evaluates the confidence and importance of the sample for representing the modal that we are interested in.

This novel weighted KDE model has many advantages such as:

(i) Enhancing the robustness of the selected samples.
Using the sample weights, we can adapt the model to samples which are identified to be important to the modal, while the disturbance caused by the false samples will be suppressed since they are attached with small weights and will be removed in the sample updating process. As a result, the representability of the selected samples is enhanced, improving the robustness of the models.

(ii) Enhancing the model flexibility.
Using the sample weights, we can determine the temporal intervals from which we select samples. If samples are selected in a long-term interval, the established model is sensitive to the stable modal, while if sample selection is operated within a short-term interval the model is more sensitive to incoming data and hence adaptive to the scene changes.

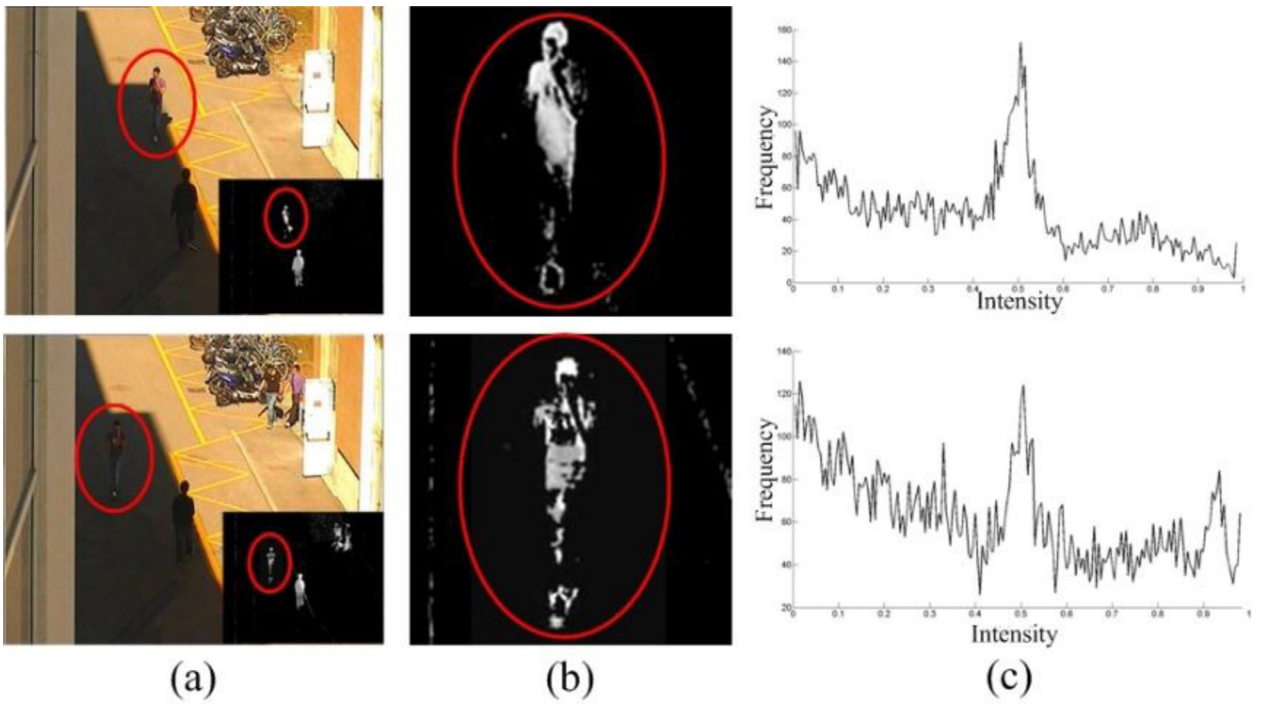

Fig. 3. Motion saliency: (a) the video frames, (b) the motion saliency, and (c) the saliency histogram. The red ellipse: the interest object and corresponding motion saliency. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(iii) Preventing the dead-lock situation.
In our model interaction mechanism, the decision given by the foreground model is introduced to update sample weights of the background model, while the decision given by the background model is used to update the foreground samples. With this weight transmission process, the weights for error samples will be degraded step by step and these error samples will be removed from the sample set by the sample updating process.

(iv) Transformation between the background and foreground models.
In the previous studies, it is almost impossible to identify the samples as to whether they belong to the background or the foreground. Hence, the KDE model is commonly specialized to the background under the assumption that the object in the scene is relatively small-sized. However, this is not the case in some scenes especially the close-range monitoring environments. This problem can be solved by our sample weighting method. With the sample weights, the weighted KDE model actively converges on our interest modal, i.e., the foreground and background. As a result, we can use the weighted KDE to establish both the background and foreground models.

5.3. Weight initialization

In order to make our weighted KDE model accurately converge on our interest modal, the sample weights need to be carefully initialized. For the weighted KDE based foreground model, our previously proposed Temporal Fourier Transform (TFT) [39] based motion saliency method is used to initialize the sample weights.

When objects move in a scene, their appearances have very high-frequency variations. This problem is further aggravated by illumination changes and occlusions [40,41]. An example is as shown in Fig. 3. In this situation, it is difficult to use the intensity features such as the grey and colour values, or the morphological features such as the contours and skeletons to stably identify foreground objects and initialize the sample weights. The motion saliency, according to our previous work, has invariant features against the scene changes [39]. From the examples shown in Fig. 3(b), the motion saliency keeps stable when the pixel intensity in the original frames changes dramatically. Moreover, the amplitude of the saliency value is absolutely related to the contrast between the object and the background. Since the T-shirt has a higher contrast to the background as shown in Fig. 3, the saliency value of the T-shirt is larger than that of the jeans. This implies that the motion saliency values are in line with the foreground appearances. This assumption is further demonstrated by the saliency histogram shown in Fig. 3(c). According to the histogram, the saliency distribution of the moving object is generally a stable uni-modal pattern. The peak of the distributions in different frames appears in the same location, which can be regarded as a temporally invariant feature for foreground objects.

The underlying principle of the TFT-based method is the correspondence between variations in the phase spectrum and the time-varied "pop-out." We define the temporal sequence for the incoming sample x_t at time t as $X_t=[x_t, x_{t+1}, x_{t+2}, \cdots, x_{t+l-1}]$ which is established by stacking samples frame by frame in the L-length temporal interval.

The Fourier Transform on this sequence and the phase spectrum are expressed as:

$$f_t = \Im(X_t),\ p_t = angle(f_t) \tag{8}$$

where $\Im$ denotes the Fourier Transform, and p_t represents the phase spectrum of the temporal sequence X_t.

Finally, the motion saliency in time t can be calculated by the Inverse Fourier Transform as:

$$f'(t) = g_t * \Im^{-1}(p_t), \tag{9}$$

where $\Im^{-1}$ denotes the Inverse Fourier Transform, and g_t is a one-dimensional Gaussian filter ($\sigma = 5$).

One of the most useful aspects of information acquired from the Inverse Fourier Transform is the wave variation. Large variation amplitudes indicate motion of foreground objects. Here, we use a threshold filter to detect the motion saliency in a temporal sequence:

$$g_t = \begin{cases} \|f'(t)\| & \|f'(t)\| > T \\ 0 & \|f'(t)\| \le T \end{cases}, \tag{10}$$

where T is the threshold.

In a motion saliency map, the intensity of any point in time t represents the probability that observing x_t belongs to the foreground. Hence, the intensity of the saliency map can be used as the cue to initially estimate the sample weights for the foreground model:

$$w_i''(0) = \frac{S_i(0)}{\sum_{i=1}^{N''} S_i(0)}, \tag{11}$$

where N'' is the number of the samples involved in the foreground model.

However, for the samples in the background model, we do not use the motion saliency based preprocessor to initialize them, preventing the dead-lock situation. Hence the sample weights for the background model are given with default values as:

$$w_i'(0) = \frac{1}{N'}, \tag{12}$$

where N' is the number of the samples involving in the background model.

6. Model interaction

6.1. Sample selection

As aforementioned, the model sensitivity is a tradeoff issue between stability and adaptability. Up to date, none of the existing methods has a capability to both stably model the long-term state and flexibly adapt to the short-term changes. In order to handle this issue, we separately control the sensitivity of the background and foreground models by different sample selection strategies. For the background model, N' samples (refer to Eq. (12)) are expected to hold the long-term state of a scene. Hence, these samples are selected within a long temporal interval. For the foreground model, N'' samples (refer to Eq. (11)) are expected to capture short-term changes. A short-term window with the length $\upsilon \times N''$ ($\upsilon = 2$ in this paper) is therefore adopted.

6.2. Weight transmission

For selected samples, their weights directly determine their importance. A natural notion to update sample weights is based on a self-loop circle where the updating process is driven by the decision given by the same model in previous time steps. However, this trick will cause a dead-lock situation when errors occur in any time steps.

In order to solve this problem, a novel weight transmission method is proposed as shown in Fig. 4. In this method, the decision of the foreground model in time t-1 is transmitted into the background model to update background sample weights in time t, as:

$$\Delta w'_i(t) = 1 - P(x'_i|F_{t-1}),$$
$$\omega'_i(t) = \begin{cases} w'_i(t-1) + \alpha \Delta w'_i(t) & \Delta w'_i(t) > \tau \\ w'_i(t-1) - \beta \Delta w'_i(t) & \Delta w'_i(t) \le \tau \end{cases},$$

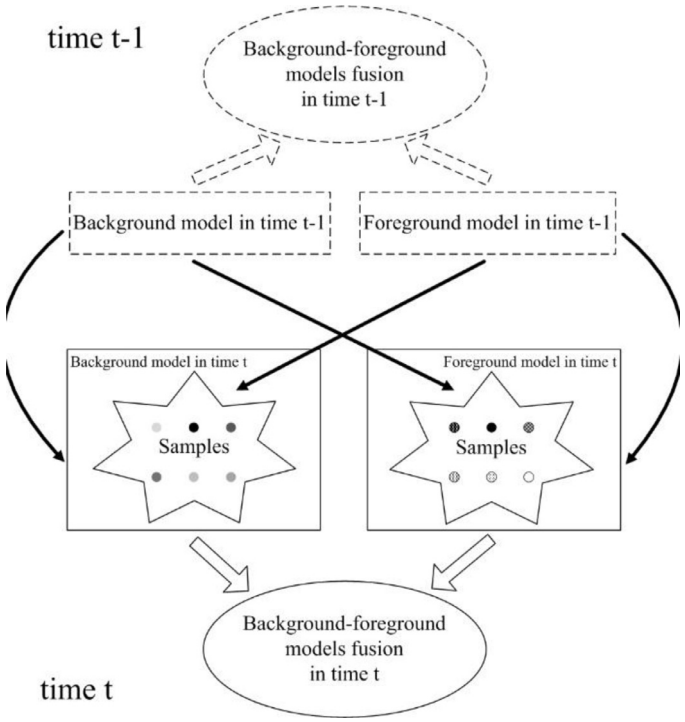

Fig. 4. Weight transmission.

$$w'_i(t) = \frac{\omega'_i(t)}{\sum_{i=1}^{N'} \omega'_i(t)}. \tag{13}$$

Similarly, the decision of the background model in time t-1 is transmitted into the foreground model to update the foreground sample weights in time t, as:

$$\begin{aligned} \Delta w''_i(t) &= 1 - P(x'_i|B_{t-1}), \\ \omega''_i(t) &= \begin{cases} w''_i(t-1) + \alpha \Delta w''_i(t) & \Delta w''_i(t) > \tau \\ w''_i(t-1) - \beta \Delta w''_i(t) & \Delta w''_i(t) \le \tau \end{cases}, \\ w''_i(t) &= \frac{\omega''_i(t)}{\sum_{i=1}^{N''} \omega''_i(t)}. \end{aligned} \tag{14}$$

where α, β and τ are the parameters defining the weight moderation; $P(x''_i|B_{t-1})$ is the probability that sample x''_i belongs to the background in time t-1, $P(x'_i|F_{t-1})$ is the probability that sample x'_i belongs to the foreground in time t-1.

From Eqs. (13) and (14), if a sample is identified as the background i.e., $\Delta w'_i(t) > \tau$ or $\Delta w''_i(t) \le \tau$, its weights are increased in the background model while decreased in the foreground model, and vice versa. In this mechanism, the updating process of one model (the background-foreground model) is driven by the decision given by the other model (foreground or background model). Notice that this interaction allows samples outside the previous distribution to be added into the updated model, adapting the model into a novel distribution. Moreover, since the LTB model is updated using the samples detected by the STF model, the adaptability of the LTB model to scene changes is enhanced. Similarly, the stability of the STF model is improved due to the introduction of the samples detected by the LTB model. Hence, our proposed weight transmission mechanism can reduce the possibility of the deadlock situation, and both the adaptability and stability of the background and foreground are improved.

6.3. Resampling

A common problem with sample based methods is the degeneracy phenomenon [36] when after a few iterations all but few samples will have negligible weights. This degeneracy effect implies that a large computational effort is devoted to the samples which however have little importance to background and foreground models. The sample degeneracy can be

measured as:

$$\mathrm{M}' = \frac{1}{\sum_{i=1}^{N'} (w'_i(t))^2} \quad \mathrm{M}'' = \frac{1}{\sum_{i=1}^{N''} (w''_i(t))^2}, \tag{15}$$

where M' and M'' are the degeneracy measurements for the background and foreground models, respectively.

Note that small values of the parameters M' and M'' indicate severe degeneracy of the background and foreground samples. A sound approach to reduce this effect is to use a resampling strategy. The basic operation of resampling is to eliminate the samples that have small weights and to focus on the samples with large weights [42,43]. Moreover, the resampling in our method is jointly used with a first-in-first-out manner. The oldest sample is discarded and a new sample is added in each step, preventing the loss of diversity among samples.

7. Experimental results

In this section, we present a set of experiments. In each of these experiments, we keep the resolution of the input frames as the original resolution of the frames themselves. The PC was equipped with a core 2.4 G and 4 G of memory. The PASCAL criterion [44], C, C_{good} and C_{false} are used to evaluate the overlap of the detection results and ground truth:

$$C = \frac{\Omega' \bigcap \Omega}{\Omega' \bigcup \Omega}, C_{good} = \frac{\Omega'_e \bigcap \Omega_o}{\Omega'_e}, C_{false} = \frac{\Omega'_e \bigcap \Omega_b}{\Omega'_e}, \tag{16}$$

where Ω' is the detected results; Ω is the ground-truth; Ω'_e is the region of the detected object area; Ω_o is the ground-truth region of the moving object to be detected; Ω_b is the background region in the ground truth; C is the overlap; C_{good} is the ratio of the correct detection in the true object region, and C_{false} is the ratio of the false extraction in the background region. Obviously, the larger the value of C_{good} and the smaller the value of C_{false}, the more robust the algorithm. In addition, the performance of our method is evaluated with respect to six criteria [23], i.e., the precision (Pr), similarity (Sim), true positive rate (TPR), F-score (FS), false positive rate (FPR), percentage of wrong classifications (PWC).

$$\mathrm{Pr} = \frac{tp}{tp+ft}, \ \mathrm{TPR} = \frac{tp}{tp+fn}, \ \mathrm{Fs} = 2 \times \frac{\mathrm{Pr} \times \mathrm{TPR}}{\mathrm{Pr} + \mathrm{TPR}}, \ \mathrm{Sim} = \frac{tp}{tp+fp+fn}, \ \mathrm{FPR} = \frac{fp}{fp+tn}, \ \mathrm{PWC} = 100 \times \frac{fn+fp}{tp+tn+fp+fn}, \tag{17}$$

where tp, tn, fp and fn denote the numbers of the true positives, true negative, false positive, and false negative, respectively. Video sequences used in this paper are from CAVIAR [45], ALOV++ [46] and the updated WallFlower [47], Shadow [48] databases.

The experiments are composed of two parts: (i) the performance of our approach is evaluated using various sample sizes; (ii) our approach is compared with the existing methods, i.e., ST-MoG (Spatio-Temporal MoG) [49], TSR (Temporal Spectral Residual) [50], BF-KDE (Background-Foreground KDE) [18], Vibe [16] and GFL(Generalized Fused Lasso) [32]. These methods use different strategies, all of which have an ability to handle dynamic backgrounds. ST-MoG utilizes spatial and temporal features to identify object motions from the dynamic noises. TSR is a typical temporal saliency detection method which has been demonstrated successful for motion detection. Vibe and GFL are excellent in feature extraction and perform well on different motion patterns. The main structure of BF-KDE is somewhat similar to our method, while the novelty of our method lies in the interaction mechanism between the foreground and background models. Typical scenes including dynamic backgrounds, illumination changes and camera motions are used to demonstrate the robustness of our approach.

7.1. Effect of the sample size

Fig. 5 demonstrates the effects of parameters in our method, i.e. N' and N''. Theoretically, a larger number of samples (larger N') support a more thorough depiction of the background. However, a large number of samples will bring many problems to our approach. Firstly, more samples will likely incur more errors in sample selection. Secondly, overfitting is another disadvantage caused by a large number of samples. Thirdly, the large number of samples will drastically increase the computational cost for model establishment.

From Fig. 5(c)–(f), (k) and (l) we can see that given a constant $N'' = 20$, the object detection accuracy firstly increases with the parameter N' and the optimal result is achieved at the turning point where $N' = 150$. After that, the accuracy does not increase anymore but decrease as N' increases. Parameter N'' is the sample size of our weighted KDE foreground model, which controls the performance of the foreground model. From Fig. 5(g)–(j), (m) and (n) we can see that given a constant $N' = 150$ the performance remains very stable when N'' changes. This result is attributed to the short temporal window for selecting foreground samples. In such a short temporal interval, the scene changes are relatively small. Limited samples thus have an enough capability to correctly model the STF distribution.

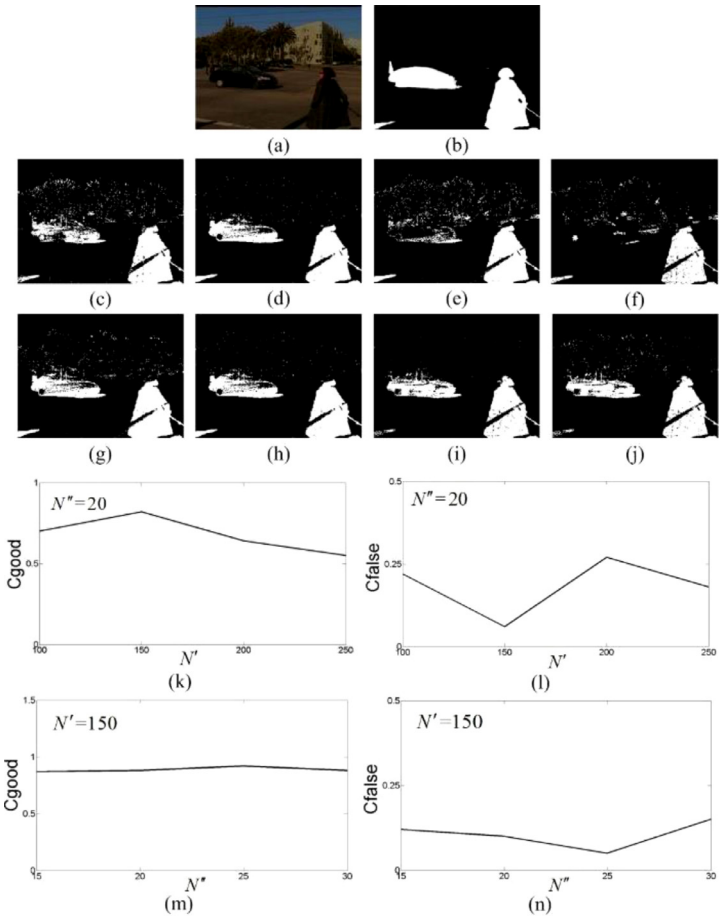

Fig. 5. Results given by different parameters N' and N''. (a) Original frame 560#. (b) Ground truth. (c) $N'' = 20$ and $N' = 100$. (d) $N'' = 20$ and $N' = 150$. (e) $N'' = 20$ and $N' = 200$. (f) $N'' = 20$ and $N' = 250$. (g) $N'' = 150$ and $N' = 15$. (h) $N'' = 150$ and $N' = 20$. (i) $N'' = 150$ and $N' = 25$. (j) $N'' = 150$ and $N' = 30$. (k) Curve of C_{good} when $N'' = 20$ and N' changes from 100 to 250. (l) Curve of C_{false} when $N'' = 20$ and N' changes from 100 to 250. (m) Curve of C_{false} when $N' = 150$ and N'' changes from 15 to 30. (n) Curve of C_{false} when $N' = 150$ and N'' changes from 15 to 30.

Table 1
Average performance comparison on video sequence *TRAFFIC*.

Method	$\bar{C}$	Pr	TPR	Fs	Sim	FPR	PWC	FPS
ST -MoG	0.6537	0.8446	0.6957	0.7629	0.6167	0.1960	3.3650	17
TSR	0.7021	0.0483	0.0704	0.0573	0.0295	0.7337	9.1462	175
BF-KDE	0.7215	0.8208	0.3595	0.5000	0.3333	0.0562	5.6849	21
Vibe	0.7167	0.6380	0.7899	0.7059	0.5455	0.0395	5.3336	20
GFL	0.7130	0.8006	0.4841	0.6034	0.4320	0.0114	5.4769	18
Our approach	0.7662	0.8605	0.7400	0.7957	0.6607	0.0968	1.9643	23

7.2. Comparison with existing methods

The average performance and the frame per second (FPS) comparisons between ST-MoG [49], TSR [50], BF-KDE [18], Vibe [16], GFL [51] and the proposed approach are presented in Figs. 6–10 and Tables 1–5. The performance of methods is demonstrated by difficult scenes with dynamic background, i.e., *TRAFFIC, SHADOW, BROWSE, WAVING TREES* and *HIGHWAY* video sequences.

Fig. 6 shows the experimental results of these six methods with the test sequence *TRAFFIC*, of 870 frames and size 320×240, as shown in Fig. 6(a). In this scene, as a result of the wind load the camera is shaking slightly and tree branches are displaced. Moreover, random image noises are obvious in the background. These factors will lead to a large number of false detections in the results if the detection method cannot remove the motion noises. The first column in Fig. 6 shows the original frames; the second column is the ground-truth; the third to seventh columns respectively show the results of ST-MoG, TSR, BF-KDE, Vibe and GFL and the last column presents the results of our approach. Visually, the results given

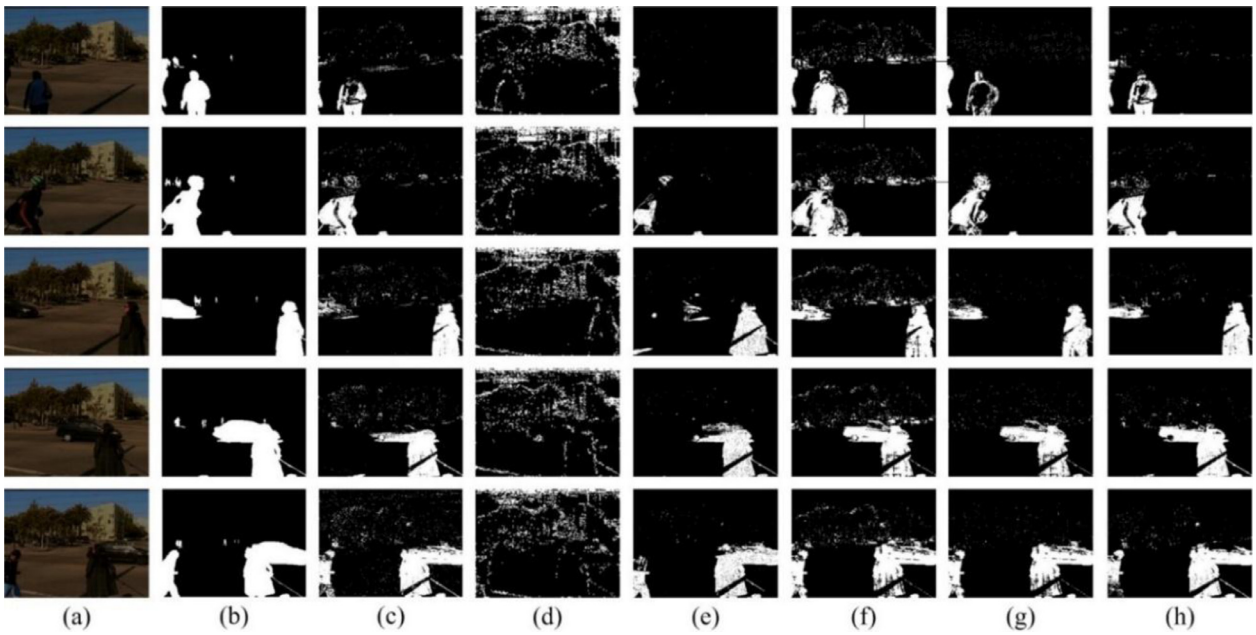

Fig. 6. Moving object detection results with video sequence *TRAFFIC*: (a) original frames, (b) ground truth, (c) ST–MoG, (d) TSR, (e) BF-KDE, (f) Vibe, (g) GFL, and (h) our approach.

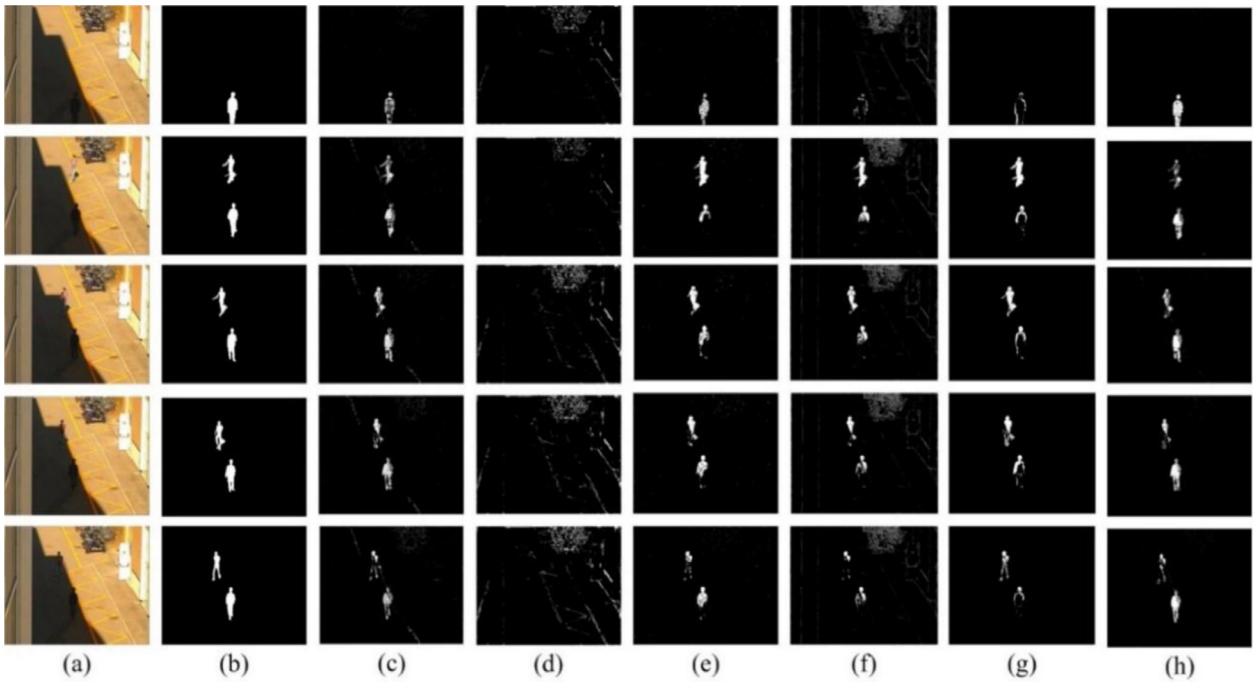

Fig. 7. Moving object detection results with video sequence *SHADOW*: (a) original frames, (b) ground truth, (c) ST–MoG, (d) TSR, (e) BF-KDE, (f) Vibe, (g) GFL, and (h) our approach.

by our approach is comparable to the results by GFL and better than results of other methods. GFL and our approach have better performances for removing the dynamic noises. In some cases such as the example in the first row, our method is better than GFL. However, the object in the second row is missed by our approach while GFL completely identifies the object region. To further examine the quantitative performance of our method, Table 1 shows the differences in the average performances of different methods; our method provides a good capacity to remove the dynamic background. The complexity of

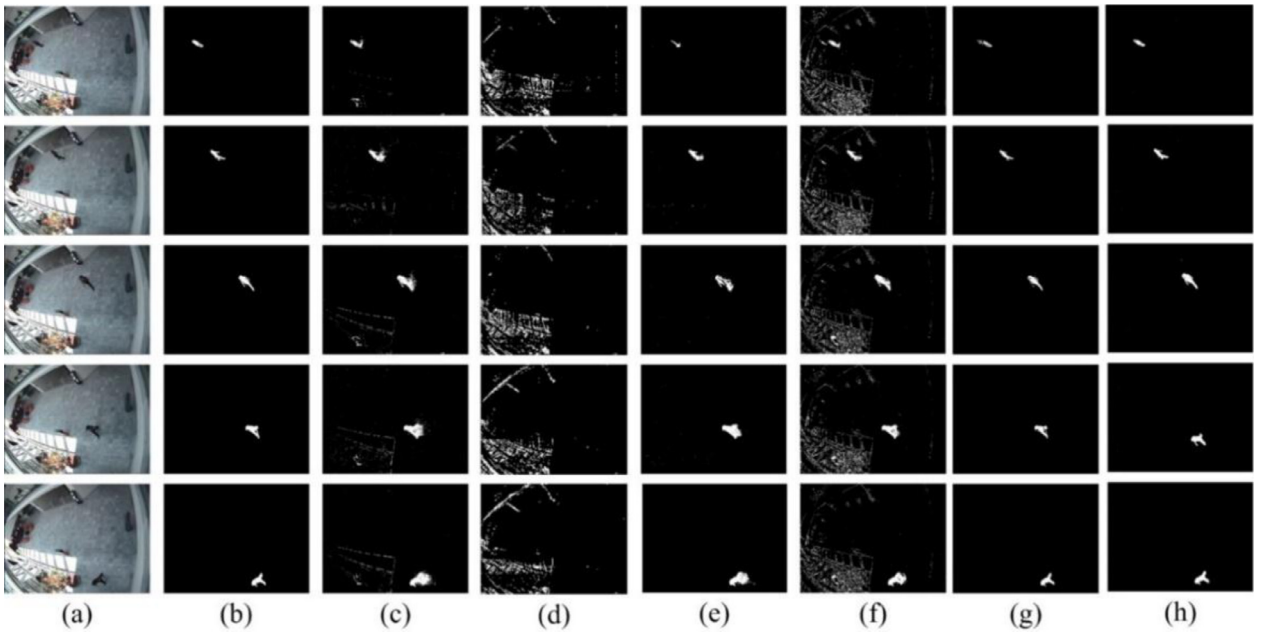

Fig. 8. Moving object detection results with video sequence *BROWSE*: (a) original frames, (b) ground truth, (c) ST–MoG, (d) TSR, (e) BF-KDE, (f) Vibe, (g) GFL, and (h) our approach.

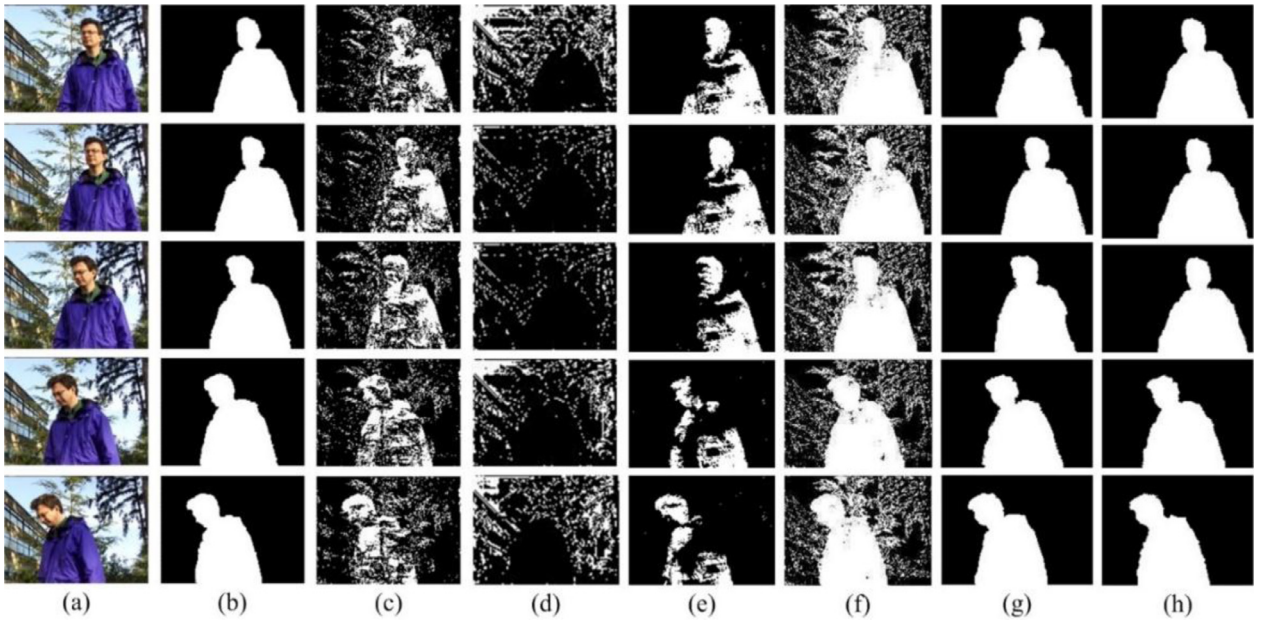

Fig. 9. Moving object detection results with video sequence *WAVING TREES*: (a) original frames, (b) ground truth, (c) ST–MoG, (d) TSR, (e) BF-KDE, (f) Vibe, (g) GFL, and (h) our approach.

Table 2
Average performance comparison on video sequence *SHADOW*.

Method	$\bar{C}$	Pr	TPR	Fs	Sim	FPR	PWC	FPS
ST -MoG	0.7531	0. 8986	0.6160	0.7271	0.5713	0.1169	8.6740	13
TSR	0.0143	0.1302	0.1070	0.1175	0.0624	0.3387	51.6779	95
BF-KDE	0.7884	0. 9565	0. 7288	0. 8273	0. 7054	0. 0401	1.8333	14
Vibe	0.7121	0.4328	0.6323	0.5139	0.3458	0.0202	2.8514	15
GFL	0.7681	0.8334	0.5476	0.6609	0.4936	0.0030	1.4760	14
Our method	0.8770	0.9167	0.9337	0.9251	0.8607	0.0766	1.1709	28

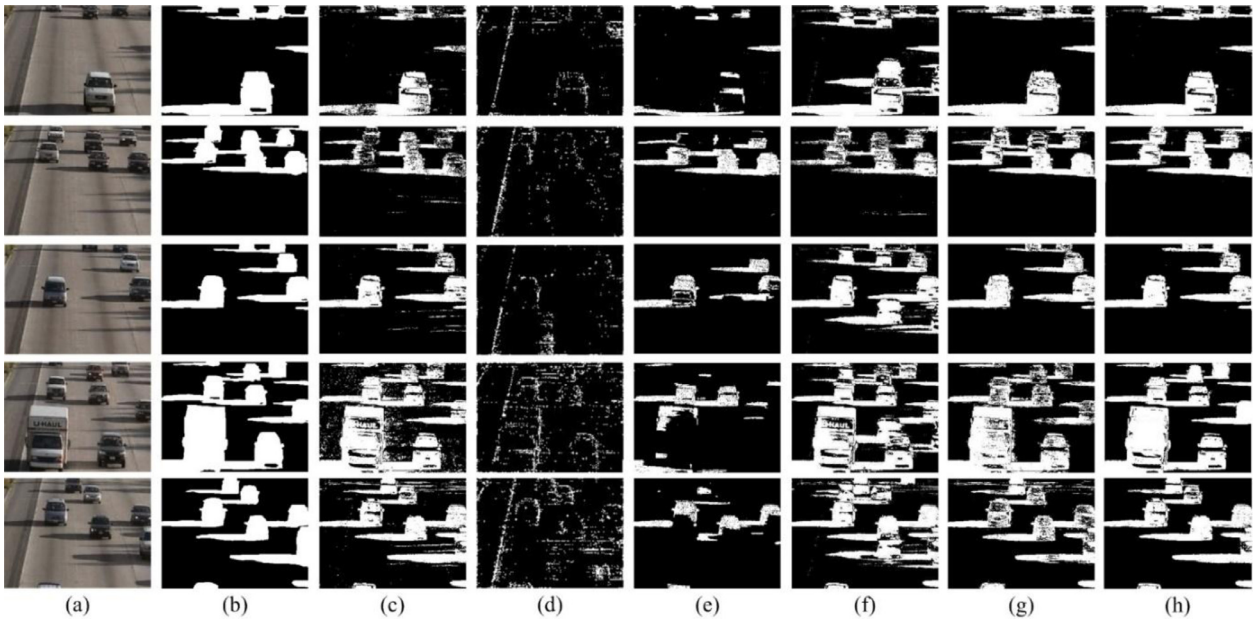

Fig. 10. Moving object detection results with video sequence *HIGHWAY*: (a) original frames, (b) ground truth, (c) ST–MoG, (d) TSR, (e) BF-KDE, (f) Vibe, (g) GFL, and (h) our approach.

Table 3
Average performance comparison on video sequence *BROWSE*.

Method	$\bar{C}$	Pr	TPR	Fs	Sim	FPR	PWC	FPS
ST -MoG	0.8416	0.7217	0.8582	0.7841	0.6440	0.0721	3.0030	21
TSR	0.0172	0.0200	0.0927	0.0329	0.0168	0.1848	7.7200	127
BF-KDE	0.8215	0.9800	0.9014	0.9391	0.8845	0.0024	1.4392	27
Vibe	0.7271	0.0749	0.9722	0.1390	0.0747	0.0413	4.1240	22
GFL	0.8928	0.9625	0.8271	0.8871	0.7971	0.0001	0.0768	15
Our method	0.9092	0.9940	0.9952	0.9946	0.9881	0.0013	1.0337	18

Table 4
Average performance comparison on video sequence *WAVING TREES*.

Method	$\bar{C}$	Pr	TPR	Fs	Sim	FPR	PWC	FPS
ST -MoG	0.6412	0.7671	0.6578	0.7082	0.5483	0.0885	16.6377	17
TSR	0.2109	0.2939	0.3040	0.2988	0.1757	0.2647	37.9384	175
BF-KDE	0.7659	0.9850	0.7006	0.8188	0.6932	0.0047	9.4510	21
Vibe	0.5914	0.5853	0.9815	0.7333	0.5483	0.0885	16.6377	20
GFL	0.9794	0.9800	0.9777	0.9789	0.9586	0.0088	1.2955	18
Our method	0.9854	0.9816	0.9944	0.9880	0.9762	0.0082	0.7393	23

our method, which is reflected by the FPS score, is relatively high. This problem can possibly be solved by using advanced computational hardware.

Fig. 7 shows the experimental results using the test sequence *SHADOW*, which contains 400 frames of size 704×567, as shown in Fig. 7(a). This sequence involves a large shadow area and the representation of the pedestrian between frames drastically changes. In contrast to the other five methods, our method achieves more accurate results under sunshine and shadow. The quantitative performance comparisons are shown in Table 2. From these results, the remarkable higher performance of our method is demonstrated. Using the motion saliency feature, the foreground model in our approach is robust against the changes in the object appearance as shown in Fig. 7(h). However, the shadow and appearance changes significantly decrease the performance of the other five methods, since numerous holes exist in the detection results as shown in Figs. 7(c)–(g).

Fig. 8 presents the results by using the video sequence *BROWSE* which contains 910 frames of size 384×288. Due to the shaking of the camera, in this scene, the background is dynamic, which leads to severe noises for motion detection. From Fig. 8, the best result is given by our method, while the second-best is achieved by the GFL method, as they both largely remove the textures in the left bright panel. However, these textural noises are stubbornly left in the results of the ST–MoG and Vibe methods and even inhibit the saliency of the moving object in the results of TSR, as shown in Fig. 8(d). Quantitative

Table 5
Average performance comparison on video sequence *HIGHWAY*.

Method	$\bar{C}$	Pr	TPR	Fs	Sim	FPR	PWC	FPS
ST -MoG	0.8294	0.9518	0.7598	0.8449	0.7314	0.0097	5.6384	20
TSR	0.4423	0.7167	0.3547	0.4746	0.3111	0.0425	18.2718	145
BF-KDE	0.3989	0.9723	0.3730	0.5392	0.3691	0.0023	11.4728	21
Vibe	0.6203	0.6513	0.8162	0.7245	0.5680	0.1018	11.7262	29
GFL	0.9120	0.9302	0.9052	0.9175	0.8476	0.0166	3.2014	24
Our method	0.9220	0.9785	0.8819	0.9277	0.8651	0.0047	2.6838	28

performance comparison of these methods is shown in Table 3. From these results, our method performs well in this scene. The capability of our method for removing the dynamic background and extracting the foreground is demonstrated.

Fig. 9 presents the results on the video sequence *WAVING TREES*. This scene is challenging to the background model since the movements caused by the moving trees in the background is much dramatic. From the results given in Fig. 9(g) and (h), our method and the GFL model give comparable results and both have a capability to completely remove the noises caused by waving trees. BF-KDE is demonstrated to be robust against the moving trees, as shown in Fig. 9(e), while this method cannot correctly extract the foreground object region. However, the waving trees cause severe background noises in the results given by other methods, i.e., ST-MOG, TSR and Vibe. According to the quantitative evaluations shown in Table 4, the general performance of our method is the best while the second best one is GFL.

The last result shown in Fig. 10 is conducted on the *HIGHWAY* video sequence. This data was originally used to test the methods for removing shadows. Since our method in this version does not consider the issue of the object shadow, the shadows of the vehicles hence are labeled as interest objects. According to Fig. 10, apart from TSR and BF-KDE, other methods all have a capability to extract the rough regions of the vehicles and their shadows. The video sequence *HIGHWAY* is short in length and all of the frames include moving foreground objects. This creates difficulties for background modelling and seriously degenerates the performance of the Vibe, ST-MoG and BF-KDE methods. Table 5 shows the quantitative evaluations on these six methods. From the scores in Table 5, our method works well, providing the best results in six criteria and the second-best results in the Pr.

Generally, from all the above results, our method gives a consistent performance in various dynamic scenes which can be demonstrated from two aspects. Firstly, among all four typical scenes with dynamic background noises, our method can correctly identify the regions of the objects of interest, as the criteria C, Pr, TPR, Fs and Sim are all kept at a high level. Especially, respecting these five criteria in the four scenes, our method is the best seventeen times and the second best three times, with the average score for each criterion larger than 0.9. Secondly, our method is also excellent in background noise removal, as the criteria FPR and PWC are maintained at a low level. In particular, with respect to these two criteria, the top two best results are obtained by our method. Our benchmarks include the three complicated conditions for object detection, i.e. illumination changes (Figs. 7 and 8), cluttered scenes (Figs. 8–10), and camouflage appearances (Fig. 9). Experimental results demonstrate that our method is robust against these challenging situations. In theory, our method presents a novel framework for object detection in complex scenes. In this framework, the background information continually interacts with the foreground information to jointly explore multiple cues in dynamic scenes, unlike previous strategies, which are only based on a single feature. This enables our method to correctly and stably detect objects in dynamic backgrounds. For example, in Fig. 10, the objects (vehicles) have fixed appearances. In this case, the foreground model plays a more important role in contrast to the background model. However, the background model is more important than the foreground model in Fig. 7, since the object (pedestrian) is variable in intensity. Hence an alternation between the foreground and background models is required for object detection. This requirement is nicely met by our method, underlying the good performance of our method in real-world applications. It is noticed that the low processing speed is likely a drawback of our method, due to the model interaction mechanism we added. However, according to the Figs. 6–10 and Tables 1–5, it can be worth the increase in time cost of our method since a significant improvement of the object detection performance is achieved. Moreover, according to the experimental results, the performance of our method and BF-KDE method outperform the ST-MoG method. This implies that the background-foreground fusion strategy is much better than the spatio-temporal fusion strategy for moving object detection in dynamic scenes.

8. Conclusions

In this paper, an interaction mechanism between the background and foreground models is proposed along with our weighted KDE model. In our approach, the weighted KDE based background model holds the long-term state of scenes, while the foreground model adapts to the short-term scene changes. Through interactions between the background and foreground models, our approach has a robust capability to detect moving objects in dynamic scenes. However, the drawback of our approach is the slightly higher computational cost. In our further work, we will explore the potential of the deep learning approach to handle dynamic scenes, generating more stable and accurate object detection results. Specifically, we will test the recently proposed Gabor Convolutional Network [29] in background modelling, as this novel network has been proven robust against scale and rotation changes.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (nos. 61671201, 61501173), Fundamental Research Funds for the Central Universities (no. 2017B01914) and Natural Science Foundation of Zhejiang Province (no. LY18F010008).

References

[1] P.V.K. Borges, N. Conci, A. Cavallaro, Video-based human behavior understanding: a survey, IEEE Trans. Circuits Syst. Video Technol. 23 (11) (2013) 1993–2008.
[2] Y. Chen, D. Zhao, L. Lv, Q. Zhang, Multi-task learning for dangerous object detection in autonomous driving, Inf. Sci. 432 (2018) 559–571.
[3] A. Monnet, A. Mittal, N. Paragios, V. Ramesh, Background modeling and subtraction of dynamic scenes, in: IEEE International Conference on Computer Vision, 2003, pp. 1305–1312.
[4] C. Stauffer, W.E.L. Grimson, Adaptive background mixture models for real-time tracking, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, pp. 246–252.
[5] X. Chen, Y. Gao, R. Wang, Online selective kernel-based temporal difference learning, IEEE Trans. Neural Networks Learn. Syst. 24 (12) (2013) 1944–1956.
[6] L. Li, W. Huang, Y. Gu, Q. Tian, Statistical modeling of complex backgrounds for foreground object detection, IEEE Trans. Image Process. 13 (11) (2004) 1459–1472.
[7] B. Zhang, Y. Gao, S. Zhao, B. Zhong, Kernel similarity modeling of texture pattern flow for motion detection in complex background, IEEE Trans. Circuits Syst. Video Technol. 21 (1) (2011) 29–38.
[8] O. Banos, S. Lee, B.H. Kang, E.S. Kim, T. Le-Tien, NIC: a robust background extraction algorithm for foreground detection in dynamic scenes, IEEE Trans. Circuits Syst. Video Technol. 27 (7) (2017) 1478–1490.
[9] T. Akilan, Q.J. Wu, Y. Yang, Fusion-based foreground enhancement for background subtraction using multivariate multi-model Gaussian distribution, Inf. Sci. 430 (2018) 414–431.
[10] J. Rittscher, J. Kato, S. Joga, A. Blake, A probabilistic background model for tracking, in: European Conference on Computer Vision, 2000, pp. 336–350.
[11] A. Forestiero, Self-organizing anomaly detection in data streams, Inf. Sci. 373 (2016) 321–336.
[12] C. Nikou, N.P. Galatsanos, A.C. Likas, A class-adaptive spatially variant mixture model for image segmentation, IEEE Trans. Image Process. 16 (4) (2007) 1121–1130.
[13] O. Tuzel, F. Porikli, P. Meer, A baysian approach to background modeling, in: CVPR Workshops in Computer Vision and Pattern Recognition, 2005, pp. 58–63.
[14] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 936–944.
[15] T.M.P. Ngoc, Adaptive optimal kernel density estimation for directional data. (2018) arXiv:1808, 02361.
[16] O. Barnich, M.V. Droogenbroeck, ViBe: a universal background subtraction algorithm for video sequences, IEEE Trans. Image Process. 20 (6) (2011) 1709–1724.
[17] J.M. Guo, C.H. Hsia, Y.F. Liu, M.H. Shih, C.H. Chang, J.Y. Wu, Fast background subtraction based on a multilayer codebook model for moving object detection, IEEE Trans. Circuits Syst. Video Technol. 23 (10) (2013) 1809–1821.
[18] A. Elgammal, R. Duraiswami, D. Harwood, L.S. Davis, Background and foreground modeling using nonparametric kernel density estimation for visual surveillance, Proc. IEEE 90 (2002) 1151–1163.
[19] T. Bouwmans, E.H. Zahzah, Robust PCA via principal component pursuit: a review for a comparative evaluation in video surveillance, Comput. Vision Image Understanding 122 (2014) 22–34.
[20] M. Wu, X. Peng, Spatio-temporal context for codebook-based dynamic background subtraction, AEU-Int. J. Electron. Commun. 64 (8) (2010) 739–747.
[21] D.M. Tsai, S.C. Lai, Independent component analysis-based background subtraction for indoor surveillance, IEEE Trans. Image Process. 18 (1) (2009) 158–167.
[22] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. Huang, S. Yan, Sparse representation for computer vision and pattern recognition, in: Proceedings of the IEEE, 2010, pp. 1031–1044.
[23] A.W. Smeulders, D.M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, M. Shah, Visual tracking: an experimental survey, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2014) 1442–1468.
[24] R.J. Bessa, V. Miranda, A. Botterud, J. Wang, E.M. Constantinescu, Time adaptive conditional kernel density estimation for wind power forecasting, IEEE Trans. Sustain. Energy 3 (4) (2012) 660–669.
[25] M.H. Yang, C.R. Huang, W.C. Liu, S.Z. Lin, K.T. Chuang, Binary descriptor based nonparametric background modeling for foreground extraction by using detection theory, IEEE Trans. Circuits Syst. Video Technol. 25 (4) (2015) 595–608.
[26] F. Cao, Y. Liu, D. Wang, Efficient saliency detection using convolutional neural networks with feature selection, Inf. Sci. 456 (2018) 34–49.
[27] Y. Chen, W. Li, C. Sakaridis, D. Dai, L. Van Gool, Domain adaptive faster R-CNN for object detection in the wild, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3339–3348.
[28] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 3431–3440.
[29] S. Luan, C. Chen, B. Zhang, J. Han, J. Liu, Gabor convolutional networks, IEEE Trans. Image Process. 27 (9) (2018) 4357–4366.
[30] M. Piccardi, Background subtraction techniques: a review, in: IEEE International Conference on Systems, Man and Cybernetics, 2004, pp. 3099–3104.
[31] K. Fragkiadaki, P. Arbeláez, P. Felsen, J. Malik, Learning to segment moving objects in videos, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 4083–4090.
[32] B. Xin, Y. Tian, Y. Wang, W. Gao, Background subtraction via generalized fused lasso foreground modeling, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 4676–4684.
[33] J.Y. Hao, C. Li, Z. Kim, Z. Xiong, Spatio-temporal traffic scene modeling for object motion detection, IEEE Trans. Intell. Trans. Syst. 14 (2013) 295–302.
[34] B.W. Silverman, Density Estimation for Statistics and Data Analysis, Routledge, Australasia, 2018.
[35] J. Zhong, S. Sclaroff, Segmenting foreground objects from a dynamic textured background via a robust Kalman filter, in: Ninth IEEE International Conference on Computer Vision, 2003, pp. 44–50.
[36] M.S. Arulampalam, S. Maskel, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2002) 174–188.
[37] X. Ren, D. Ramanan, Histograms of sparse codes for object detection, in: IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 3246–3253.
[38] X. Zhang, C. Zhu, S. Wang, Y. Liu, M. Ye, A Bayesian approach to camouflaged moving object detection, IEEE Trans. Circuits Syst. Video Technol. 27 (9) (2017) 2001–2013.
[39] Z. Chen, X. Wang, Z. Sun, Z. Wang, Motion saliency detection using a temporal Fourier transform, Opt. Laser Technol. 80 (2016) 1–15.
[40] A. Yilmaz, O. Javed, M. Shah, Object tracking: a survey, ACM Comput. Surv. (CSUR) 38 (2006) 1–45.
[41] H. Zhang, X. Bai, J. Zhou, J. Cheng, H. Zhao, Object detection via structural feature selection and shape model, IEEE Trans. Image Process. 22 (12) (2013) 4984–4995.

[42] G. Kitagawa, Monte Carlo filter and smoother for non-Gaussian nonlinear state space models, J. Comput. Graphical Stat. 5 (1) (1996) 1–25.
[43] C. Lacour, P. Massart, V. Rivoirard, Estimator selection: a new method with applications to kernel density estimation, Sankhya A 79 (2) (2017) 298–335.
[44] M. Everingham, L.J.V. Gool, C.K.I. Williams, J.M. Winn, A. Zisserman, The Pascal visual object classes VOC challenge, Int. J. Comput. Vision 88 (2) (2010) 303–338.
[45] CAVIAR. [Online]. Available: http://www-prima.inrialpes.fr/PETS04/caviar_data.html.
[46] ALOV++. [Online]. Available: http://www.alov300.org.
[47] WallFlower. [Online]. Avaliable: http://research.microsoft.com/en-us/um/people/jckrumm/WallFlower/TestImages.htm.
[48] Shadow. [Online]. Avaliable: http://cvrr.ucsd.edu/aton/shadow/.
[49] S.D. Babacan, T.N. Pappas, Spatiotemporal algorithm for background subtraction, in: International Conference on Acoustics, Speech and Signal Processing, 2007, pp. 1065–1068.
[50] X. Cui, Q. Liu, S. Zhang, F. Yang, D.N. Metaxas, Temporal Spectral Residual for fast salient motion detection, Neurocomputing 86 (2012) 24–32.
[51] H. Wei, C. Yang, Q. Yu, Efficient graph-based search for object detection, Inf. Sci. 385 (2017) 395–414.