Neurocomputing 155 (2015) 84–98

Contents lists available at ScienceDirect

Neurocomputing

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

Summarizing surveillance videos with local-patch-learning-based abnormality detection, blob sequence optimization, and type-based synopsis

Weiyao Lin [a,*], Yihao Zhang [a], Jiwen Lu [b], Bing Zhou [c], Jinjun Wang [d], Yu Zhou [e]

[a] Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China
[b] Advanced Digital Sciences Center, Singapore, Singapore
[c] School of Information Engineering, Zhengzhou University, Zhongyuan, China
[d] Institute of Artificial Intelligence and Robotics, Xi'an Jiao Tong University, Xi'an, China
[e] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 4 October 2014
Received in revised form
7 December 2014
Accepted 17 December 2014
Communicated by Tao Mei
Available online 29 December 2014

Keywords:
Video synopsis
Blob sequence optimization
Abnormality detection

ABSTRACT

In this paper, we propose a new approach to detect abnormal activities in surveillance videos and create suitable summary videos accordingly. The proposed approach first introduces a patch-based method to automatically model normal activity patterns and key regions in a scene. In this way, abnormal activities can be effectively detected and classified from the modeled normal patterns and key regions. Then, a blob sequence optimization process is proposed which integrates spatial, temporal, size, and motion correlation among objects to extract suitable foreground blob sequences for abnormal objects. With this process, blob extraction errors due to occlusion or background interference can be effectively avoided. Finally, we also propose an abnormality-type-based method which creates short-period summary videos from long-period input surveillance videos by properly arranging abnormal blob sequences according to their activity types. Experimental results show that our proposed approach can effectively create satisfying summary videos from input surveillance videos.

1. Introduction

Video surveillance is of increasing importance in many applications including traffic control, unusual alarming [1–6]. In many scenarios, people need to browse videos to find events of interest or perform analysis. However, since surveillance videos are usually long, it is laborious to watch the entire videos. Thus, it is essential to create short-period summary (or abstract/synopsis) videos which summarize important events in long-period surveillance videos. In this way, people's labor can be greatly saved by only watching these short summary videos [1,2,14,38]. Therefore, in this paper, we focus on creating suitable summary videos for input surveillance videos.

First, since most people are interested in abnormal activities in surveillance videos, detecting abnormalities in videos is crucial in analyzing and summarizing surveillance videos. Many algorithms have been proposed on abnormality detection [3,5,6,17–19,27–36]. However, most of these works only focus on detecting abnormalities while the differentiation of abnormality types is seldom addressed. In practice, differentiating abnormal activity types is important in creating well-organized summary videos.

Second, it is also important to extract accurate foreground blob[1] sequences for objects such that objects can be suitably separated and arranged to create satisfying summary videos. Although many tracking algorithms have been proposed [7–10,12,16,37], their performances are still less satisfactory due to the interferences from object occlusion or complex background. Besides, most tracking-based methods only focus on achieving object bounding boxes while the suitable segmentation of object blobs is not addressed. In practice, achieving accurate object blob is non-trival in creating satisfying summary videos.

Third, creating suitable summary videos from long surveillance videos is another key issue. Recently, video synopsis methods [1,2,24] were proposed which extracted and put together object

* Corresponding author. Tel.: +86 21 34208843; fax: +86 21 34204155.
E-mail address: hellomikelin@gmail.com (W. Lin).

[1] In this paper, a blob refers to a connected foreground region for one or several objects [10], as in Fig. 4(c).

http://dx.doi.org/10.1016/j.neucom.2014.12.044
0925-2312/

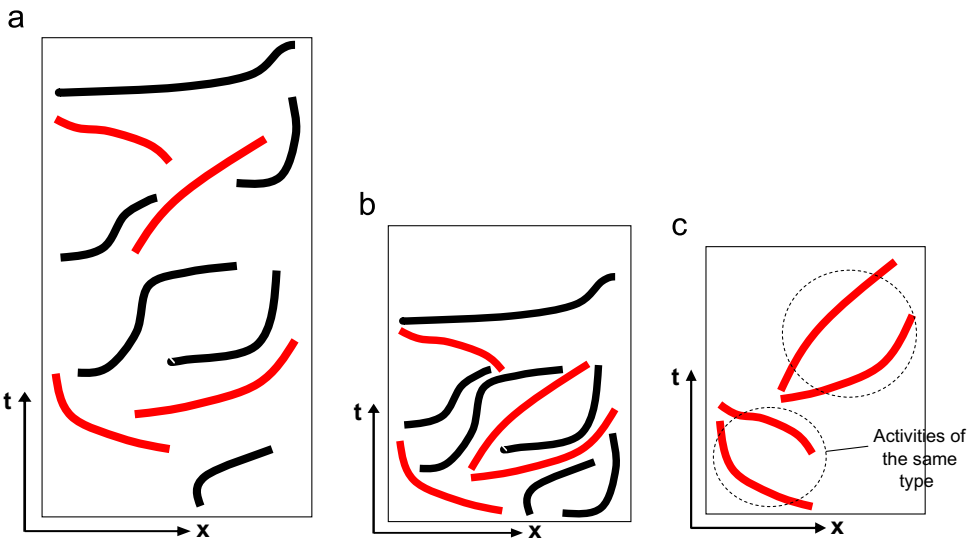


Fig. 1. (a): An input video including object trajectories (i.e., blob sequences); (b): the synopsis video by [1] which moves and puts together object trajectories from different periods; (c): the synopsis video by our approach which only performs synopsis on abnormal trajectories and put together trajectories of the same type, i.e., put together trajectories which start from the same region and end in the same regions into the same time period. (*Note*: *t* represents the time domain and *x* represents the spatial domain of a video. The red lines represent abnormal trajectories and the black lines represent normal trajectories, best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

blob sequences from different periods, as in Fig. 1(b). Video synopsis has the advantage of creating short summary videos while suitably maintaining all blob sequences of interest. However, the existing synopsis methods still have the following limitations: (1) They are still less effective in summarizing crowd-scene videos where the huge number of blob sequences will make the synopsis videos chaotic and less understandable. (2) They only focus on compressing the length of videos while seldom consider the proper arrangement of similar activity types.

In this paper, we propose a new approach to detect abnormal activities from surveillance videos and create suitable summary videos accordingly. The contributions our approach can be summarized as follows:

(1) We introduce a patch-based method to automatically model normal activity patterns and use them to detect abnormal activities. Besides, based on the observation that each scene should include "key regions" and all activity trajectories in a scene should go through part of them (as in Fig. 2), we also propose to extract key regions from a scene and use them to classify abnormal activities into different types (i.e., activities are classified into the same type when they pass through the same key regions, as in Fig. 2). By introducing key regions, we are not only able to improve the abnormality detection accuracy, but are also able to organize abnormalities into different types which enables the creation of well-organized summary videos in later steps.
(2) Based on the assumption that most people are interested in abnormal activities in surveillance videos, we propose an abnormality-type-based video synopsis method which summarizes surveillance videos by only synopsizing over abnormal blob sequences. Moreover, the proposed method further introduces an activity-type cost during the synopsis process such that blob sequences of the same activity type (i.e., activities passing through the same key regions) can be arranged closely in summary videos. With this method, we are able to create well-organized summary videos even for crowded scenes, as in Fig. 1(c).
(3) We also propose a blob sequence optimization process which integrates spatial, temporal, size, and motion correlation among objects to extract suitable foreground blob sequences for abnormal objects. With this process, blob extraction errors due to occlusion or background interferences can be effectively avoided.

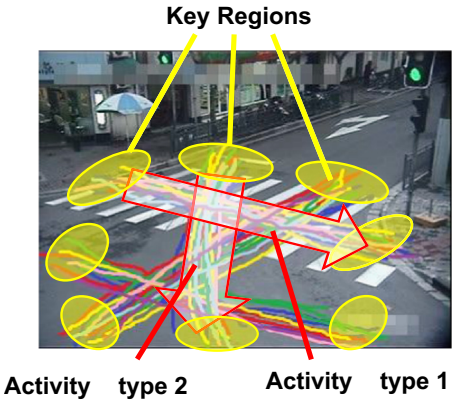


Fig. 2. An example of key regions (yellow circles) and activity types (red arrows). Key regions refer to regions in a scene where all activity trajectories in the scene should go through part of them (e.g., cluster of trajectory terminals), and activity types refer to activity classes whose trajectories pass through the same key regions (best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The rest of the paper is organized as follows: Section 2 discusses the related work. Section 3 describes the framework of our proposed approach. Sections 4–6 describe the details of our approach. Section 7 shows the experimental results. Section 8 concludes the paper.

2. Related works

Since abnormal activity detection is one of the most important issues in surveillance video analysis, it has attracted a lot of research works [3,5,6,17–19,27–36]. Many people detected abnormalities by parsing the motion trajectories of objects. For example, Zelniker

et al. [3] created global trajectories through tracking and detected abnormal activities if the current global trajectory deviates from the normal paths. Rao et al. [19] further introduced view-invariant dynamic time warping to recognize trajectories with different variations. Lin et al. [5] calculated network-transmission energies to detect abnormal trajectories. Morris et al. [27] and Wang et al. [32] introduced graphical models to model the patterns of normal trajectory classes, and used these patterns to predict abnormal trajectories. Zhou et al. [28] first introduced 3D tubes to embed the motion and normality of trajectories and then derived a droplet from these 3D tubes for detecting abnormalities. Kim et al. [18] introduced Gaussian process flows to model the location and velocity probability for each point in a trajectory and detected abnormalities by measuring the unlikelihood against normal patterns. However, most of these works only focus on detecting abnormalities while the differentiation of abnormality types is seldom addressed. In practice, differentiating abnormal activity types is important in creating well-organized summary videos. Furthermore, besides using motion trajectories for abnormality detection, other researches tried to localize abnormalities by analyzing the local or global motion flow fields. For example, Mehran [34] and Cui [35] derived social-force-based features from input optical flow fields to detect abnormal crowd activities. Kim and Grauman [17] utilized a mixture of probabilistic principal component analyzers (MPPCA) to learn normal patterns of activities for local patches in a scene and inferred a space-time Markov random field (MRF) to detect abnormal activities. Cong et al. [22] introduced a sparse reconstruction cost (SRC) over the normal dictionary to measure the normality, and use it to detect local and global abnormalities in a scene. Although these methods can effectively localize abnormalities in a frame, they do not include temporal correlation among the abnormal regions in different frames. Thus, they are less suitable for video summary applications which require the extraction of object blob sequences.

Furthermore, in order to include object activities into a summary video, it is also important to accurately extract foreground blob sequences for objects. Many object tracking methods have been proposed to achieve object trajectories. For example, Kalal et al. [8] utilized an online object detector and integrated it with an object tracker to achieve improved tracking performance. Zhang et al. [9] constructed compressed vectors from multi-scale image features and used these vectors to train a classifier to identify objects being tracked. Bleme et al. [37] and Henriques et al. [21] introduced correlation filters to model the pattern of objects being tracked. In recent years, some multi-object tracking algorithms [12,16] were also proposed which improved tracking performances by including trajectory association correlations among objects. However, the existing tracking methods still have limitations under severe occlusion or background interferences. Besides, most tracking-based methods only focus on achieving object bounding boxes while the suitable segmentation of object blobs is not addressed.

Moreover, creating suitable summary videos is another key issue in summarizing surveillance videos. Many video summary methods [14,20] extracted key frames from the original videos and concatenated them together. For example, Lee et al. [26] developed region cues indicative of high-level saliency in egocentric video and used them to predict the relative importance of any new region. And based on these predictions, key frames can be selected to construct summary videos. Ngo et al. [23] introduced a motion attention model to evaluate the importance of different scenes and frames, and utilized a temporal graph to hierarchically summarize videos from the scene level, cluster level, shot level up to subshots level. However, only extracting static key frames from videos cannot maintain the dynamic patterns of events in the summary video. Although some methods extracted important video clips instead of frames to keep the temporal information [25], their summary efficiency will be poor if there exist important events in each frame of a video. Besides, since the important video clips may include both important events and less important events, simply concatenating these video clips will also include various less important events in the summary video. In recent years, Pritch et al. [1,24] proposed a video synopsis approach which extracted and put together object blob sequences from different periods. Nie et al. [2] further improved the efficiency of video synopsis by utilizing global shift optimization to avoid motion collision among blob sequences. These synopsis-based video summary methods have the advantage of creating short summary videos while suitably maintaining all blob sequences of interest. However, as mentioned, they still have limitations in summarizing crowded videos and in properly arranging activities of the same type.

3. Overview of the approach

The framework of our approach is shown in Fig. 3. In Fig. 3, the two modules “normal pattern learning” and “key patch extraction and correlation modeling” in the “patch-based training” stage first learn normal activity patterns and model key region correlations in a scene from the training videos offline. Then, for an input test surveillance video, candidate abnormal blob sequences are first detected according to the learned normal activity patterns. After that, the blob sequence optimization process is applied to delete blob extraction errors in the previous step for achieving more precise blob sequences. Then, the

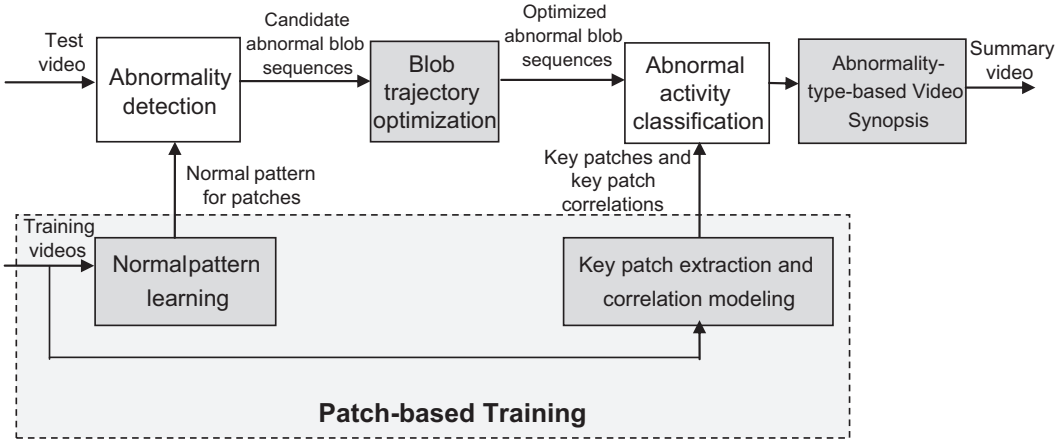


Fig. 3. Framework of the proposed approach.

detected abnormal blob sequences are further classified into different types according to the key regions decided from the "key patch extraction and correlation modeling" module. Finally, an abnormality-type-based method is applied on the classified abnormal blob sequences to create a summary video which properly arranges and displays abnormalities from the input video.

In the following, we will describe the details of our approach. Note that the modules of "normal pattern learning", "key patch extraction and correlation modeling", "blob sequence optimization", and "abnormality-type-based video synopsis" are the key parts of our approach.

4. Patch-based training method

As mentioned, in order to create satisfactory summary videos for abnormal activities, we need to (1) accurately detect abnormalities in an input video, and (2) classify the detected abnormalities such that abnormalities of the same type can be suitably organized in the summary video.

Therefore, in this paper, we propose a patch-based training method which addresses the above requirements by two processes: "learning normal activity patterns" and "modeling key regions and their correlations". They are described in the following.

4.1. Learning normal activity patterns

In this process, we first divide an input scene into identical rectangular local patches, as in Fig. 4(a), and then learn a normal activity pattern for each local patch, respectively. More specifically, we first extract all blob sequences for normal activities from the training data where each blob sequence contains foreground blobs of an object located at different patches during the object's normal activity period. And all blobs from these normal activity blob sequences can construct a normal blob training set. Then, for each local patch P_B, we extract motion descriptor features [15] for blobs which are located at P_B in the normal blob training set. And based on these motion descriptor features, we further utilize kernel density estimation (KDE) [13] to construct a normal pattern probability density $\mu_{P_B}(\cdot)$ for patch P_B, and this normal pattern probability density will be used as the normal activity pattern for P_B. Basically, a larger $\mu_{P_B}(\cdot)$ implies higher normality likelihood in P_B and a smaller $\mu_{P_B}(\cdot)$ implies smaller normality likelihood (or higher abnormality likelihood).

Then, in the testing stage, for an input blob sequence $R(u, q)$, we first calculate the motion descriptor feature $\mathbf{MD}(B, P_B)$ for each blob B in the blob sequence $R(u, q)$. Then, the normal pattern probability density function $\mu_{P_B}(\cdot)$ learned from the "normal pattern learning" process is used to evaluate the normality likelihood of each blob (i.e., $\mu_{PB}(\mathbf{MD}(B, P_B))$). And if the overall normality likelihood for a blob sequence $R(u, q)$ is smaller than a threshold, it implies that $R(u, q)$ is different from the trained normal patterns. Thus, $R(u, q)$ will be detected as a candidate abnormal activity, as Eq. (1):

$$R(u,q)\text{is abnormal if } \frac{1}{|R(u,q)|}\sum\nolimits_{B\in R(u,q)}\mu_{P_B}(\mathbf{MD}(B,P_B)) \le \tau_a \tag{1}$$

where $R(u, q)$ is a blob sequence which starts from point u and ends at q. B is a blob in $R(u, q)$ and P_B is the patch that B is located in. $\mathbf{MD}(B, P_B)$ is the motion descriptor feature of $R(u, q)$'s blob B at patch P_B. In our experiments, histogram of oriented optical flow (HOOF) [15] is utilized as the motion descriptor. $|R(u, q)|$ is the length of blob sequence $R(u, q)$. And $\mu_{P_B}(\cdot)$ is the normal pattern probability density function for patch P_B. τ_a is a threshold whose value can be decided by minimizing the total normality/abnormality classification error in the training set, as in Eq. (2).

$$\tau_a = \underset{\hat{\tau}_a}{\text{argmin}}\,\text{TEF}(\hat{\tau}_a) \tag{2}$$

where τ_a is the final decided threshold value and $\hat{\tau}_a$ is one candidate value. $\text{TEF} = (N_{\text{error,ab}} + N_{\text{error,nor}}/N_{\text{total}})$ is the total normality/abnormality classification error where $N_{\text{error,ab}}$, $N_{\text{error,nor}}$, and N_{total} are the number of mis-detected abnormalities, mis-detected normalities, and all blob sequences in the training data, respectively. From Eq. (2), we can see that in order to decide the value of τ_a, we just try different τ_a values to classify the normal and abnormal blob sequences in the training set by the method in Eq. (1). And the value with the smallest classification error TEF will be selected as the final threshold τ_a.

From Eq. (1), we can see that we model a normal activity pattern based on each patch. Compared with the methods which model normal activity patterns over trajectory clusters [6,18,19], our method can have stronger capability in handling activities with large variations. For example, when the trajectories of normal activities have large variations, the trajectory-cluster-based methods [6,18,19] may not be able to construct reliable models for normal patterns. However, by modeling activity patterns over patches, we may still be able to construct suitable models by catching the local activity pattern within each patch.

4.2. Modeling key regions and their correlations

As mentioned, properly classifying abnormal activities can greatly facilitate the creation of well-organized synopsis videos. However, abnormal activities are difficult to be classified due to their large variations and uncertainties. To address this problem, we argue that each scene should include key regions and all activity trajectories in the scene should go through part of them. For example, in Fig. 5, all trajectories in the scene, no matter normal or abnormal, should go through some of the yellow-circled

Fig. 4. (a) Segmenting the scene into patches, (b) a frame of an input test video, (c) the detected abnormal blob in (b).

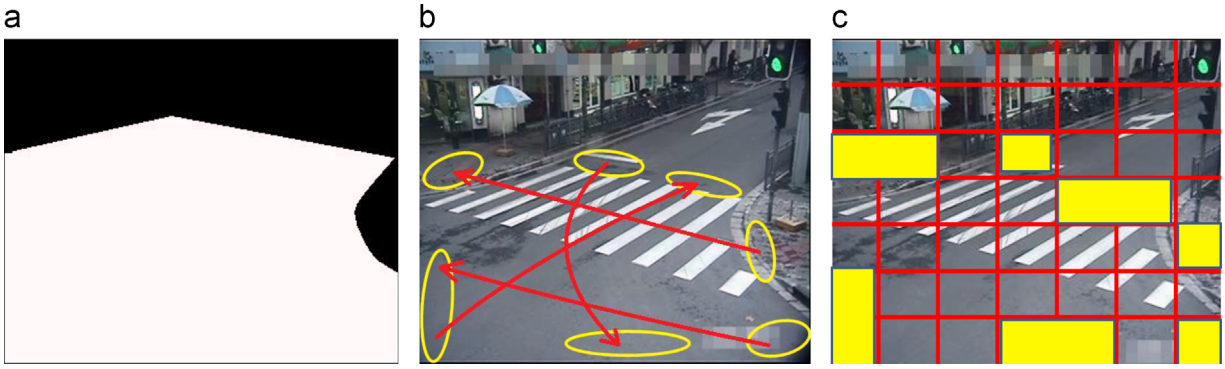


Fig. 5. (a) A region-of-interest (ROI) in a scene. (b) Key regions inside an ROI. (c) Key patches corresponding to key regions in (b).

key regions when they enter or leave the region-of-interest (ROI)[2]. Thus, by extracting and modeling the key regions, the problem of abnormality classification can be simplified to the analysis of key regions that the abnormal object passes.

Based on the above discussion, we propose a new process to detect and model key regions for abnormality classification. The proposed process includes three-steps:

Step 1: *Detect key patches.* In this step, key patches in a scene are detected to represent key regions. We first cluster the trajectory terminals in the training data to achieve key regions. Then, patches which have large overlap with the key regions are detected as key patches, as in Fig. 5.

Step 2: *Model the correlation among key patches.* In this step, the activity correlations among key patches are calculated by:

$$S(P_i,\ P_j) = \#\{R(u,q) | u \in P_i, q \in P_j\} \tag{3}$$

where $S(P_i, P_j)$ is the correlation between key patches P_i and P_j. $R(u, q)$ is a trajectory (i.e., blob sequence) in the training set which starts at point u and ends at point q. And $\#\{\bullet\}$ is the number of elements in a set. From Eq. (3), the correlations among key patches are calculated by the total number of trajectories passing these patches in the training set. In this way, a large $S(P_i, P_j)$ implies a normal activity pattern between patches, while a small $S(P_i, P_j)$ implies that going from P_i to P_j is abnormal.

Step 3: *Classify abnormal trajectories.* After achieving key patches and their correlations, we can use them to classify abnormal trajectories in testing videos. A candidate abnormal blob sequence $R(u, q)$ can be classified as abnormal activity type $A^*(P_m, P_n)$ if:

$$\begin{cases} A^*(P_m,P_n) = \arg \min\limits_{A(P_i,P_j),P_i,P_j \in \mathbf{KP}} S(P_i,P_j) \times D\big(R(u,q),A(P_i,P_j)\big) \\ S(P_m,P_n) \le \tau_t \end{cases} \tag{4}$$

where $S(P_i, P_j)$ is the key patch correlation by Eq. (3). $A(P_i, P_j)=[P_i, P_j]^T$ is a candidate activity type which includes trajectories start from patch P_i and end at P_j. $A^*(P_m, P_n)=[P_m, P_n]^T$ is the final classified activity type for input trajectory $R(u, q)$. **KP** is the set of key patches decided in Step 1. τ_t Is a threshold which can be decided in a similar way as Eq. (2). $D(R(u, q), A(P_i, P_j))$ is the dissimilarity between the input blob sequence $R(u, q)$ and the activity type $A(P_i, P_j)$ and is calculated by:

$$D\big(R(u,q),A(P_i,P_j)\big) = d(P_u,P_i) \times d\big(P_q,P_j\big) \tag{5}$$

where $d(P_u, P_i)$ is the distance between patches P_u and P_i. And P_u and P_q are the patches where $R(u, q)$'s starting point u and finishing point q are located, respectively.

From Eqs. (4) and (5), we can see that an abnormal blob sequence $R(u, q)$ will be classified as type $A^*(P_m, P_n)$ if its terminal patches P_u and P_q are close to the key patch pair P_m and P_n of this type. Besides, the patch correlation term $S(P_m, P_n)$ is also included to guarantee that $A^*(P_m, P_n)$ is an abnormal type (i.e., $S(P_m, P_n)$ is small). In this way, we can filter out wrong candidate abnormal activities in the previous process (i.e., Eq. (1)) and achieve more precise results. Therefore, with the introduction of key regions, we can not only organize abnormalities into different types to facilitate the creation of well-organized summary videos in later steps, but also effectively improve the detection accuracy by filtering out wrong candidate abnormal activities.

5. Abnormal blob sequence extraction and optimization

After modeling normal activity patterns and key patches for a scene, we are able to extract abnormal blob sequences from the input surveillance videos. The process of abnormal blob sequence extraction includes three major modules:

5.1. Abnormality detection

In this module, we first extract blob sequences for all objects in an input video and then select blob sequences with abnormal patterns as the candidate abnormal blob sequences.

More specifically, in this paper, we first perform foreground extraction to achieve object foreground blobs [10], then foreground blobs in neighboring frames are associated to construct blob sequences. In this paper, we simply associate blobs which have both large overlapping areas and similar Histogram of Gradient (HOG) features [10]. That is,

$$B_t^i \text{ and } B_{t-1}^j \text{ will be associated if} \\ : \frac{\text{size}\left(B_t^i \cap B_{t-1}^j\right)}{\text{size}\left(B_t^i \cup B_{t-1}^j\right)} > 0.5 \text{ and } \text{SIM}_{HI}\left(\mathbf{H}\left(B_t^i\right), \mathbf{H}\left(B_{t-1}^j\right)\right) > 0.5 \tag{6}$$

where B_t^i and B_{t-1}^j represents the ith and jth blob in frame t and frame $t-1$, respectively. $size(B)$ is the size of blob B. $\mathbf{H}(B)$ is the HOG feature for blob B, and $\text{SIM}_{HI}\left(\mathbf{H}\left(B_t^i\right), \mathbf{H}\left(B_{t-1}^j\right)\right)$ is the histogram intersection similarity [31] between HOG features for blobs B_t^i and B_{t-1}^j.

Finally, a blob sequence is detected as a candidate abnormal sequence if it does not fit the normal pattern, as by Eq. (1).

5.2. Blob sequence optimization

As mentioned, the candidate abnormal sequences achieved by Section 5.1 are still less accurate. For example, in Figs. 6 and 7, the detected blob sequences wrongly include multiple trajectories or

[2] Note that in our experiments, we define manually the ROI of each scene. However, in practice, we can also include automatic ROI extraction methods [29,30] to automatically identify ROIs.

unwanted normal objects due to the interference of occlusion and complex foregrounds. Therefore, we further propose a blob sequence optimization module which integrates spatial, temporal, size, and motion correlation among objects to extract suitable abnormal blob sequences. The proposed blob optimization module includes the following three steps.

Step 1: *Detect problematic blobs.* In the first step, we detect problematic blobs which are wrongly segmented and need to be processed (such as the blobs in Figs. 6 and 7(b)). Based on the observation that most problematic blobs have unusually large sizes due to the inclusion of other objects or foregrounds, we propose to detect problematic blobs by:

$$B_t \in R \text{ is a problematic blob if } \text{size}(B_t) \geq \omega \times \text{size}_R^m \tag{7}$$

where R is a blob sequence. size(B) is the size of blob B. $\text{size}_R^m = \underset{B_v \in R}{\text{med}}(\text{size}(B_v))$ is the median over all blob sizes in blob sequence R where med($\bullet$) is the median operator and B_v is a blob belonging to blob sequence R. $\omega = 1.5$ is a constant factor and its value is decided in the same way as Eq. (2). From Eq. (7), we can see that a problematic blob will be detected if its size is obviously larger than the median blob size size_R^m in the blob sequence. Furthermore, in order to avoid size differences due to camera perspective effect, camera calibration [4] is utilized to normalize blob sizes at different locations.

Step 2: *Problematic blob optimization.* In this step, we propose to construct and minimize a segmentation cost function to optimally segment the problematic blob into suitable blobs, as:

$$\mathbf{\Psi}^* = \arg\ \min_{\mathbf{\Psi}} \lambda_1 \times C_{\text{intra}}(\mathbf{\Psi}) + \lambda_2 \times C_{\text{inter}}(\mathbf{\Psi}) + \lambda_3 \times C_{\text{temp}}(\mathbf{\Psi}) \tag{8}$$

where $\mathbf{\Psi} = \{B_{t,1}, B_{t,2}, \ldots, B_{t,Nt}\}$ is a candidate segmentation result of the problematic blob B_t and $B_{t,k} \in B_t$ is a segmented sub-blob in B_t, as in Fig. 8. N_t is the total number of sub-blobs in B_t. λ_1, λ_2, and λ_3 ($\lambda_1+\lambda_2+\lambda_3=1$) are the weights. C_{intra}, C_{inter}, and C_{temp} are the intra, inter, and temporal costs measuring the segmentation suitability. These costs are described in Eqs. (9)–(12), respectively.

$$C_{\text{intra}} = \sum_{k=1}^{N_t} \left(\frac{1}{\text{size}(B_{t,k})} \sum_{(x_i,y_i) \in B_{t,k}} \| \vec{f}'(x_i,y_i) - \overline{\vec{f}'_{B_{t,k}}} \|^2 \right) \tag{9}$$

where $\vec{f}'(x_i,y_i)$ is the "coherent" motion flow for pixel (x_i, y_i) and $\overline{\vec{f}'_{B_{t,k}}}$ is the average coherent motion flow in sub-blob $B_{t,k}$. $\vec{f}'(x_i,y_i)$ and $\overline{\vec{f}'_{B_{tk}}}$ are calculated by:

$$\begin{cases} \vec{f}'(x_i,y_i)|_{(x_i,y_i) \in b_z} = \frac{1}{\text{size}(b_z)} \sum\limits_{(x_j,y_j) \in b_z} \vec{f}\left(x_j,y_j\right) \\ \overline{\vec{f}'_{B_{t,k}}} = \frac{1}{\text{size}(B_{t,k})} \sum\limits_{(x_j,y_j) \in B_{t,k}} \vec{f}'\left(x_j,y_j\right) \end{cases} . \tag{10}$$

where $\vec{f}(x_i,y_i)$ is the original optical flow for pixel (x_i, y_i) and b_z is a similar-flow cluster in the blob including pixels of similar optical flows [11], as in Fig. 8. From Eqs. (9) and (10), we can see that the intra cost C_{intra} is calculated by the average motion flow variances within each sub-blob and C_{intra} will be minimized when motion flows in each sub-blob have small variance. Moreover, in order to avoid the interference from noisy optical flows, we also propose to decompose the problematic blob into similar-flow clusters and use the cluster's coherent flow $\vec{f}'(x_i,y_i)$ to represent pixel motion flows instead of the original flows.

$$C_{\text{inter}} = -\frac{1}{N_t} \sum_{k=1}^{N_t} \| \overline{\vec{f}'_{B_{t,k}}} - \overline{\vec{f}'_{B_t}} \|^2 \tag{11}$$

where N_t is the total number of sub-blobs in the current problematic blob B_t. $\overline{\vec{f}'_{B_t}}$ is the average coherent motion flow

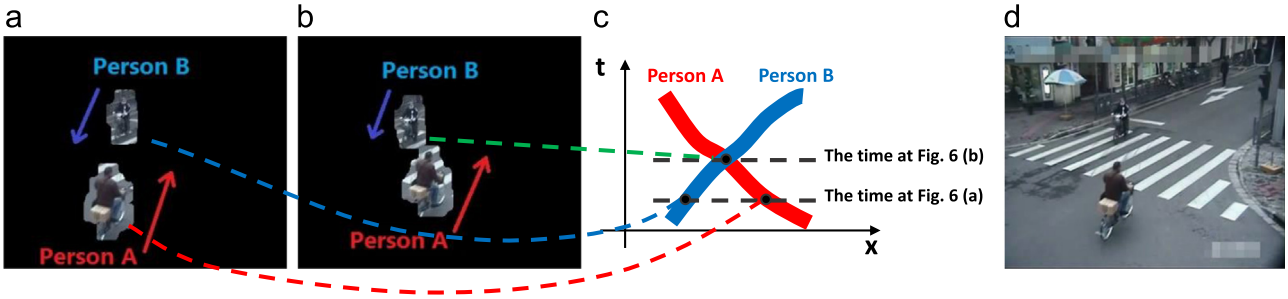


Fig. 6. (a) and (b): Blobs detected at different frames; (c) the two abnormal trajectories are wrongly connected into the same blob sequence due to occlusion in (b); (d) The original scene (since the green light on the upright is off, both trajectories are abnormal since they are crossing the red lights, best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

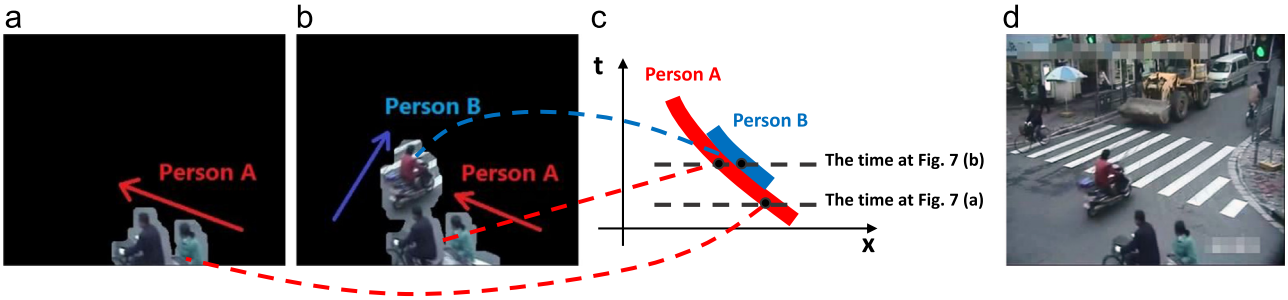


Fig. 7. (a) and (b) Blobs detected at different frames; (c) the blob sequence wrongly includes normal objects due to foreground connection in (b); (d) the original scene (since the green light on the upright is on, the blue trajectory is normal while the red trajectory is abnormal, best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

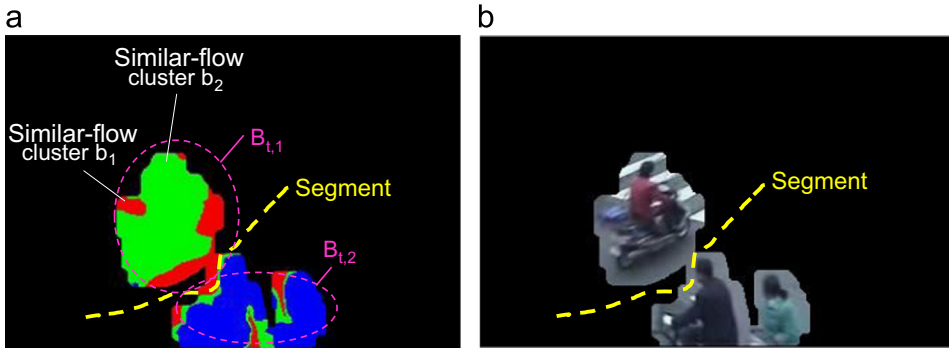


Fig. 8. An example of blob segmentation: (a) a problematic blob is composed of similar-flow clusters and can be segmented into sub-blobs according to these similar-flow clusters; (b) the segmentation result. (Best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

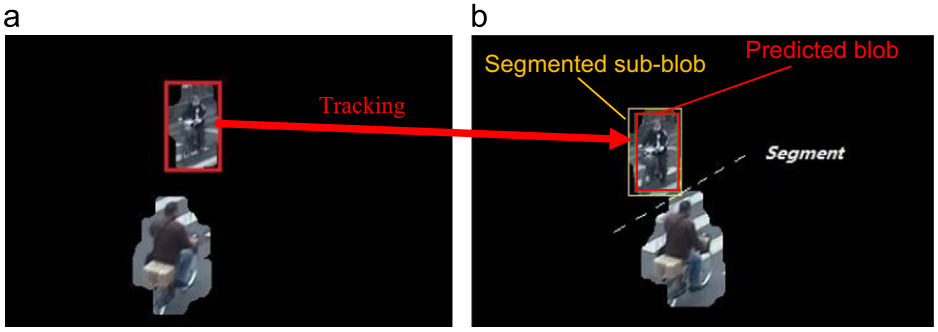


Fig. 9. (a) The blob in the previous non-problematic frame; (b) the segmented sub-blob and the predicted blob in the current frame.

for the entire problematic blob B_t. By Eq. (11), the inter cost can be calculated by the variance of average motion flows from different sub-blobs. Note that we add a negative sign in Eq. (11) such that minimizing C_{inter} can maximize the motion flow differences among different sub-blobs.

$$C_{\text{temp}} = -\frac{1}{N_{t-\varepsilon}}\sum_{s=1}^{N_{t-\varepsilon}} \|\mathbf{H}^{\text{pred}}_{B_{t-\varepsilon,s}} - \arg\min_{\mathbf{H}_{B_{t,k}}} \|\mathbf{H}_{B_{t,k}} - \mathbf{H}^{\text{pred}}_{B_{t-\varepsilon,s}}\|\|^2 \quad (12)$$

where $N_{t-\varepsilon}$ is the total number of blobs in frame $t-\varepsilon$.$\mathbf{H}_{B_{t,k}}$ and $\mathbf{H}^{\text{pred}}_{B_{t-\varepsilon,s}}$ are the feature vectors of the segmented sub-blob $B_{t,k}$ in the current frame and the predicted blob from neighboring frames, respectively. In this paper, we use histogram of gradients (HOG) as the feature vectors and the predicted blob $B_{t-\varepsilon,s}$ is achieved by tracking [21] from the blobs in the nearest non-problematic frame, as in Fig. 9. From Eq. (12), we can see that the temporal cost C_{temp} will be minimized when the segmented sub-blobs matches the blobs in the neighboring frame.

Based on Eqs. (8)–(12), the best segmentation can be achieved when: (1) the motion flows in each sub-blob are similar, (2) the flows from different sub-blobs are different, and (3) the segmented sub-blobs match the ones in the neighboring frames. Furthermore, the optimization problem in Eq. (8) can be solved by different ways. In this paper, we apply meanshift [11] with different cluster numbers over the blob's motion and color fields, respectively, to create multiple candidate segmentation results $\mathbf{\Psi}$. And the optimal segmentation result can then be achieved by selecting the one with the minimum cost in Eq. (8).

Furthermore, note that the weighting factors λ_1, λ_2, and λ_3 in Eq. (8) are basically selected to balance the relative reliabilities of the three segmentation suitability costs C_{intra}, C_{inter}, and C_{temp}. More specifically, a weighting factor should become large if its corresponding cost has higher reliability in measuring the candidate segmentation result $\mathbf{\Psi}$. For example, if the intra cost C_{intra} is more reliable than the inter cost C_{inter} when evaluating whether a candidate segmentation result $\mathbf{\Psi}$ is satisfactory or not, C_{intra} 's corresponding weighting factor λ_1 should be larger than C_{inter} 's corresponding weighting factor λ_2. In the experiments in this paper, we manually set λ_1, λ_2, and λ_3 to be 0.3, 0.3, and 0.4, according to the experimental statistics. However, in practice, we can also utilize the similar method as in Eq. (2) to automatically decide the values of these weighting factors, that is, finding a set of weighting factor values that can minimize the overall segmentation error in the training set.

Step 3: *Achieving optimized blob sequences.* After Step 2, the connected objects and the noisy foregrounds in the problematic blobs can be suitably segmented. Thus, by re-associating the segmented blobs with blobs in neighboring frames, accurate blob sequences can be achieved.

5.3. Abnormal activity classification

After achieving the optimized abnormal blob sequences, the "abnormal activity classification" module will utilize the key regions to classify the abnormal blob sequences, as by Eqs. (4) and (5).

6. Abnormality-type-based video synopsis

Finally, based on the detected and classified abnormal blob sequences, video synopsis is performed to achieve summary videos. Our proposed abnormality-type-based synopsis method can be described by Eq. (13).

$$\mathbf{V}^* = \arg\min_{V=\{\hat{R}\}} \sum_{R,R'\in\mathfrak{R}} \left(\alpha \times E_t\left(\hat{R},\hat{R}'\right) + \beta \times E_c\left(\hat{R},\hat{R}'\right)\right) + \sum_{R\in\mathfrak{R}} \gamma \times E_l\left(\hat{R}\right) \quad (13)$$

where $\Re$ is the set of all abnormal blob sequences. **V** is the synopsis video which shifts the abnormal blob sequences R into new time periods. $\hat{R}$ is the shifted result of R in the synopsis video and R' denotes a different activity (blob sequence) from R. And α, β, and γ are the weighting factors. $E_t\left(\hat{R},\hat{R}'\right)$ is the temporal consistency cost which tends to preserve the temporal relations between activities R and R' in the original video, and $E_c\left(\hat{R},\hat{R}'\right)$ indicates the collision cost to avoid blob sequences R and R' colliding each other. $E_t\left(\hat{R},\hat{R}'\right)$ and $E_c\left(\hat{R},\hat{R}'\right)$ can be calculated by [1]. $E_l\left(\hat{R}\right)$ is our newly introduced term to put together same-type blob sequences in the synopsis video, as in Eq. (14).

$$E_l\left(\hat{R}\right)=\frac{\tau\times A_{\hat{R}}^2}{\hat{t}_{\hat{R}}}+\hat{t}_{\hat{R}} \tag{14}$$

where $A_{\hat{R}}$ is the abnormality type label of blob sequence $\hat{R}$. $\hat{t}_{\hat{R}}$ is the starting time of $\hat{R}$ in the synopsis video. τ controls the time interval among different types of abnormalities. Since $E_l\left(\hat{R}\right)$ is minimized when $\hat{t}_{\hat{R}}=\sqrt{\tau}A_{\hat{R}}$, by minimizing $E_l\left(\hat{R}\right)$, we can guarantee that same-type blob sequences are shifted to the same time while different-type blob sequences will be shifted to different time periods in the synopsis video.

Moreover, the weighting factors α, β, and γ in Eq. (13) are used to balance the relative importance of the three costs. More specifically, a larger weighting factor will increase the importance of its corresponding cost. Thus, the constraint defined by the cost will be considered with higher priority in the synopsis result. Fig. 10 shows some synopsis results under different values of α, β, and γ. Comparing Fig. 10(a) and (b), we can see that when decreasing the weight β for the collision cost, more collisions (i.e., more blob overlaps) will appear in the resulting summary video. Furthermore, comparing Fig. 10(a) and (c), we can also see that when decreasing the weight γ for the type-based cost $E_l\left(\hat{R}\right)$, $E_l\left(\hat{R}\right)$ will have lower impact on the synopsis result and thus blob sequences of the same type tend to be more dispersedly located in the summary video. In our experiments, in order to decide α, β, and γ, we first select one training video clip and create multiple synopsis results under different α, β, and γ values. Then, the values that create the most satisfactory result are selected and these values are used for creating synopsis results for all the testing videos. With the above process, α, β, and γ is set to be 0.1, 0.4, 0.5 in our experiments.

7. Experimental results

In this section, we show experimental results for our approach. Note that the way to decide parameter values in our approach has been described when introducing these parameters in the previous sections. We perform experiments on 15 long surveillance videos whose duration are between 40 and 120 min. Each video is captured from a different scene. Figs. 11 and 13 shows some of the example sequences. In our experiments, for each long video, we use the first 10-min clip for training and use the remaining clip for testing and creating summary videos. Furthermore, in order to decide a suitable patch size for a given scene, we randomly select 20 object blobs from the scene's training video clip, and the patch width/height is then set to be half of these blobs' average width.

Note that these videos are challenging in that: (1) Most videos are crowded, making it difficult to create highly-compressed summary videos; (2) There are frequent object occlusions in the videos, making it difficult to extract reliable blob sequences; (3) The abnormal activities in the videos have large variations, making it difficult to reliably detect and classify abnormalities.

7.1. Performance on abnormality detection and classification

In this experiment, we compare abnormality detection performances of our approach with five methods: (1) The network-

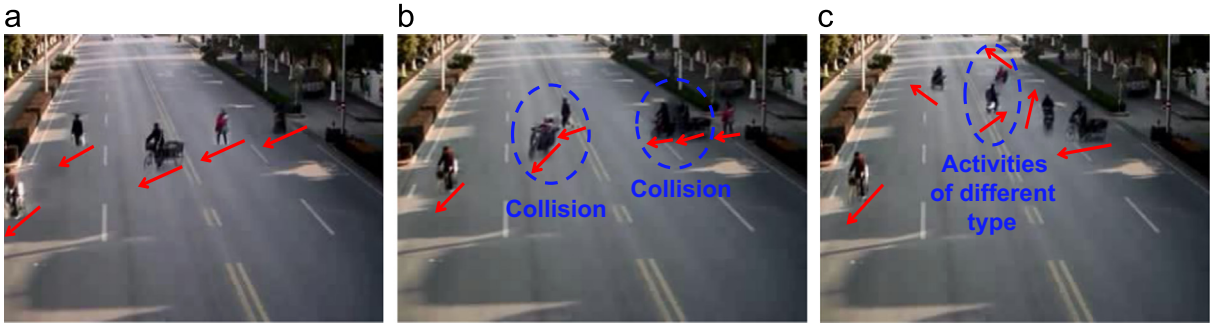


Fig.10. Synopsis results with different α, β, and γ values in Eq. (13). (a) $\alpha=0.1$, $\beta=0.4$, $\gamma=0.5$, (b) $\alpha=0.1$, $\beta=0.05$, $\gamma=0.5$, (c) $\alpha=0.1$, $\beta=0.4$, $\gamma=0.05$.

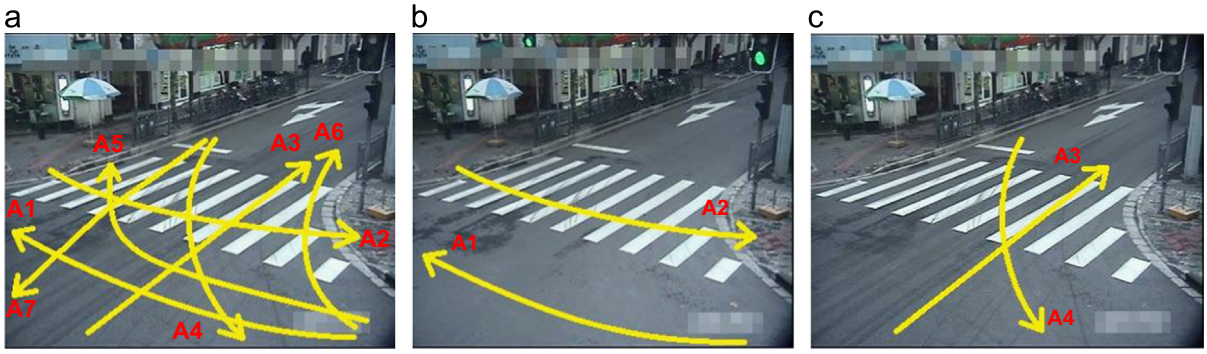


Fig. 11. (a) All activity types in the video; (b) abnormal activities when the green light on the up-right corner is on; (c) abnormal activities when the green light on the up-right corner is off. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

transmission-based method (NTB) [5]; (2) The kernel density estimation method which directly models probability densities for different activities based on training trajectories (KDE) [5,13]; (3) The dynamic time warping method which detect abnormalities by finding their closest time-warped trajectories in the training set [6,19] (DTW); (4) The Gaussian process regression flow method which utilizes Gaussian process flows to model the location and velocity probability for each point in a trajectory [18] (GPRF); (5) Using our candidate abnormality detection result as the final result (i.e., detect abnormalities by Eq. (1) without further applying the key region information in Eq. (5) to classify activity types (Our-without key region).

We first perform experiments on a 40-min long video as in Fig. 11. In Fig. 11(a) shows all activity types in the scene while (b) and (c) show the abnormal activities. Note that since the crossroad in the scene has red/green light switches, different abnormalities are defined when the up-right green light in the scene is on or off.

Table 1 compares the miss detection rates (Miss) for normal and abnormal activities [10] as well as the total normality/abnormality classification error rate (TEF) as defined in Eq. (2). TEF reflects the overall accuracy of a method [5]. Furthermore, Fig. 12 further compares the classification confusion matrixes of three methods for all the normal and abnormal activities in Fig. 11 (a). Note that the results of NTB and GPRF methods are not included in Fig. 12 since they can only detect abnormalities and cannot differentiate different abnormal activity types.

From Table 1 and Fig. 12, we can have the following observations:

(1) Our proposed approach has the best performances in both abnormality detection and activity classification. This demonstrates the effectiveness of our patch-based method.
(2) Comparing our approach with the "our-no key region" method, we can see that by including the key region correlation information, we can not only effectively classify different activity types, but also obviously improve the abnormality detection accuracy by filtering wrongly classified activities (i.e., blob sequences passing through high correlated key regions will be corrected to be normal).
(3) Comparing our approach with the KDE, DTW, and GPRF methods, we can see that: (a) By constructing normal patterns for each patch instead of for the entire trajectory, we can have stronger capabilities to handle the variations of different abnormalities. Thus, the abnormality miss rate can be obviously decreased. (b) By utilizing key regions instead of trajectory similarities to classify activities, the interferences of trajectory uncertainties can be effectively avoided. Thus more accurate classification results can be achieved by our approach.

Moreover, Table 2 further compares the average error rates of different methods over the rest 14 sequences. Some example frames of the sequences and their corresponding abnormal activity types are shown in Fig. 13. Table 2 further demonstrates the effectiveness of our proposed approach.

7.2. Performance on blob sequence extraction

Figs. 14 and 15 shows two blob sequence extraction results by six methods: TLD tracking (TLD) [8], CT tracking (CT) [9], K-shortest paths optimization (KSP) [16], kernelized correlation filter tracking (KCF) [21], simply applying blob association in Section 5.1 without using the blob optimization process (blob associate) [10], our approach which introduces blob optimization to achieve blob sequences (our). Moreover, Table 3 further compares the success rates and the center location errors [9] over all detected abnormal blob sequences in our dataset for the above six methods. The success rate is defined by the number of success tracking bounding boxes divided by the total number of tracking bounding boxes. A tracking bounding box BB_T is a success bounding box if $\text{size}(BB_T \cap BB_G)/\text{size}(BB_T \cup BB_G) > 0.5$, where BB_G is the groundtruth bounding box. The center location error is the pixel distance between the centers of a tracking bounding box and the corresponding groundtruth bounding box [9]. All the frames are normalized to 352×288 when measuring this center location error.

From Table 3 and Figs. 14 and 15, we can have the following observations:

(1) It is clear that blob sequences achieved by our approach are more accurate than the other methods. For example, in the

Table 1
The Miss and TER rates for different abnormality detection methods on the sequence of Fig. 11.

	NTB [5]	KDE [13]	DTW [19]	GPRF [18]	Our-no key region	**Our**
Normal miss (%)	10.7	15.7	11.3	10.2	12.3	**9.0**
Abnormal miss (%)	20.5	19.0	18.7	18.1	11.3	**11.3**
TER (%)	15.0	16.5	13.5	12.7	11.8	**9.7**

Table 2
The average Miss and TER rates for different methods over the other 14 long sequences in our dataset.

	NTB [5]	KDE [13]	DTW [19]	GPRF [18]	Our-no key region	**Our**
Normal miss (%)	28.9	39.5	45.7	27.7	25.6	**10.7**
Abnormal miss (%)	41.6	35.2	37.9	29.3	14.1	**14.1**
TER (%)	33.0	38.3	43.3	28.2	22.2	**12.8**

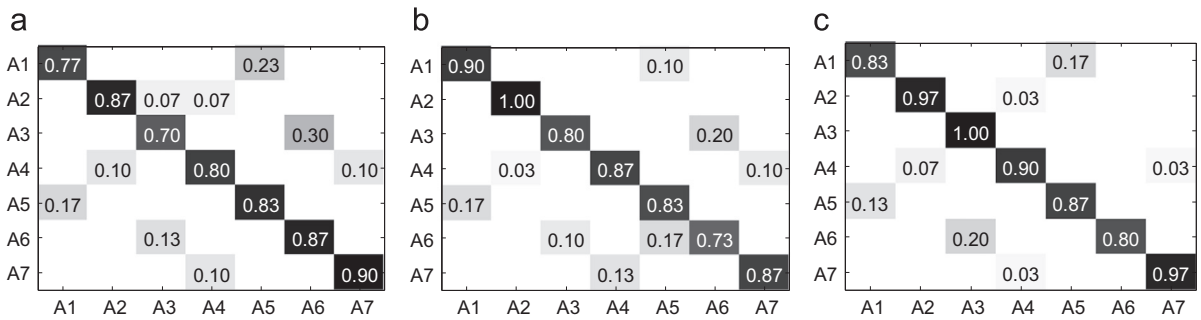


Fig. 12. Activity classification confusion matrixes for the activities in Fig. 11(a). (a) KDE, (b) DTW, (c) Our.

Fig. 13. The example sequences as well as their corresponding abnormal activity types (best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

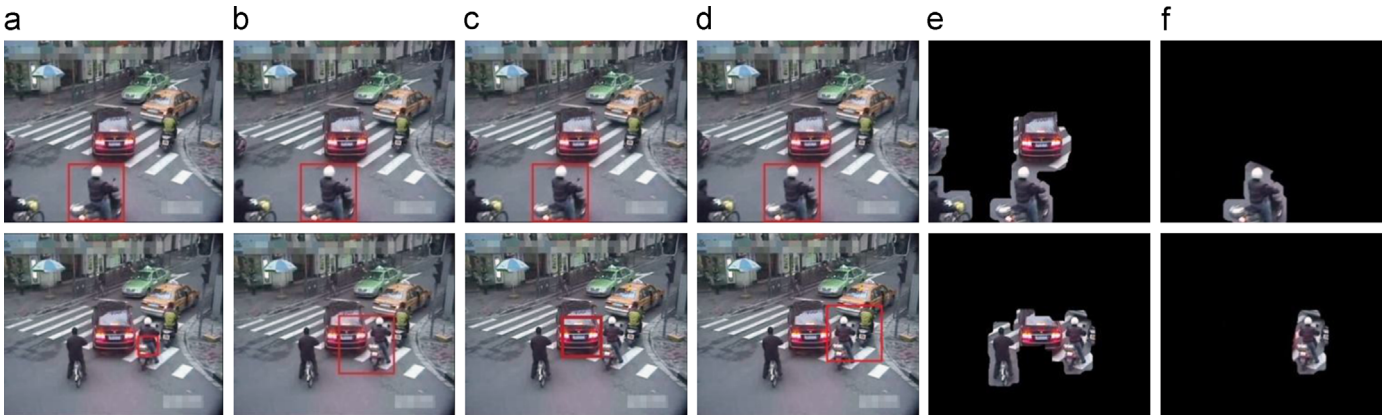


Fig. 14. Blob sequence extraction results for different methods. (a) TLD, (b) CT, (c) KSP, (d) KCF, (e) blob associate, (f) our approach.

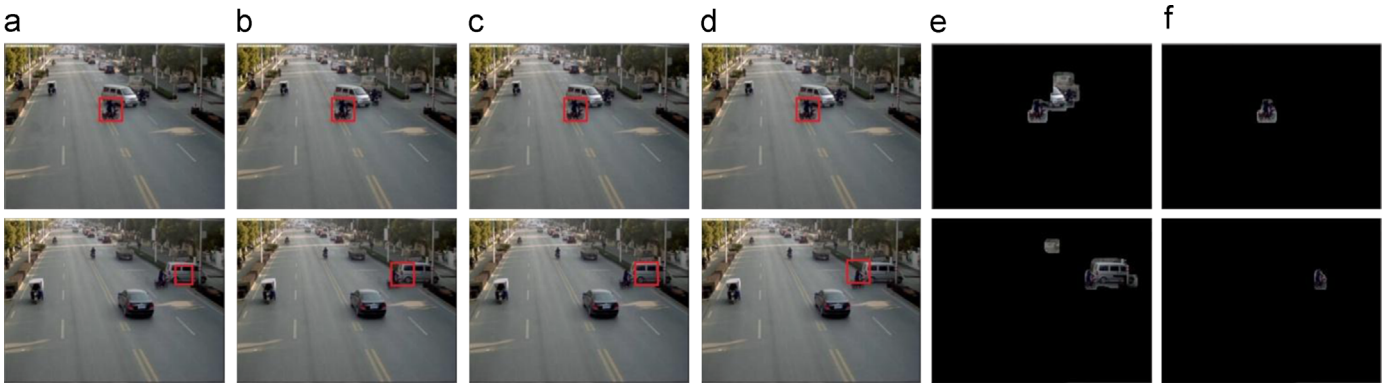


Fig. 15. Blob sequence extraction results for different methods for another sequence. (a) TLD, (b) CT, (c) KSP, (d) KCF, (e) blob associate, (f) our approach.

Table 3
Blob sequence extraction accuracy of different methods over 15 long sequences in our dataset.

Methods	Particle [7]	TLD [8]	CT [9]	KSP [16]	KCF [21]	Blob Associate [10]	Our
Success rate (%)	49.8	64.1	73.6	58.7	79.2	62.3	86.7
Center location error	91	35	19	24	10	27	6

bottom line of Fig. 14, when object occlusion takes place, the tracking methods such as TLD almost miss the object blob. Although the CT, KCF, and "blob associate" methods can include the object, their extracted blobs are far from satisfactory which include large regions of irrelevant objects. Furthermore, the KSP method which performs multiple-object tracking also cannot achieve satisfying results due to the interferences of occlusion and object detection errors in crowded scenes. Comparatively, by using our blob optimization process, more precise blob sequences can be achieved by integrating the spatial-temporal correlation among blob sequences for handling occlusion, as in Fig. 14(f).

(2) More importantly, it should be noted that most tracking-based methods (as Fig. 14(a)–(d)) only focus on achieving object bounding boxes while the suitable segmentation of object blobs is not addressed. Even if their bounding boxes suitably locate the object, the foreground blobs inside bounding boxes may still include unwanted regions which may obviously affect the qualities of summary videos. Comparatively, our blob optimization process is designed to extract accurate object blobs. And this is another important advantage of our approach.

7.3. Performance on summary video creation

Figs. 17–23 show the created summary videos. In Figs. 17 and 19, the summary videos by three methods for the input videos in Figs. 16 and 18 are compared: using the synopsis method by Pritch [1] on all blob sequences in a video (Pritch+All), only using the

Fig. 16. Example frames for an input video.

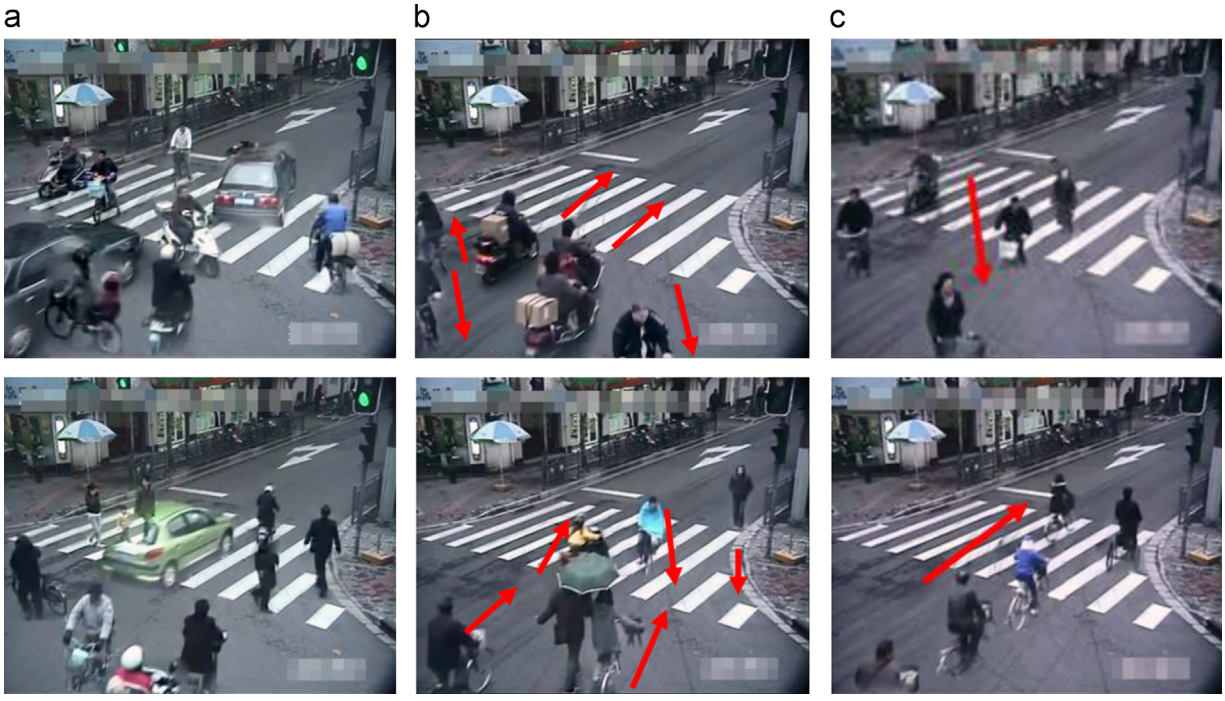


Fig. 17. Summary videos for the video of Fig. 16 created by different methods (best viewed in color). (a) Pritch+all, (b) Pritch+Abnormal, (c) Our. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

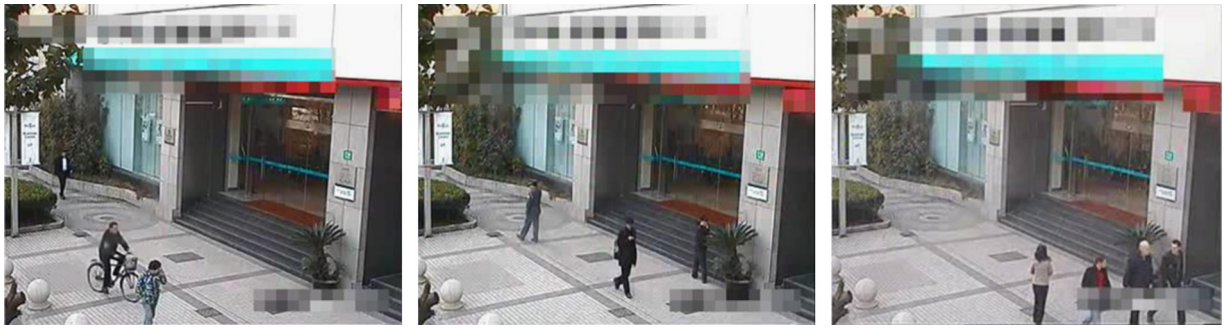


Fig. 18. Example frames for an input video.

synopsis method by Pritch [1] on the extracted abnormal blob sequences (Pritch+Abnormal), our approach which introduces an activity-type cost (Eq. (14)) to properly organize abnormalities in the summary video (Our). Besides, Figs. 22 and 23 show the additional summary videos created by our approach for various input surveillance videos[3]. Moreover, in order to further demonstrate the effectiveness of our proposed synopsis approach, we further perform experiments on a public MCT surveillance dataset [39] and compare with the other methods. The resulting summary videos are shown in Fig. 21.

From Figs. 17, 19 and 21, we can see that the summary video by the "Pritch+all" method is too crowded due to the inclusion of all activity blobs in a short period. Although the "Pritch+abnormal" method obviously decreases the crowdedness, object activities in the summary video are still less suitably organized with large numbers of disordered motions. Comparatively, the summary

[3] An example summary video created by our approach is available at https://www.dropbox.com/s/iea9lgfrxd8mt9x/Supplementary.zip?dl=0.

Fig. 19. Summary videos for the video of Fig. 18 created by different methods (best viewed in color). (a) Pritch+all, (b) Pritch+Abnormal, (c) Our. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 20. Example frames for videos in the public MCT dataset.

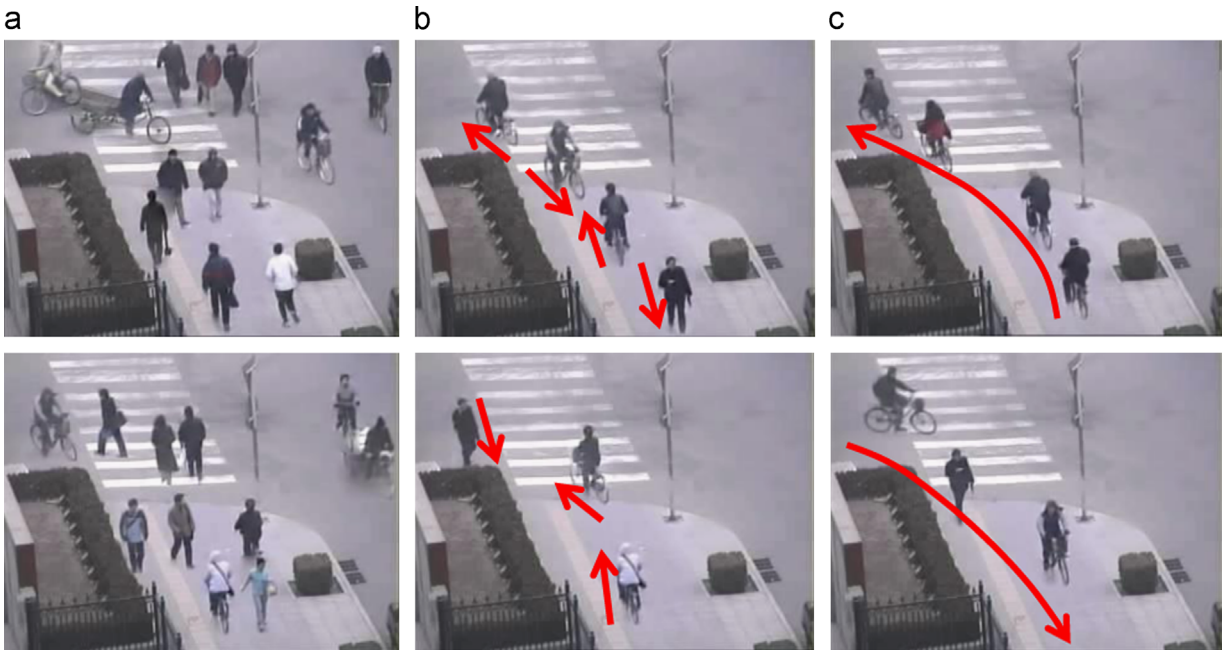


Fig. 21. Summary videos for the video of Fig. 20 created by different methods (best viewed in color). (a) Pritch+all, (b) Pritch+Abnormal, (c) Our. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

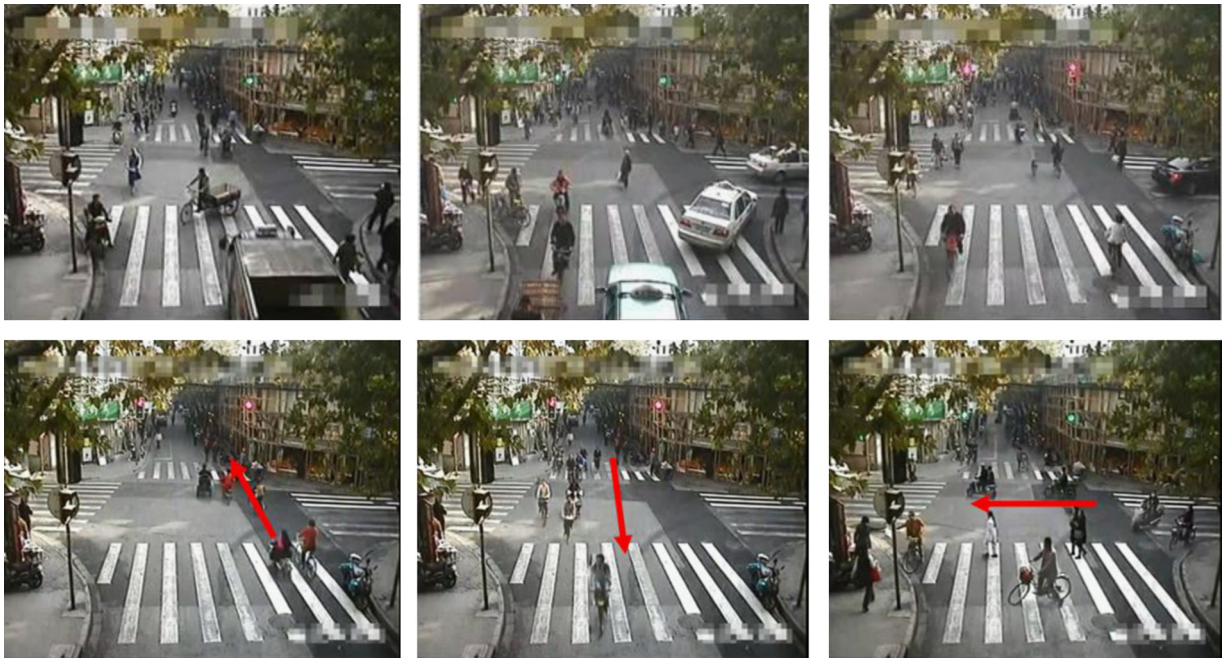


Fig. 22. Up: example frames for an input video. Down: summary video created by our approach. (best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

video created by our approach is more appealing where abnormalities of different classes are properly organized together in the summary video. Thus, people can find their interested abnormalities more conveniently. Furthermore, the results in Figs. 21 and 22 further demonstrate that our approach is able to create more appealing and better-organized summary videos.

8. Conclusion

In this paper, a new approach is proposed to detect abnormal activities in surveillance videos and create suitable summary videos accordingly. The proposed approach extracts key regions for improving abnormality detection accuracy, utilizes a blob optimization process to achieve suitable blob sequences, and introduces an activity-type cost to suitably organize abnormalities. Experimental results demonstrate the effectiveness of the proposed approach.

Future work will include: (1) combining with trajectory association methods [12] to further improve the blob sequence extraction accuracy; (2) combining with automatic ROI detection methods [29,30] to automatically decide ROIs in a scene; (3) extending the activity detection module from abnormal activity detection to the detection of other activity types, such that the approach is capable of creating summary videos for arbitrary activities of interest.

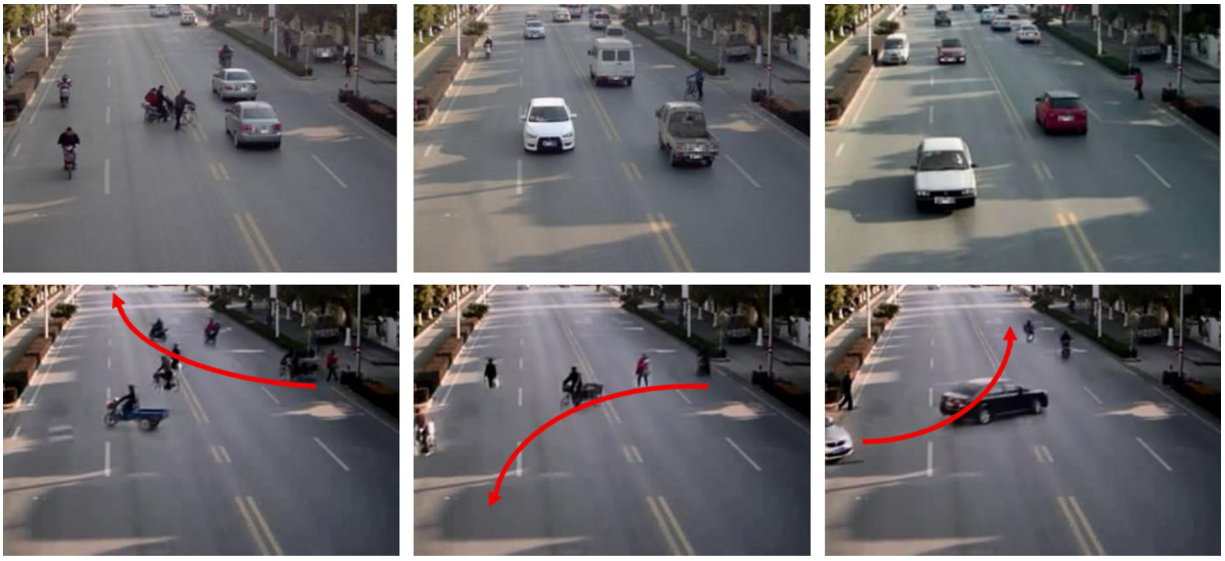


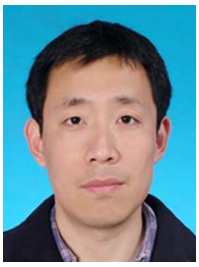
Fig. 23. Up: example frames for an input video. Down: summary video created by our approach. (best viewed in color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Acknowledgements

This paper is supported in part by the following grants: National Science Foundation of China (nos. 61471235, 61001146, 61303170, and 61379079), and Chinese National 973 Grants (2013CB329603).

References

[1] Y. Pritch, A. Rav-Acha, S. Peleg, Nonchronological video synopsis and indexing, IEEE Trans. Pattern Anal. Mach. Intell. 30 (11) (2008) 1971–1984.
[2] Y. Nie, C. Xiao, H. Sun, Compact video synopsis via global spatiotemporal optimization, IEEE Trans. Vis. Comput. Graph. 19 (10) (2013) 1664–1676.
[3] E.E. Zelniker, S. Gong, T. Xiang, Global abnormal behavior detection using a network of CCTV cameras, Int. Workshop. Visual Surveillance 1 (2008) 1–8.
[4] Z. Zhang, Flexible camera calibration by viewing a plane from unknown orientations, Int. Conf. Comput. Vis. (ICCV) 1 (1999) 666–673.
[5] W. Lin, Y. Chen, J. Wu, H. Wang, B. Sheng, H. Li, A new network-based algorithm for human activity recognition in videos, IEEE Trans. Circuits Syst. Video Technol. 24 (5) (2014) 826–841.
[6] C. Rao, M. Shah, T. Syeda-Mahmood, Action recognition based on view invariant spatio-temporal analysis, ACM Multimed 1 (2003) 518–527.
[7] R. Hess, A. Fern, Discriminatively trained particle filters for complex multi-object tracking, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2009) 240–247.
[8] Z. Kalal, K. Mikolajczyk, J. Matas, Tracking-Learning-Detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (7) (2012) 1409–1422.
[9] K. Zhang, L. Zhang, M. Yang, Real-time compressive tracking, Eur. Conf. Comput. Vis. (ECCV) 1 (2012) 864–877.
[10] X. Su, W. Lin, X. Zheng, X. Han, H. Chu, X. Zhang, A new local-main-gradient-orientation HOG and contour differences based algorithm for object classification, Int. Symp. Circuits Syst. 1 (2013) 2892–2895.
[11] D. Comaniciu, P. Meer, Mean shift: A robust approach toward feature space analysis, IEEE Trans. Pattern Anal. Mach. Intell. 24 (5) (2002) 603–619.
[12] B. Benfold, I. Reid, Stable multi-target tracking in real-time surveillance video, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2011) 3457–3464.
[13] Kernel Density Estimation Toolbox, ⟨http://www.ics.uci.edu/~ihler/code/⟩.
[14] C. Kim, J. Hwang, An integrated scheme for object-based video abstraction, ACM Multimed. (MM) 1 (2000) 303–311.
[15] R. Chaudhry, A. Ravichandran, Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2009) 1932–1939.
[16] J. Berclaz, F. Fleuret, E. Türetken, P. Fua, Multiple object tracking using K-shortest paths optimization, IEEE Trans. Pattern Anal. Mach. Intell. 33 (9) (2011) 1806–1819.
[17] J. Kim, K. Grauman, Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2009) 2921–2928.
[18] K. Kim, D. Lee, I. Essa, Gaussian process regression flow for analysis of motion trajectories, Intl. Conf. Comput. Vis. (ICCV) 1 (2011) 1164–1171.
[19] C. Rao, A. Yilmaz, M. Shah, View-invariant representation and recognition of actions, Int. J. Comput. Vis. 50 (2) (2002) 203–226.
[20] S. Mei, G. Guan, Z Wang, M. He1, X. Hua, D. Feng, L20 constrained sparse dictionary selection for video summarization, Int. Conf. Multimed. Expo (ICME) 1 (2014) 1–6.
[21] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, High-speed tracking with kernelized correlation filters, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 1–14.
[22] Y. Cong, J. Yuan, J. Liu, Sparse reconstruction cost for abnormal event detection, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2011) 3449–3456.
[23] C. Ngo, Y. Ma, H. Zhang, Video summarization and scene detection by graph modeling, IEEE Trans. Circuits Syst. Video Technol. 15 (2) (2005) 296–305.
[24] A. Rav-Acha, Y. Pritch, S. Peleg, Making a long video short: Dynamic video synopsis, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2006) 435–441.
[25] B. Truong, S. Venkatesh, Video abstraction: a systematic review and classification, ACM Trans. Multimed. Comput. Commun. Appl. 3 (1) (2007) 1–37.
[26] Y. Lee, J. Ghosh, K. Grauman, Discovering important people and objects for egocentric video summarization, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2012) 1346–1353.
[27] B. Morris, M. Trivedi, Trajectory learning for activity understanding: unsupervised, multilevel, and long-term adaptive approach, IEEE Trans. Pattern Anal. Mach. Intell. 33 (11) (2011) 2287–2301.
[28] Y. Zhou, W. Lin, H. Su, J. Wu, J. Wang, Y. Zhou, Representing and recognizing motion trajectories: a tube and droplet approach, ACM Multimed. (MM) 1 (2014) 1077–1080.
[29] P. Kapsalas, K. Rapantzikos, A. Sofou, Y. Avrithis, Regions of interest for accurate object detection. in: Proceedings of the International Workshop on Content-Based Multimedia Indexing, vol. 1, pp. 147–154, 2008.
[30] M. Rokunuzzaman, K. Sekiyama, T. Fukuda, Automatic ROI detection and evaluation in video sequences based on human interest, J. Robot. Mechatron. 22 (1) (2010) 65–75.
[31] M. Swain, D. Ballard, Color indexing, Int. J. Comput. Vis. 7 (1) (1991) 11–32.
[32] X. Wang, K.T. Ma, G. Ng, E. Grimson, Trajectory analysis and semantic region modeling using nonparametric Bayesian Models, Int. J. Comput. Vis. 96 (2011) 287–321.
[33] J. Nascimento, M. Figueiredo, J. Marques, Trajectory classification using switched dynamical hidden Markov models, IEEE Trans. Image Process. 19 (5) (2010) 1338–1348.
[34] R. Mehran, A. Oyama, M. Shah, Abnormal crowd behavior detection using social force model, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2009) 935–942.
[35] X. Cui, Q. Liu, M. Gao, D. Metaxas, N, Abnormal detection using interaction energy potentials, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2011) 3161–3167.
[36] J.K. Aggarwal, M.S. Ryoo, Human activity analysis: a review, ACM Comput. Surv. (CSUR) 43 (16) (2011) 1–47.
[37] D.S. Bolme, J.R. Beveridge, B.A. Draper, Y. Lui, Visual object tracking using adaptive correlation filters, IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 1 (2010) 2544–2550.
[38] T. Mei, Y. Rui, S. Li, Q. Tian, Multimedia search reranking: a literature survey, ACM Comput. Surv. (CSUR) 46 (3) (2014) 1–36.
[39] M.C.T. Dataset, ⟨http://mct.idealtest.org/Datasets.html⟩.



Weiyao Lin received the B.E. degree from Shanghai Jiao Tong University, China, in 2003, the M.E. degree from Shanghai Jiao Tong University, China, in 2005, and the Ph.D. degree from the University of Washington, Seattle, USA, in 2010, all in electrical engineering. Currently, he is an associate professor at the Department of Electronic Engineering, Shanghai Jiao Tong University.

His research interests include video processing, machine learning, computer vision and video coding & compression.

Yihao Zhang received the received the B.S. degree from Shanghai Jiao Tong University, Shanghai, China, in 2012 in electrical engineering. He is currently working toward the M.E. degree in electrical engineering from Shanghai Jiao Tong University, China.

His research interests include image & video processing, machine learning, computer vision and multimedia technologies.

Jiwen Lu received the BEng degree in mechanical engineering and the MEng degree in electrical engineering, both from the Xi'an University of Technology, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from the Nanyang Technological University, Singapore, in 2011. He is currently a research scientist at the Advanced Digital Sciences Center, Singapore.

His research interests include computer vision, pattern recognition, machine learning, and biometrics. He has authored/coauthored more than 70 scientific papers in peer-reviewed journals and conferences including some top venues such as the IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Circuits and Systems for Video Technology, IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, IEEE Transactions on Information Forensics and Security, ICCV, CVPR, and ACM MM. He received the First-Prize National Scholarship and the National Outstanding Student awarded by the Ministry of Education of China, in 2002 and 2003, and the Best Student Paper Award from PREMIA of Singapore in 2012, respectively. He is a member of the IEEE.

Bing Zhou received the B.S. and M.S. degrees from Xi'an Jiao Tong University in 1986 and 1989, respectively, and the Ph.D. degree in Beihang University in 2003, all in computer science.

He is currently a professor at the School of Information Engineering, Zhengzhou University, Henan, China. His research interests cover video processing and understanding, surveillance, computer vision, multimedia applications.

Jinjun Wang received the B.E. and M.E. degree from Huazhong University of Science and Technology, Wuhan, China, in 2000 and 2003. He received the Ph.D. degree from Nanyang Technological University, Singapore, in 2008. Currently, he is a professor at Institute of Artificial Intelligence and Robotics, Xi'an Jiao Tong University, China.

His research interests include contentbased sports video analysis and retrieval, semantic event detection, pattern classification, and automatic video editing.

Yu Zhou holds a Ph.D. degree in computer science from Harbin Institute of Technology, China. He is a full assistantprofessor in the Institute of Information Engineering, Chinese Academy of Sciences (CAS), China. His current research interests include security problems in multimedia. Before joining CAS, he was a postdoc research fellow in Shanghai Jiao Tong University from 2010 to 2012. He served as a member of the technical program committee of the 2nd International Workshop on Emerging Multimedia Systems and Applications (in conjunction with ICME 2013), and as reviewers of Journal of Systems Science & Complexity, the 2nd International Workshop on Emerging Multimedia Systems and Applications, and the 19th Asia Pacific Conference on Communications. He has published over ten technical articles in refereed journals and conference proceedings in the areas of multimedia and pattern recognition.