Journal of
Electronic Imaging

JElectronicImaging.org

Improving video foreground segmentation with an object-like pool

Xiaoliu Cheng
Wei Lv
Huawei Liu
Xing You
Baoqing Li
Xiaobing Yuan

SPIE.
IS&T
imaging.org

Improving video foreground segmentation with an object-like pool

Xiaoliu Cheng,[a,b] Wei Lv,[a] Huawei Liu,[a,b] Xing You,[a,*] Baoqing Li,[a] and Xiaobing Yuan[a]
[a]Chinese Academy of Sciences, Shanghai Institute of Microsystem and Information Technology, Wireless Sensor Network Laboratory, No. 865 Changning Road, Changning District, Shanghai 200050, China
[b]University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100049, China

Abstract. Foreground segmentation in video frames is quite valuable for object and activity recognition, while the existing approaches often demand training data or initial annotation, which is expensive and inconvenient. We propose an automatic and unsupervised method of foreground segmentation given an unlabeled and short video. The pixel-level optical flow and binary mask features are converted into the normal probabilistic superpixels, therefore, they are adaptable to build the superpixel-level conditional random field which aims to label the foreground and background. We exploit the fact that the appearance and motion features of the moving object are temporally and spatially coherent in general, to construct an object-like pool and background-like pool via the previous segmented results. The continuously updated pools can be regarded as the "prior" knowledge of the current frame to provide a reliable way to learn the features of the object. Experimental results demonstrate that our approach exceeds the current methods, both qualitatively and quantitatively. [DOI: 10.1117/1.JEI.24.2.023034]

Keywords: video foreground segmentation; object-like pool; unsupervised; unlabeled; conditional random field; probabilistic superpixels.

Paper 15016 received Jan. 9, 2015; accepted for publication Mar. 27, 2015; published online Apr. 23, 2015.

1 Introduction

Video foreground segmentation plays a prerequisite role in a variety of visual applications such as safety surveillance[1] and intelligent transportation.[2] The existing algorithms usually use supervised or semisupervised methods and achieve satisfying results. However, the performances are still limited when they are applied for unsupervised and short videos, because the supervised methods usually demand many training examples that are expensive to manually label. Furthermore, the training examples cannot cover all the conditions and need to retrain the new examples to improve the generalization. Some semisupervised methods require accurate object region annotation only for the first frame, then they exploit the region-tracking methods to segment the rest of the frames. However, many visual applications like safety surveillance demand intelligent and unattended operations, which make the initial annotation impractical. The available video frames may be insufficient sometimes since the objects can move rapidly into and out of the visual field when they are near the camera.

There has been a substantial amount of work related to foreground segmentation. Classical segmentation methods that operate at the pixel level are often based on local features like textons,[3] then they are augmented by Markov random field or graph-cut based methods to gain the refined results.[4,5] Furthermore, some new methods of this type regard the meaningful superpixels as the basic units instead of the rigid pixels to get better results,[6–10] because superpixels are efficient in practice and more robust to noise than pixels, and work well for representing objects as well. For instance, Tian et al.[6] propose two superpixel-based data terms and smooth terms defined on the spatiotemporal superpixel neighborhood with a shape cue to implement the segmentation. Their method can handle arbitrary length video sequences although it demands that the first frame be manually labeled. Shu et al.[9] apply a superpixel-based bag-of-words model to iteratively refine the output of a generic detector, then an online-learning appearance model is exploited to train a support vector machine and to achieve the exact objects using conditional random field (CRF). However, it requires a mass of various examples to train the classifier, and it is not well adapted to short videos.

Perhaps the work that is related most to ours is that of Schick et al.[8] They convert the traditional pixel-based segmentation into a probabilistic superpixel representation and integrate the structure information and similarities into Markov random field (MRF) to improve the segmentation. The shape of the object in the given foreground segmentation is improved by their probabilistic superpixel Markov random field (PSP-MRF) method. Moreover, it also reduces the noisy regions and improves recall, precision, and F-measure. However, it stringently depends on the binary mask (see Sec. 3.3). For instance, if the given binary mask is quite poor because of the cluttered background, the performance will rapidly decline. In addition, full use is not made of the local features and environmental information to achieve more robust results.

In order to improve the performance of unsupervised and short video segmentation, we proposed an online unsupervised learning approach inspired by Ref. 9. The intuition is that the appearance and motion features of the moving object vary slowly frame by frame in a typical video. According to the temporal and spatial coherence, we can

*Address all correspondence to: Xing You, E-mail: simit52@163.com

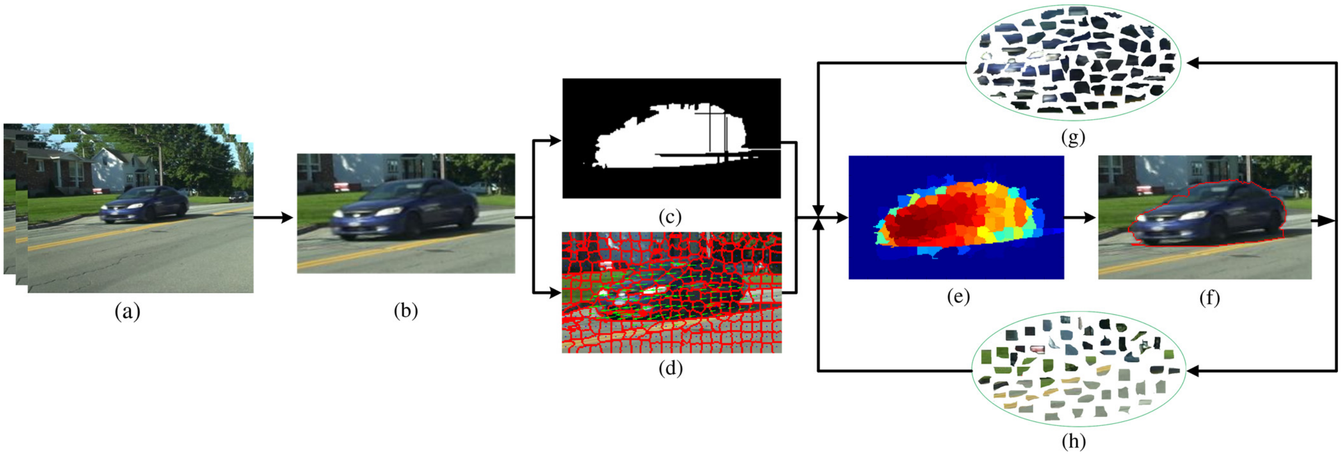

Fig. 1 The overview of our approach: (a) input sequential frames, (b) moving region, (c) binary mask, (d) superpixel-level optical flow, (e) foreground likelihood, (f) segmented results, (g) object-like pool, and (h) background-like pool.

exploit the segmented result of the previous frame to provide valuable cues for the current segmentation.

This paper aims to segment the moving foreground from the unlabeled and short video in an unsupervised way without prior knowledge. The overview of our approach is illustrated in Fig. 1. The main contributions of our work are listed as follows: (1) The pixel-level optical flow and binary mask features are converted into the normalized probabilistic superpixels, which fit very well for the CRF. (2) Because of the temporal and spatial coherence of appearance and motion features of the moving object, we leverage the previous segmented result to build an object-like pool and background-like pool, which serve as the "prior" knowledge of the current segmentation. The continuously updated pools provide a reliable and continuous way to learn the features of the object. The proposed algorithm has been validated by several challenging videos from the change detection 2014 dataset, and experimental results demonstrate that our approach outperforms the other methods both in accuracy and robustness, even when the basic features suffer from great interference.

The rest of this paper is organized as follows: Sec. 2 presents our detailed approach. Experimental results are given in Sec. 3 and conclusions are discussed in Sec. 4.

2 Our Approach

Since we have no prior knowledge about the unlabeled video, we actually know nothing about the object at first: we do not know its type, size, moving direction, and so on. Similarly, the scenario is also unpredictable: it may suffer from swaying trees, illumination change, bad weather, shadows, and so on. Therefore, an unsupervised and efficient approach should be developed because of the limited information in the short video.

First, the optical flow field is regarded as the initial detector to extract the moving region, which is actually a coarse bounding box. Second, the pixel-level optical flow and binary mask features are converted into the normalized probabilistic superpixels. Combining the normalized probabilistic superpixels with the foreground likelihood that is generated by the object-like pool and background-like pool, we build a superpixel-based CRF model to provide a natural way to learn the conditional distribution over the class labeling. Afterward, the graph-cut based method is adopted to achieve the foreground segmentation. Last, an exceptional handling mechanism is applied to avoid error accumulation in the case of abnormal events.

2.1 *Superpixel Segmentation*

Superpixels[11,12] have become a significant tool in computer vision. They group pixels into meaningful subregions instead of rigid pixels which can greatly reduce the complexity of the task in image processing. What is more, the superpixels have uniform information in color and space and adhere well to the contour of the object. So far they have become the basic blocks of many computer vision algorithms, such as object segmentation,[9] depth estimation,[13] and object tracking.[14] As a kind of middle-level feature, superpixels both increase the speed and improve the quality of the segmented results.

Simple linear iterative clustering (SLIC)[15] is an efficient method of superpixel segmentation, which is also simple to implement and easy to apply in practice. In this paper, we set a proper size of superpixels (8×8 in all the experiments) and segment the image with the SLIC algorithm. Then we acquire the table of the labeled superpixels, the seeds of the superpixels, and the number of the superpixels. Specifically, the table shows the label values of all the pixels and the maximum value represents the total number of the final superpixels. Note that the exact number of the segmented superpixels is usually not equal to the given number because some small superpixels are integrated into the larger ones. The seeds of the superpixels are used to judge the neighbor information since the labeled values of superpixels are not in order.

2.2 *Probabilistic Superpixels*

The pixel-level processing is vulnerable to unpredictable noise and it suffers from a heavy calculation burden as well. In order to achieve a robust and efficient segmentation, we operate at the superpixel level in the following steps. According to Ref. 8, a probabilistic superpixel gives the probability that its pixels belong to a certain class, so it fits well into the probabilistic frameworks like CRF, as we will show later.

Though without prior knowledge, the pixel-level optical flow and binary mask can be converted into probabilistic superpixels to measure the foreground likelihood. Let B be the pixel-level binary mask and sp a superpixel with pixels $p \in \mathrm{sp}$ and $|\mathrm{sp}|$ its size, so the likelihood of the superpixel-based binary mask to construct the object is defined as[8]

$$L_{\mathrm{binary}}(\mathrm{sp}) = \frac{\sum_{p\in \mathrm{sp}} B(p)}{|\mathrm{sp}|}. \tag{1}$$

The optical flow of each superpixel is represented by the average optical flow of the inside pixels. Then the likelihood of a superpixel sp (let $\vec{\mathrm{sp}}$ be its optical flow vector) to form the foreground based on optical flow is defined as

$$L_{\mathrm{flow}}(\vec{\mathrm{sp}}) = \cos\langle \vec{\mathrm{sp}}, \vec{r}\rangle \cdot \frac{\|\vec{\mathrm{sp}}\|}{\|\vec{r}\|}, \tag{2}$$

where $\langle \vec{\mathrm{sp}}, \vec{r}\rangle$ denotes the angle between the vectors $\vec{\mathrm{sp}}$ and $\vec{r}$. The reference optical flow vector $\vec{r}$ is defined by the mean optical flow of all the superpixels in the moving region. Finally, the superpixel-level optical flow and binary mask are normalized to represent the foreground and background probabilities by the following equations:

$$P_{\mathrm{fg}} = \alpha \cdot L_{\mathrm{flow}} + (1-\alpha)\cdot L_{\mathrm{binary}}, \tag{3}$$

$$P_{\mathrm{bg}} = 1 - P_{\mathrm{fg}}, \tag{4}$$

where $\alpha \in (0{,}1)$ represents the tradeoff between the features of the binary mask and the optical flow.

2.3 *Superpixel-Based Conditional Random Field*

CRF[16] is a class of statistical modeling methods widely applied to computer vision. According to the result of superpixel segmentation, the foreground objects are usually over-segmented and are consisted of more than one superpixel. Therefore, it is essential to cluster and label the superpixels based on their features. Fortunately, CRF provides a natural way to incorporate superpixel-based features into a single unified model[3] to learn the conditional distribution over the class labeling.

Let $G(S, E)$ be the adjacent graph of superpixels sp_i ($\mathrm{sp}_i \in S$) in a frame, and E is the set of edges formed between pairs of adjacent superpixels in the eight-connected neighbors. Let $P(c|G;\, w)$ be the conditional probability[10] of the set of class assignments c given the adjacent graph $G(S, E)$ and a weight w

$$-\log[P(c|G;w)] = \sum_{\mathrm{sp}_i\in S} \Psi(c_i|\mathrm{sp}_i) + w \sum_{(\mathrm{sp}_i,\mathrm{sp}_j)\in E} \Phi(c_i, c_j|\mathrm{sp}_i, \mathrm{sp}_j), \tag{5}$$

where $\Psi(\cdot)$ and $\Phi(\cdot)$ represent the unary potential and pairwise edge potential, respectively.

The unary potential $\Psi(\cdot)$ defines the cost of labeling superpixel sp_i with label c_i, and it is represented as follows:

$$\Psi(c_i|\mathrm{sp}_i) = -\log[P_{\mathrm{fg}}(c_i, \mathrm{sp}_i)]. \tag{6}$$

The relationship between two adjacent superpixels sp_i and sp_j is modeled by the pairwise potential[4] $\Phi(\cdot)$

$$\Phi(c_i, c_j|\mathrm{sp}_i, \mathrm{sp}_j) = [c_i \neq c_j]\exp(-\beta\|c_i - c_j\|^2), \tag{7}$$

$$\beta = (2\langle \|c_i - c_j\|^2\rangle)^{-1} |(\mathrm{sp}_i, \mathrm{sp}_j) \in E, \tag{8}$$

where $[\cdot]$ denotes the indicator function with values 0 or 1, $\|c_i - c_j\|^2$ is the L_2 norm of the color difference between two adjacent nodes in LAB color space, and $\langle\cdot\rangle$ is the expectation operator.

The conditional probability can be optimized by graph cuts.[17] Once the CRF model has been built, we minimize Eq. (5) with the multilabel graph-cuts[18–20] based on an optimization library[10] using the swap algorithm. This is quite efficient since the CRF model is defined on the superpixel-level graph.

2.4 *Pools Construction*

Now the superpixels are classified into two clusters: foreground and background. In order to learn the features of the object from the segmented result, the superpixels belonging to the foreground and background are separately selected to construct the object-like pool o^{t-1} and the background-like pool bg^{t-1}

$$o^{t-1} = \{\mathrm{sp}_i\},\, \mathrm{sp}_i \in \text{foreground}, \tag{9}$$

$$\mathrm{bg}^{t-1} = \{\mathrm{sp}_j\},\, \mathrm{sp}_j \in \text{background}, \tag{10}$$

where o^{t-1} and bg^{t-1} are the independent object-like pool and background-like pool that are generated from the segmented result of the $(t-1)$'th frame. The color distribution and optical flow of each superpixel within the pools have already been recorded. Based on the temporal and spatial coherence of appearance and motion features, the real object in the next frame should be similar to the previous segmented foreground for both color and optical flow. Therefore, the two pools can be regarded as the "prior" knowledge for the object in the next frame. By comparing the features of the "new" superpixels in the current frame and the "old" superpixels in the two pools, we assign each "new" superpixel a likelihood of its belonging to the foreground.

2.5 *Foreground Likelihood*

Based on the segmented result of the previous frame, the object-like pool o^{t-1} formed by the $(t-1)$'th frame is achieved. As discussed above, o^{t-1} can be regarded as the "prior" knowledge of current frame t, hence the key features about the object can be learned. Let sp_i^t be the i'th superpixel in frame t and sp_k^{t-1} ($\mathrm{sp}_k^{t-1} \in o^{t-1}$) be one of the nearest M_k neighbors of sp_i^t. The similarity to the object about sp_i^t is denoted as

$$S_o^t(\mathrm{sp}_i^t) = \frac{1}{M_k} \sum_{\mathrm{sp}_k^{t-1}\in N(sp_i^t)} H(\mathrm{sp}_k^{t-1}) \cdot H(\mathrm{sp}_i^t)^T \cdot \exp\left[-\frac{D(\vec{\mathrm{sp}}_k^{t-1}, \vec{\mathrm{sp}}_i^t)}{\eta}\right], \tag{11}$$

where $H(\cdot)$ and $D(\cdot)$ are the histogram distribution and the Euclidean distance between optical flow vectors, respectively.

The optical flow vector of sp_i^t is denoted as $\vec{\mathrm{sp}}_i^t$ and η is the expectation of $D(\cdot)$.

Similarly, we repeat the aforementioned procedures with the background-like pool bg^{t-1} and obtain the background similarity S_{bg}^t, so the likelihood of a certain superpixel in frame t belonging to the foreground should be

$$L_{\mathrm{fg}}^t = S_o^t / (S_o^t + S_{\mathrm{bg}}^t). \tag{12}$$

The comprehensive probability of the superpixels to form the foreground is represented as

$$P_{\mathrm{fg}} = \beta \cdot L_{\mathrm{flow}} + \gamma \cdot L_{\mathrm{fg}} + (1 - \beta - \gamma) \cdot L_{\mathrm{binary}}, \tag{13}$$

where β and γ weight the three features. $\beta, \gamma \in (0{,}1)$ and $(\beta + \gamma) \in (0{,}1)$.

Then we jump to Sec. 2.3, where P_{fg} is calculated by Eq. (13) instead of Eq. (3). Just as before, a new superpixel-based CRF model is built and a new segmentation is implemented by graph cut.

2.6 *Exception Handing*

The object-like pool works well most of the time, and the segmented results will theoretically be improved frame by frame. However, when the previous segmented foreground is mixed with some noise, it will have a negative effect on the object-like pool. Furthermore, the error will be accumulated in the current segmentation based on the inaccurate object-like pool, so the vicious circle occurs. This is most likely to happen from the first initial segmentation because the initially segmented result is coarse in general. Therefore, some measures should be taken to prevent the error accumulation.

Let R_n^t be the mean ratio of the number of superpixels in the object-like pool from frame $(t - n)$ to frame t

$$R_n^t = \frac{1}{n} \sum_{i=1}^{n} \frac{N_{\mathrm{sp}}^{t-i+1}}{N_{\mathrm{sp}}^{t-i}}, \tag{14}$$

where N_{sp}^t represents the number of the foreground superpixels from frame t. Therefore, R_1^t is the ratio of the foreground superpixels from frame $(t - 1)$ to frame t. Let R be the set of the normal ratios. Then the state of the object-like pool is represented as

$$\text{state} = \begin{cases} \text{normal}, R_1^t \in R; & \text{if } R_1^t / R_n^{t-1} \in (1 - \lambda, 1 + \lambda) \\ \text{abnormal}, R_1^t \notin R; & \text{others} \end{cases}. \tag{15}$$

The parameter n ($n = 3$ recommended) denotes the number of previous reference frames, and the parameter λ ($\lambda = 0.2$ in our experiments) is the offset of the floor and ceiling bounds, respectively.

Once the state of the object-like pool is abnormal, the exception handling is activated. Then, we discard the object-like pool and the background-like pool and reinitialize the foreground likelihood based on Eq. (3) instead of Eq. (13). The exception handling mechanism is quite effective to avoid error accumulation.

3 Experimental Results

Our algorithm is evaluated by several challenging datasets: “bungalows,” “twoPositionPTZCam,” “highway,” “fall,” “snowFall,” and “blizzard.” They are from the Change Detection 2014 dataset and provide a range of running out of sight, direction change, shadow, dynamic background, partial occlusion, bad weather, and similar color. The proposed algorithm (ours) is compared with a binary mask (BM), ours-shortcut (ours-SC), and PSP-MRF algorithms.[8] Note that the ours-SC algorithm is short of the object-like pool and background-like pool that provide “prior” information for the next segmentation. In addition, only a few sequential frames (less than 25 in all the experiments) are chosen to run our unsupervised algorithm, because we do not need huge frames to build and update the background model or to serve as the training frames. In addition, we only pay attention to a single rigid moving object with the motionless camera in our experiments.

3.1 *Qualitative Evaluation*

The dataset provides various noises: “bungalows” shows the condition where the moving object is running out of the camera’s visual field, so several frames only capture a part of the object. In the “twoPositionPTZCam,” the object continuously changes its moving direction around the corner. The car in “highway” suffers from shadows from the upper trees, and “fall” presents the dynamic background of the swaying leaves and the partial occlusion from the middle tree. In addition, a mass of the snow is falling down in the “snowFall,” in very bad weather. In “blizzard,” the small car has a similar color as the snowy background.

Figure 2 shows the qualitative results of ours, ours-SC, PSP-MRF, and ground truth. BM results are not drawn because they are mostly fragmentary which will make the results cluttered. According to the visual evaluation, the PSP-MRF method performs the worst on average because of the incomplete and even fragmentary segmentations. Furthermore, ours-SC achieves better results than PSP-MRF, although it still lacks some detailed components of the object. By learning the object-like pool and background-like pool, our approach outperforms all the compared methods in terms of robustness and completeness.

3.2 *Quantitative Evaluation*

The performances of different methods are evaluated by two measures: F-measure and percentage of wrong classification (PWC). F-measure is the harmonically weighted balance of precision and recall.[21] F-measure and PWC are specifically defined as

$$\text{F-measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{16}$$

$$\text{PWC} = \frac{\text{FN} + \text{FP}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{17}$$

where TP, TN, FP, and FN are abbreviations for true positive, true negative, false negative, and false negative, respectively. The detailed quantitative performances are shown in Fig. 3. Although ours-SC shows comparatively good results in “snowFall” and “blizzard,” it sometimes produces terrible

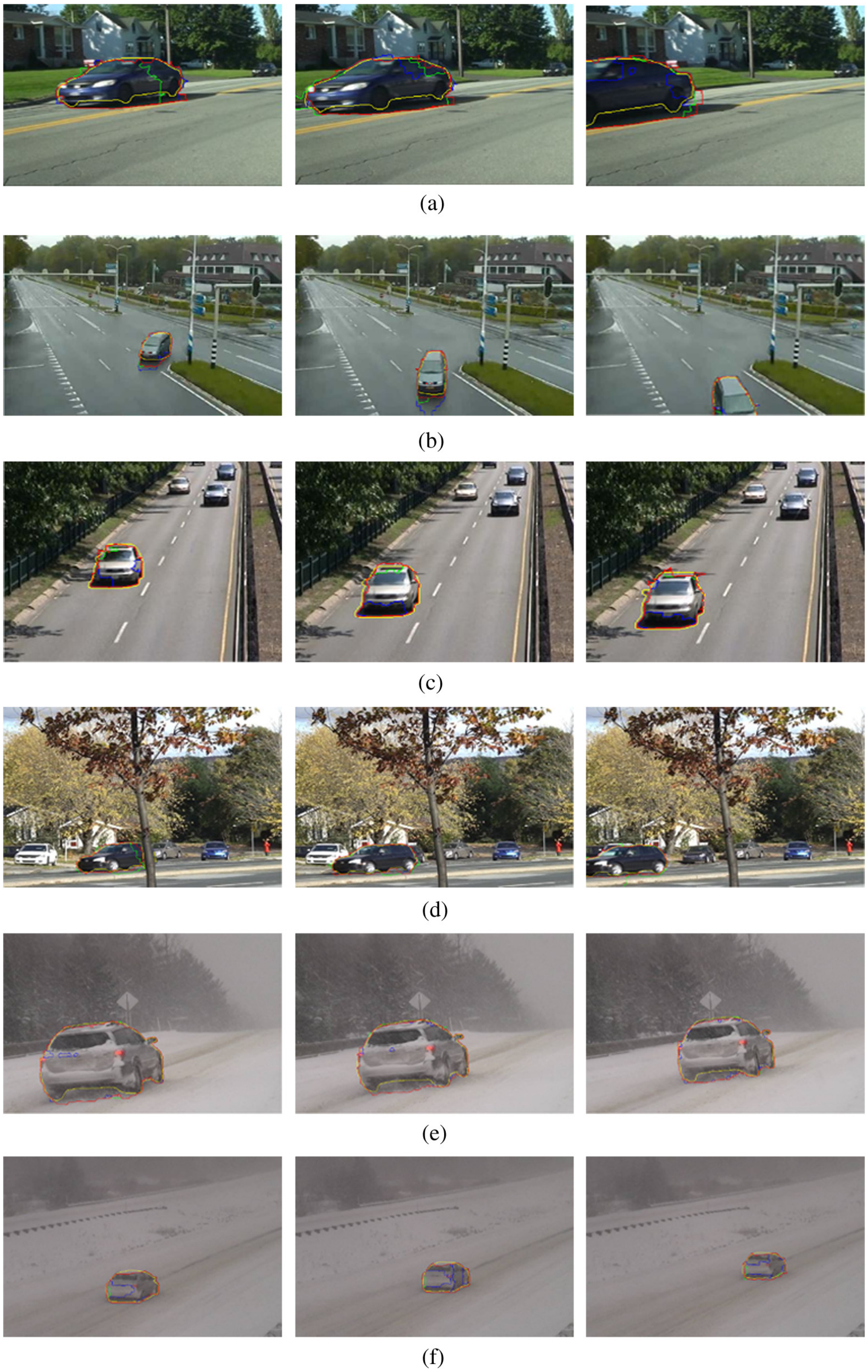

Fig. 2 Visual segmentation results: (a) bunglows, (b) twoPositionPTZCam, (c) highway, (d) fall, (e) snowfall, and (f) blizzard. The results of ours, ours-SC, PSP-MRF, and ground truth are respectively represented by red, green, blue, and yellow curves.

results (see the result of “fall”). We conclude that it is not robust and neither is the PSP-MRF. Above all, the average scores of our method in terms of F-measure and PWC perform the best compared with the others.

3.3 *Impact of Binary Mask*

Binary mask is one of the basic cues which is exploited by PSP-MRF, ours-SC, and ours. Specifically, it makes up the probabilistic superpixels in the PSP-MRF and occupies a weighted part in both ours-SC and ours, so their results are closely related to the binary mask. In the implementation of the binary mask, we use the temporal difference method. Although it is simple and sensitive for detecting changes, it has poor antinoise performance and outputs an incomplete object with “ghosts” (see the rapidly descending magenta line in “bungalows” in Fig. 3).

In Fig. 3, it is easy to see that the blue PSP-MRF line has a certain positive correlation with the magenta BM line. According to Ref. 8, the binary mask directly determines the unary term, which captures the likelihood of superpixels belonging to the foreground. As a result, the performance of PSP-MRF gets worse when the binary mask goes bad. Furthermore, ours-SC method fuses the optical flow and binary mask together, so its performance is partly influenced by the binary mask. Moreover, with the object-like pool and background-like pool, our method is only slightly influenced by the binary mask even when it goes bad (see red line in “bungalows,” “twoPositionPTZCam,” and “blizzard” in Fig. 3). Overall, the proposed algorithm is the least sensitive to the performance of the binary mask.

3.4 *Impact of Optical Flow*

Similar to the binary mask discussed previously, optical flow constitutes one of the elements of ours-SC and ours. However, it is vulnerable to noise that may be generated from the illumination change or an area with the same color. For example, in the “fall” dataset of Fig. 3, the reflection of the ground increases the error of the optical flow and the green line goes bad quickly even though the binary mask is not so bad. In contrast, our algorithm remains the best under this condition. Similar to the binary mask, the proposed algorithm is also the least sensitive to the performance of the optical flow.

3.5 *Effectiveness of Object-Like Pool*

To further evaluate the effectiveness of our object-like pool, a comparison is conducted between the method with (ours) and without the object-like pool (ours-SC). According to the performance in Fig. 3, our proposed algorithm achieves the smoothest and highest F-measure curves and the least PWC on average, while the curves of ours-SC fluctuate heavily and perform worse than ours. The reason is that the object-like pool provides a reliable and continuous way to propagate the

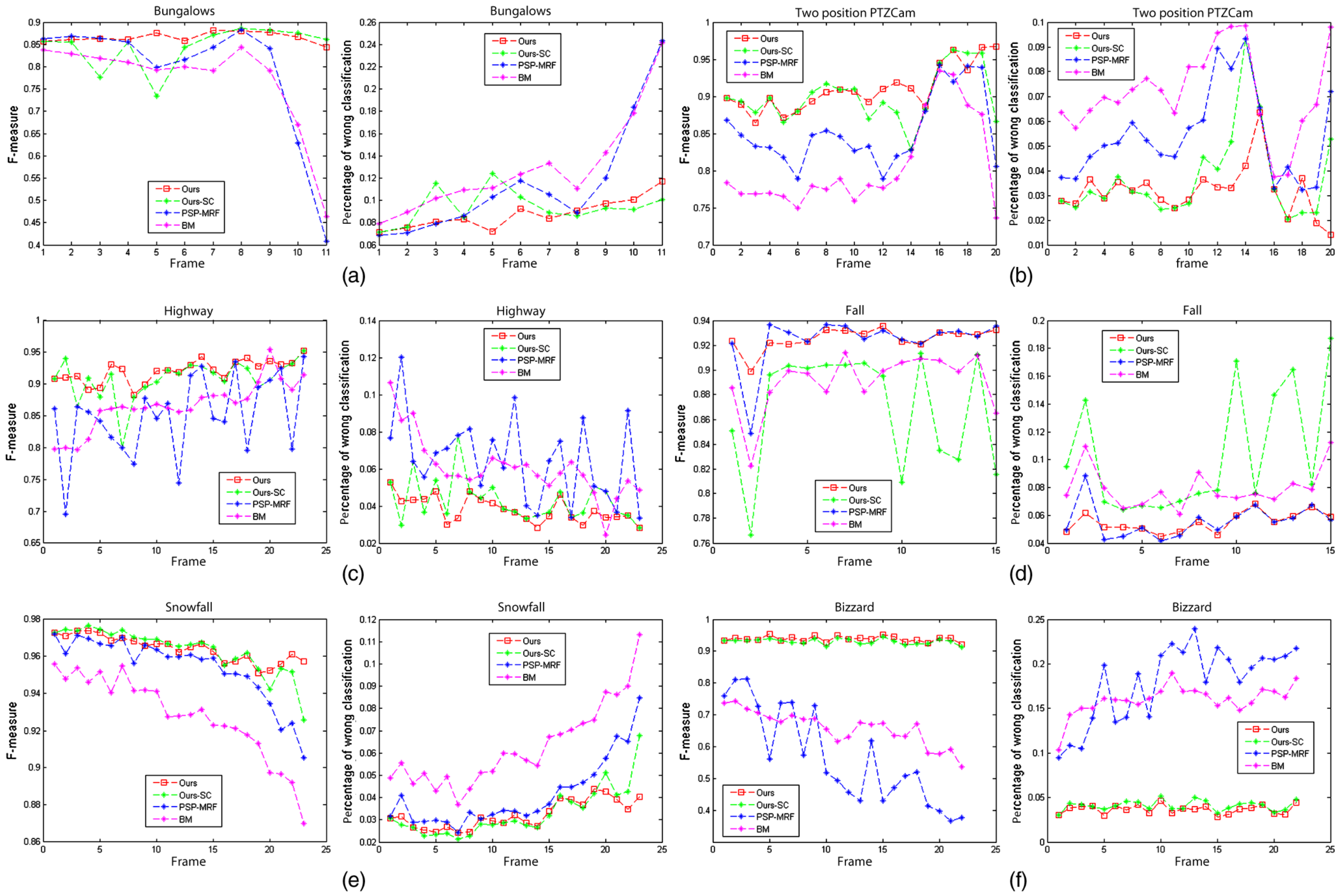

Fig. 3 Performance comparison of different methods. (a) the quantitative result of bunglows, (b) the quantitative result of twoPositionPTZCam, (c) the quantitative result of highway, (d) the quantitative result of fall, (e) the quantitative result of snowfall, and (f) the quantitative result of blizzard.

object against the noise from other features. Besides, the details of the objects with our algorithm can still be improved even when ours-SC has already achieved good results, as with the performances of “snowFall” and “blizzard” as shown in Fig. 3. In brief, the proposed method with an object-like pool achieves more robust and accurate results than the methods without the object-like pool.

3.6 *Impact of Parameters Selection*

To study the sensitivity of parameter selection, different parameters of α, β, and γ are chosen. Taking the typical “bungalows” as an example, we calculate the segmented results based on three groups of parameters and the performance is illustrated in Fig. 4. We call the “bungalows” typical because the last two frames have achieved comparatively satisfying optical flows but terrible binary masks, which are balanced by α, β, and γ. According to the F-measure curves in Fig. 4, the last two points of ours-SC descend quickly with the increasing weight of the binary mask. However, our approach still maintains an excellent performance even while being faced with the awful binary mask. Therefore, our approach is more robust than ours-SC in terms of the parameters.

3.7 *Comparison of Computational Complexity*

The computational complexity is introduced to make a scientific comparison of the time cost in different approaches. We first establish the notations used.

1. Let H and W be the height and width of the video frame.
2. Let h and w be the height and width of the moving region.

Table 1 Computational complexity of different methods.

Method	Computational complexity
Binary mask (BM)	$O(WH)$
PSP-MRF	$O(whTL^2/K) + O(WH) = O(WH)$
Ours-shortcut (ours-SC)	$O(whTL^2/K) + O(WH) + O(K) = O(WH)$
Ours	$O(whTL^2/K) + O(WH) + O(K) + O(NK) = O(WH)$

3. Let K be the total number of the superpixels.
4. Let S be the number of the pixels between two adjacent seeds of the superpixels.
5. Let T be the iterations of superpixel segmentation in the SLIC method.
6. Let L be the length of the search range in the SLIC method.
7. Let N be the number of the neighbors described in Eq. (11).

According to the detailed algorithm of SLIC, it’s running time is $O(whTL^2/K)$. We set $T = 10$ and $T = 3$ for the realization of SLIC in all the experiments, and K is generally larger than 100. Therefore, we have $O(whTL^2/K) \leq O(wh) < O(WH)$. The proposed object-like pool and background-like pool cost $O(NK)$ running time in total, in which we choose $N = 9$ as the nine-connected neighbors

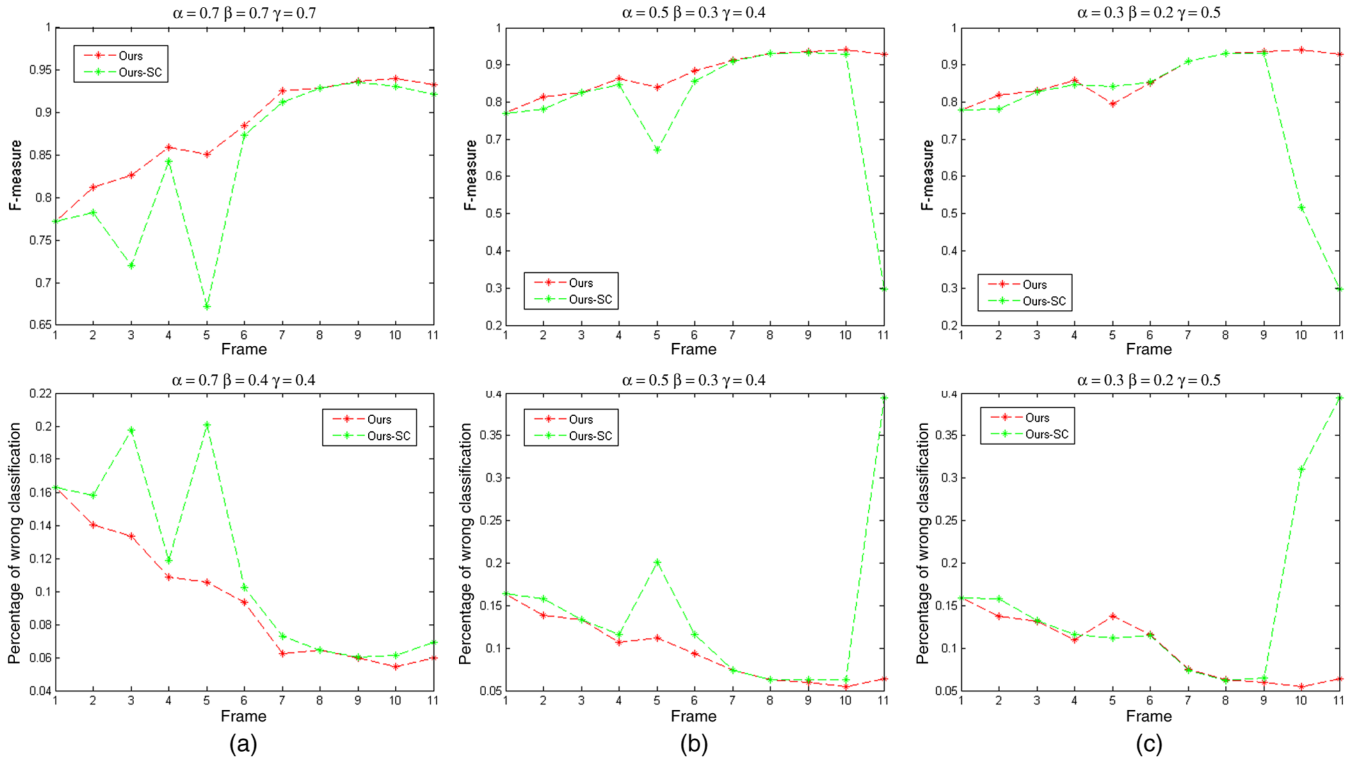

Fig. 4 Performance comparison of different parameters. (a) High weight for optical flow and low weight for binary mask. (b) Equal weights of optical flow and binary mask. (c) Low weight for optical flow and high weight for binary mask.

in Eq. (11). Since the features of the binary mask and optical flow are defined at the superpixel level, we can figure out that they take at most $O(K) \leq O(WH)$ running time. The implementation of graph cut costs $O(wh/S^2) = O(K)$ running time because of $S = \sqrt{wh/K}$.

Based on the mentioned inferences, we compare our approach (ours) in terms of computational complexity with ours-SC, PSP-MRF, and BM in Table 1. We find that the computational complexity of all the methods is equal in polynomial time.

4 Conclusions

We proposed a robust and effective method to improve the unlabeled short video segmentation based on the object-like pool. Our approach exploits the temporal and spatial coherence of appearance and motion features of the moving object to generate the foreground likelihood across the frames. According to the qualitative and quantitative results, our approach exceeds the other compared methods, both in accuracy and robustness, even when the binary mask and optical flow suffer from great interference.

However, the proposed algorithm still has some limitations. Occasionally we need to empirically tune the weighted parameters among different features to produce satisfactory results, so an intelligent and adaptive method to automatically generate weights should be developed. In addition, our method works worse for nonrigid objects than rigid objects because of the conflicting optical flow within them. Therefore, a more generalized algorithm should be proposed to solve this problem in further work.

Acknowledgments

This work is partly supported by the National Natural Science Foundation of China (14ZR1447200).

References

1. S. C. Huang, "An advanced motion detection algorithm with video quality analysis for video surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.* **21**(1), 1–14 (2011).
2. N. C. Mithun, N. U. Rashid, and S. M. M. Rahman, "Detection and classification of vehicles from video using multiple time-spatial images," *IEEE Trans. Intell. Transp. Syst.* **13**(3), 1215–1225 (2012).
3. J. Shotton et al., "Textonboost: joint appearance, shape and context modeling for multi-class object recognition and segmentation," *Lec. Notes Comput. Sci.* **3951**, 1–15 (2006).
4. D. Zhang, O. Javed, and M. Shah, "Video object segmentation through spatially accurate and temporally dense extraction of primary object regions," in *Proc. 2013 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 628–635 (2013).
5. X. M. He, R. S. Zemel, and M. A. Carreira-Perpinan, "Multiscale conditional random fields for image labeling," in *Proc. 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Vol. 2, pp. 695–702 (2004).
6. Z. Q. Tian et al., "Video object segmentation with shape cue based on spatiotemporal superpixel neighbourhood," *IET Comput. Vision* **8**(1), 16–25 (2014).
7. X. F. Wang and X. P. Zhang, "A new localized superpixel Markov random field for image segmentation," in *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME 2009)*, Vol. 1–3, pp. 642–645 (2009).
8. A. Schick, M. Bauml, and R. Stiefelhagen, "Improving foreground segmentations with probabilistic superpixel Markov random fields," in *Proc. 2012 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 27–31 (2012).
9. G. Shu, A. Dehghan, and M. Shah, "Improving an object detector and extracting regions using superpixels," in *Proc. 2013 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3721–3727 (2013).
10. B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *Proc. 2009 IEEE 12th Int. Conf. on Computer Vision (ICCV)*, pp. 670–677 (2009).
11. X. F. Ren and J. Malik, "Learning a classification model for segmentation," in *Proc. Ninth IEEE Int. Conf. on Computer Vision*, Vol. 1, pp. 10–17 (2003).
12. A. Levinshtein et al., "Turbopixels: fast superpixels using geometric flows," *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(12), 2290–2297 (2009).
13. Y. Yuan, J. W. Fang, and Q. Wang, "Robust superpixel tracking via depth fusion," *IEEE Trans. Circuits Syst. Video Technol.* **24**(1), 15–26 (2014).
14. F. Yang, H. C. Lu, and M. H. Yang, "Robust superpixel tracking," *IEEE Trans. Image Process.* **23**(4), 1639–1651 (2014).
15. R. Achanta et al., "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2274–2281 (2012).
16. C. Sutton, A. McCallum, and K. Rohanimanesh, "Dynamic conditional random fields: factorized probabilistic models for labeling and segmenting sequence data," *J. Mach. Learn. Res.* **8**, 693–723 (2007).
17. Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *Int. J. Comput. Vision* **70**(2), 109–131 (2006).
18. V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(2), 147–159 (2004).
19. Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(11), 1222–1239 (2001).
20. Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(9), 1124–1137 (2004).
21. L. Maddalena and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Trans. Image Process.* **17**(7), 1168–1177 (2008).

Xiaoliu Cheng received his BS degree in electronic science and technology from Zhengzhou University, Zhengzhou, China, in 2011. From 2011 to 2012, he studied signal processing at the University of Science and Technology of China, Hefei, China. Currently, he is pursuing his PhD degree at the Shanghai Institute of Microsystem and Information Technology (SIMIT), Chinese Academy of Sciences (CAS), Shanghai, China. His research interests include computer vision, machine learning, and wireless sensor networks.

Wei Lv received his MS degree from Harbin Engineering University, Harbin, China, in 2007. She is an assistant researcher at SIMIT, CAS, Shanghai, China. Her research interests include image processing and wireless sensor networks.

Huawei Liu received his MS degree from Harbin Engineering University, Harbin, China, in 2008. He is an assistant researcher at SIMIT, CAS, Shanghai, China. His research interests include sensor signal processing and wireless sensor networks.

Xing You received her PhD from SIMIT, CAS, Shanghai, China, in 2013. She is an assistant professor at SIMIT, CAS, Shanghai, China. Her research interests include video processing and information hiding.

Baoqing Li received his PhD from the State Key Laboratory of Transducer Technology, Shanghai Institute of Metallurgy, CAS, Shanghai, China, in 1999. Currently, he is a professor at SIMIT, CAS, Shanghai, China. His research interests include signal processing, microelectromechanical systems, and wireless sensor networks.

Xiaobing Yuan received his PhD from the Changchun Institute of Optics, Fine Mechanics and Physics, CAS, Changchun, China, in 2000. Currently, he is a professor at SIMIT, CAS, Shanghai, China. His research interests include wireless sensor networks, information transmission and processing.