HYBRID CENTER-SYMMETRIC LOCAL PATTERN FOR DYNAMIC BACKGROUND SUBTRACTION

Gengjian Xue, Li Song, Jun Sun, Meng Wu

Institute of Image Communication and Information Processing,
Shanghai Jiao Tong University, Shanghai 200240, China
Shanghai Key Labs of Digital Media Processing and Communication, Shanghai 200240, China
{xgjsword, song_li, junsun, wmeng}@sjtu.edu.cn

ABSTRACT

Effective foreground detection in dynamic scenes is a challenging task in computer vision applications. In this paper, we propose a novel background modeling method to tackle this problem. First, we propose a second-order center-symmetric local derivative pattern (CS-LDP) which extracts more detail information compared with the first-order center-symmetric local binary pattern (CS-LBP). Then by concatenating the CS-LBP and CS-LDP histograms, a new hybrid histogram feature is presented. The length of this histogram is much shorter than the local binary pattern (LBP) histogram. Based on this hybrid feature, a novel background modeling method is proposed where the pixel process is modeled with a group of adaptive hybrid histograms. The major advantage of our method is its low complexity. Experiments on three challenging sequences demonstrate that the proposed method is effective and fast, producing comparable results to state-of-art algorithm while reducing the computation time greatly.

Index Terms— Background modeling, local binary pattern (LBP), center-symmetric local binary pattern (CS-LBP), center-symmetric local derivative pattern (CS-LDP).

1. INTRODUCTION

Moving object detection is often the first step in video processing applications, such as transportation, security and surveillance. Its output is usually as an input to a higher level process. Therefore, its performance can have huge effect on the performance of higher level tasks. As a common approach to this problem, background subtraction has been widely used in the last decades, but it is still a difficult task when the background scenes are dynamic in nature, *e.g.* swaying trees, rippling water, moving vegetation.

Gaussian mixture models (GMM) [1] is one of the most popular techniques for background modeling, where each pixel is modeled by a mixture of K Gaussian distributions, and each Gaussian represents the intensity distribution of one of the different environment components. Then the techniques of adaptive learning speed and adaptive component number for each pixel were extended on the standard GMM in the work [2] and [3] respectively. However, these methods have the limitation of Gaussian distribution assumption which does not always hold in practice. Another common method is the nonparametric statistical model. The kernel density estimation (KDE) [4] technique has been proposed for background subtraction. Given the previous pixels, the probability density of the intensities in current frame was estimated by kernel density estimation without any assumptions on distribution. In [5], multimodal kernel density estimation using pixel position and color information was proposed for background modeling. But kernel based methods are computationally intensive and the performance is not very good in dynamic scenes. Li *et al.* [6] proposed to characterize the background with principal features of each pixel according to their statistics, but it relied on look-up tables for training. Theses methods mentioned above assume the pixels are independent in most cases, which degrades their performance in dynamic scenes.

The recently proposed background subtraction approach based on local binary pattern (LBP) [7] has received noticeable attention. It models each pixel with a group of LBP histograms and shows promising performance in dynamic scenes. However, the LBP operator produces long feature set since it only adopts the first-order gradient information between center pixel and its neighbors. Center-symmetric local binary pattern (CS-LBP) [8] used for matching is an effective extension to LBP. Although it has much shorter feature length, it can not contain enough information for background modeling. To the best of our knowledge, the thought of high-order local derivative pattern has shown many advantages and was successfully applied in face recognition [9] and image representation [10], but it has not been appropriately used for background subtraction.

In this paper, we first propose a second-order center-

This work was partly supported by 973 Program (2010CB731401, 2010CB731406), NSFC (60702044,60902073), the 111 project and STCSM (10511500802).

978-1-61284-350-6/11/$26.00 ©2011 IEEE

symmetric local derivative pattern (CS-LDP) operator which extracts more detail local information than CS-LBP. Then we concatenate the CS-LBP histogram and CS-LDP histogram to get a new hybrid feature. Based on this feature, we propose our background subtraction method. Experiments on challenging sequences indicate that our method can produce comparable results while using less computation time compared to the LBP based method.

The rest of this paper is organized as follows: In Section 2, we introduce our CS-LDP operator and propose the novel hybrid feature. In Section 3, the background modeling method based on this feature is presented. Experimental results comparison and evaluation are conducted in Section 4. Finally, we conclude the paper with some discussions.

2. PROPOSED FEATURE

2.1. LBP and CS-LBP operator

LBP operator was first defined as a grayscale invariant texture measure. The LBP operator labels the pixels of an image region by thresholding the neighborhood of each pixel with the value of central pixel and concatenating the results binomially to form a decimal number:

$$\mathrm{LBP}_{\mathrm{R,N}} = \sum_{i=0}^{N-1} s(z_i - z_c)2^i \tag{1}$$

where z_c corresponds to the grey value of central pixel (x_c, y_c) and z_i to the grey values of N equally spaced pixels on a circle R. The thresholding function $s(x)$ is defined as:

$$s(x) = \begin{cases} 1 & x \geqslant 0 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

LBP operator has the property of being invariant to illumination changes and has been used in many fields. CS-LBP operator is an effective extension to LBP by reducing the feature length greatly. Different from LBP that compares each neighbor with the central pixel, it compares the grey values of pairs of pixels in center symmetric direction:

$$\mathrm{CS\text{-}LBP}_{\mathrm{R,N}} = \sum_{i=0}^{(N/2)-1} s(z_i - z_{i+(N/2)})2^i \tag{3}$$

where z_i and $z_{i+(N/2)}$ are the values of neighborhood pixels in center symmetric direction.

2.2. CS-LDP operator

Theoretically, LBP can be considered as a non-directional first order local pattern, which is the binary result of the first order derivative image. CS-LBP produces much shorter feature set than LBP, but it is also a first order local pattern in center symmetric direction and it ignores the central pixel information. Since the first order derivative pattern is incapable of

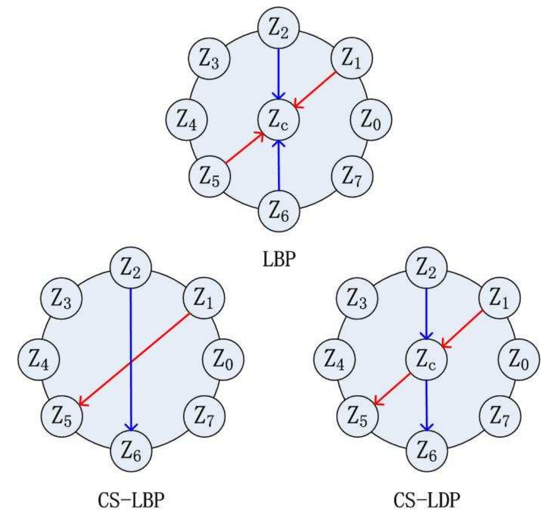

Fig. 1. Example of LBP, CS-LBP and CS-LDP operator with 8 neighbors.

describing more information and the high order local derivative patterns in [9] [10] produce very long feature set that is not suitable for background modeling, here we first propose a novel CS-LDP operator which is a second order derivative pattern in center symmetric direction. CS-LDP operator can capture more detail information while having the same feature length to CS-LBP, and it is defined as:

$$\mathrm{CS\text{-}LDP}_{\mathrm{R,N}} = \sum_{i=0}^{(N/2)-1} t[(z_i - z_c)\cdot(z_c - z_{i+(N/2)})]2^i \tag{4}$$

where the parameters z_c, z_i, $z_{i+(N/2)}$, R, N are the same as above. The threshold function $t(\cdot,\cdot)$ is used to determine the types of local pattern transition and is defined as:

$$t(x_1, x_2) = \begin{cases} 0 & \text{if } x_1 \cdot x_2 > 0 \\ 1 & \text{if } x_1 \cdot x_2 \leqslant 0 \end{cases} \tag{5}$$

Fig.1 shows an example of obtaining the LBP, CS-LBP and CS-LDP patterns with eight neighbors around z_c. It can been seen that LBP encodes all eight direction first order derivative binary result. CS-LBP pattern calculates the first order center symmetric derivatives at z_c along 0°, 45°, 90° and 135° directions that can be written as:

$$\mathrm{CS\text{-}LBP}_{0^\circ}(z_c) = s(z_0 - z_4) \tag{6}$$

$$\mathrm{CS\text{-}LBP}_{45^\circ}(z_c) = s(z_1 - z_5) \tag{7}$$

$$\mathrm{CS\text{-}LBP}_{90^\circ}(z_c) = s(z_2 - z_6) \tag{8}$$

$$\mathrm{CS\text{-}LBP}_{135^\circ}(z_c) = s(z_3 - z_7) \tag{9}$$

CS-LDP pattern is considered to encode the second order center symmetric derivatives at z_c along 0°, 45°, 90° and 135° directions. They can be represented as:

$$\mathrm{CS\text{-}LDP}_{0^\circ}(z_c) = t[(z_0 - z_c), (z_c - z_4)] \tag{10}$$

$$\text{CS-LDP}_{45^\circ}(z_c) = t[(z_1 - z_c), (z_c - z_5)] \tag{11}$$

$$\text{CS-LDP}_{90^\circ}(z_c) = t[(z_2 - z_c), (z_c - z_6)] \tag{12}$$

$$\text{CS-LDP}_{135^\circ}(z_c) = t[(z_3 - z_c), (z_c - z_7)] \tag{13}$$

From the above illustration, we find that CS-LDP and CS-LBP both compute the same directional derivative information and produce the same feature length. They have much shorter length than LBP. Given 8 neighbors for example, CS-LBP and CS-LDP both produce 4 bits binary sequence while LBP yields 8 bits. Besides, as a complement, CS-LDP captures more detail information by encoding the relationship between central pixel and center symmetric neighbors.

2.3. Hybrid histogram feature

CS-LBP is to encode the coarse gradient information while CS-LDP focuses on describing the detail of the local region. Combining the advantages of these two patterns, we propose a novel histogram feature for background modeling. Let R_{region} be a circular region around the center pixel (x, y). The CS-LBP histogram $\text{H}_{\text{CS-LBP}}$ and CS-LDP histogram $\text{H}_{\text{CS-LDP}}$ over this region are as follows:

$$\text{H}_{\text{CS-LBP, j}} = \sum_{(x,y)\in R_{region}} I\{\text{CS-LBP}_{\text{N,R}}(x, y) = j\} \tag{14}$$

$$\text{H}_{\text{CS-LDP, j}} = \sum_{(x,y)\in R_{region}} I\{\text{CS-LDP}_{\text{N,R}}(x, y) = j\} \tag{15}$$

where $j = 0, \ldots, 2^{(N/2)} - 1$, $\text{H}_{\text{CS-LBP, j}}$ and $\text{H}_{\text{CS-LDP, j}}$ are the histogram values at j^{th} bin of $\text{H}_{\text{CS-LBP}}$ and $\text{H}_{\text{CS-LDP}}$ respectively. $I(A)$ is defined as:

$$I(A) = \begin{cases} 1 & \text{if A is true} \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

Then, our hybrid histogram feature H_{hybrid} is defined as the concatenation of the $\text{H}_{\text{CS-LBP}}$ and $\text{H}_{\text{CS-LDP}}$ histograms:

$$\text{H}_{\text{hybrid}} = \text{H}_{\text{CS-LBP}} \_\_ \text{H}_{\text{CS-LDP}} \tag{17}$$

where "__" means the concatenation operation.

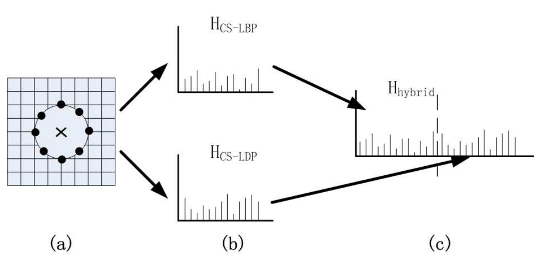

Fig. 2. The computing procedure of H_{hybrid} histogram on center pixel (marked with X).

The H_{hybrid} histogram is suitable for background modeling. First, it is robust to monotonic grey level changes. Next, it produces short feature length and has low complexity. Given N neighbors, LBP produces 2^N histogram bins, CS-LBP and CS-LDP both yield $2^{(N/2)}$ bins. By concatenation, the length of H_{hybrid} is only $2 * 2^{(N/2)}$ which is much shorter than LBP histogram. Once more, the greatly feature set reduction does not result in much information loss because high order local pattern captures more detail information.

3. BACKGROUND SUBTRACTION BASED ON HYBRID HISTOGRAM

In this section, we introduce our method for background subtraction. The algorithm can be divided into two parts: background modeling, background maintaining and foreground detection. We model each pixel identically in the algorithm.

3.1. Background modeling

As for background modeling, we consider the feature vector of the pixel over time as a pixel process and the proposed hybrid histogram is used as the feature vector. The dynamic background model of the pixel is built using a group of hybrid histograms $\{m_1, m_2, \cdots, m_K\}$, where K is the number of models selected by user. Each histogram has a weight between 0 and 1, and the sum of all K histograms equals to one. Bigger weight means higher probability it may be as a background representation. The weight of the k_{th} histogram is denoted as ω_k.

3.2. Model maintaining and foreground detection

When a new frame is arriving, we first compute its hybrid histogram feature in current frame. Next, we sort the existing K histograms in descending order according to its weight ω. Then the first B histograms are selected as background representation:

$$\omega_1 + \omega_2 + \cdots + \omega_B > T_B \tag{18}$$

where $T_B \in (0, 1)$ is defined by user. The larger of T_B is, the more histograms are selected as background feature. Then the current histogram is compared with the selected B features for proximity measurement. Here, we adopt histogram intersection as the similarity measure according to [7].

$$\cap(m, h) = \sum_{j'=0}^{L-1} \min(m_{j'}, h_{j'}) \tag{19}$$

where j' means the histogram bin index and L stands for the histogram length, m and h are existing and current histogram features respectively. Then the user-settable threshold T_P is used to compare with the similarity value. If the similarity is below the value T_P for all background histograms, we consider that the current feature is not matching with any background distributions. Then, the corresponding pixel is classified as foreground, and the model histogram with the lowest weight is replaced with current histogram and given to a low initial weight. If the similarity is higher than T_P for at least one of the background feature, then the corresponding pixel is labeled as background and we update the histogram feature with the highest similarity value. The best matching model histogram denoted by m_k is updated with the new feature as follows:

$$m_k = (1-\alpha)m_k + \alpha h \tag{20}$$

meanwhile the weights for all models are updated as:

$$\omega_k = (1-\alpha)\omega_k + \alpha M_k \tag{21}$$

where M_k is 1 for the matching feature and 0 for the others. $\alpha \in (0, 1)$ is the learning rate, bigger learning rate means faster background adaption.

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed approach, three challenging video sequences characterized by dynamic scenes are adopted for testing. These sequence are all publicly available, and the groundtruth images are obtained by manually segmented. The proposed approach is also compared with three widely used approaches including GMM [1], KDE [4] and LBP [7]. For all algorithms, no morphological operation is applied. Both visual and numerical methods are used for comparison. The *Precision* and *Recall* rates are used for evaluation. (TP is number of positives correctly detected, FP is number of positives incorrectly detected and FN is number of true positives not detected.)

$$\text{Precision} = \frac{TP}{TP + FP} \tag{22}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{23}$$

The first sequence is from [11] which involves heavily waving trees. The second experiment is conducted on rippling water sequence and the third test sequence is about campus environment containing moving tree branches. The last two sequences are both from [6]. All the three sequences are containing dynamic background but each of them has different

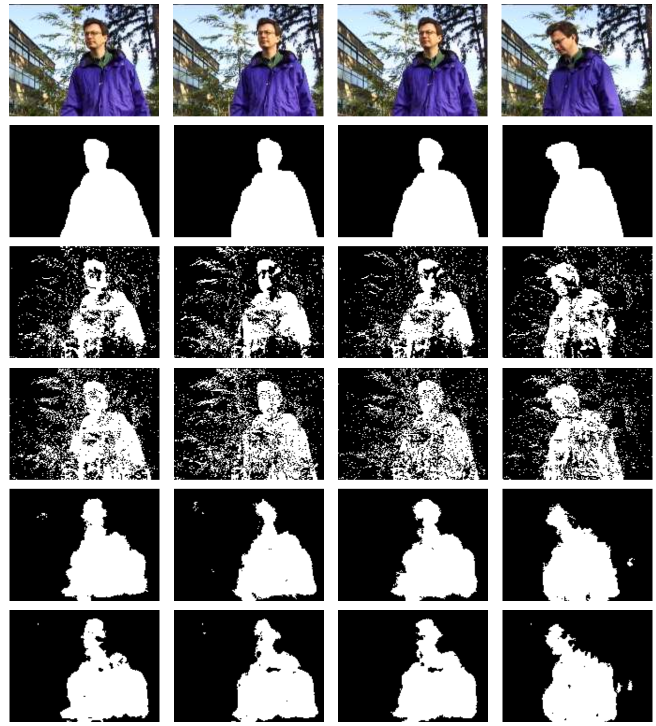

Fig. 3. Comparison results on waving trees. The top row is the original frames named as 247^{th}, 250^{th}, 252^{th} and 254^{th} frames. The second row is corresponding ground truth frames. The third, forth and fifth rows are the results obtained by GMM, KDE and LBP methods respectively. The last row is the results obtained by proposed method.

situation. We run both GMM and KDE algorithms with default parameters. For fair comparison, most parameters for LBP and our method are the same for the three experiments. We only turn the threshold value T_P according to sequence environment in order to get better results. The values of the parameters for LBP and our method are given in Table.1. The learning rate is set to $\alpha = 0.01$ for all experiments.

As shown in the figures, we can see that GMM and KDE methods detect large number of dynamic background pixels as foreground and many foreground positives on the inner areas are not detected. In contrast, LBP and our method are more robust to dynamic scenes. They can distinguish background pixels from true moving objects. Meanwhile, most part of the foreground is correctly detected. Table.2∼4 are the corresponding quantitative evaluations for each sequence which further prove the effectiveness of our method.

From quantitative comparisons above, we can see that LBP and our approach outperform the GMM and KDE methods. This is because LBP and our method are region-based which is more tolerable to dynamic pixels and noises. In addition, our approach produces comparable results to LBP method. Although the *Precision* is slightly lower than LBP,

Table 1. The parameter values of LBP and our method for the results in Figs. 3 to 5

Fig.	K	N	R	R_{region}	T_B	T_P^{LBP}	T_P^{Our}
3	4	8	2	9	0.80	0.60	0.80
4	4	8	2	9	0.80	0.57	0.74
5	4	8	2	9	0.80	0.61	0.84

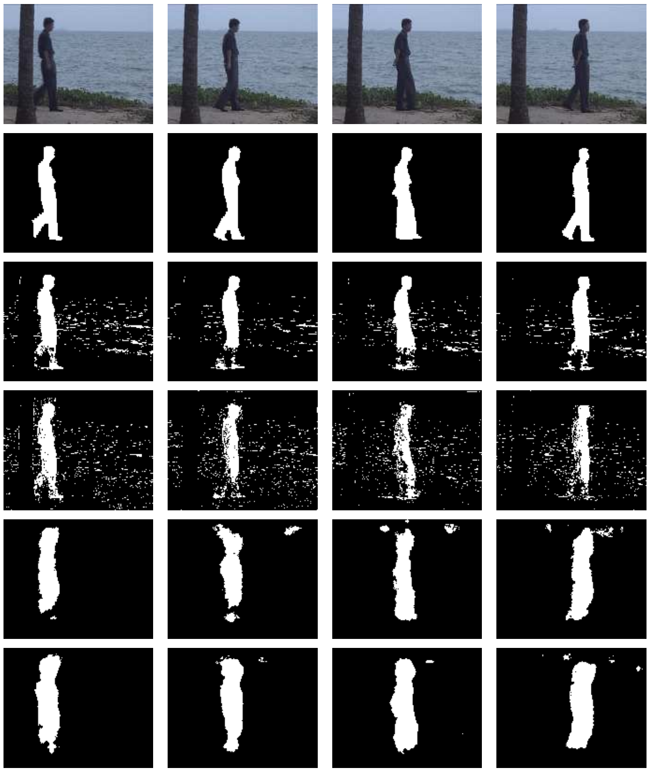

Fig. 4. Comparison results on rippling water. The top row is the original frames named as 1498^{th}, 1508^{th}, 1514^{th} and 1523^{th} frames. The second row is corresponding ground truth frames. The third, forth and fifth rows are the results obtained by GMM, KDE and LBP methods respectively. The last row is the results obtained by proposed method.

the *Recall* rate detected by ours is higher.

Another advantage of our method is that it has low complexity than LBP. In our three experiments, the feature dimension in LBP is $2^8 = 256$ while in our method the feature length is just $(2 \times 2^{8/2}) = 32$. By comparing, the feature dimension is reduced by 87.5% which shows the low complexity of our method, and the feature reduction also results in less computation time. Table.5 shows the speed comparison for both methods on three sequences. As an example, we test all the algorithms on Matlab7.1 on PC computer with 2.4GHz Intel CPU, 2G RAM. The frame size in the first sequence is 120×160 and 128×160 for the last two sequences. We can see that our approach reduces about 25%-29% computation time per frame than LBP method.

From above experiments, we can conclude that our method has low complexity and is robust to dynamic scenes. First, it yields comparable performance to LBP method, with slightly lower precision and higher recall rates. More important, our approach is more faster than LBP method. The reason is that our method adopts small feature set for background modeling which saves the computation time, but it dose not result in much information loss since high order local pattern can capture more detail information.

Table 2. Quantitative comparison of Precision and Recall rates on waving trees sequence

Method		GMM	KDE	LBP	Ours
Precision(%)	247_{th}	75.44	71.01	92.47	90.32
	250_{th}	70.69	66.64	94.78	91.88
	252_{th}	75.73	70.65	92.35	90.84
	254_{th}	80.19	77.56	95.00	93.50
Recall(%)	247_{th}	64.04	79.88	87.99	88.09
	250_{th}	61.49	72.36	85.66	88.52
	252_{th}	63.07	70.95	88.57	89.44
	254_{th}	64.35	75.25	88.86	89.05

Table 3. Quantitative comparison of Precision and Recall rates on rippling water sequence

Method		GMM	KDE	LBP	Ours
Precision(%)	1498_{th}	63.22	56.50	73.74	67.07
	1508_{th}	75.17	56.50	62.42	64.87
	1514_{th}	72.42	61.93	75.41	70.38
	1523_{th}	71.63	63.02	67.31	63.41
Recall(%)	1498_{th}	85.17	86.76	76.13	84.40
	1508_{th}	76.68	70.16	69.96	80.30
	1514_{th}	80.21	68.48	85.03	90.71
	1523_{th}	82.83	70.13	85.10	86.28

5. CONCLUSION

In this paper, we first propose a novel CS-LDP operator which can extract more detail local information. Then by combining the histograms of CS-LBP and CS-LDP, we define a new hybrid histogram feature. Based on this feature, we propose our background subtraction method. Experiments on challenging sequences demonstrate that our method has low complexity and is effective to detect true foreground in dynamic scenes.

Furthermore, we believe that the combination of CS-LBP and CD-LDP operators is not limited to the field of background modeling. It is also expected to be effective in other research areas, such as object recognition and matching.

6. REFERENCES

[1] C. Stauffer and W.E.L. Grimsom, "Adaptive background mixture models for real-time tracking," *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 246–252, 1999.

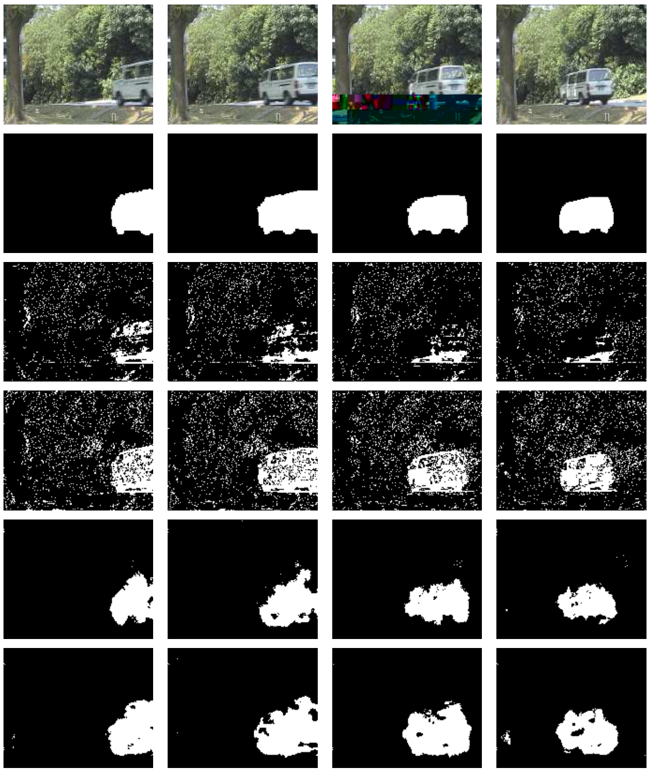

Fig. 5. Comparison results on campus sequence. The top row is the original frames named as 1202^{th}, 1204^{th}, 1206^{th} and 1208^{th} frames. The second row is corresponding ground truth frames. The third, forth and fifth rows are the results obtained by GMM, KDE and LBP methods respectively. The last row is the results obtained by proposed method.

Table 4. Quantitative comparison of Precision and Recall rates on campus sequence

Method		GMM	KDE	LBP	Ours
Precision(%)	1202_{th}	39.74	43.19	88.82	77.09
	1204_{th}	40.31	51.71	79.82	75.68
	1206_{th}	31.14	48.11	87.61	71.00
	1208_{th}	19.40	45.48	89.63	72.34
Recall(%)	1202_{th}	44.54	78.14	85.06	92.75
	1204_{th}	33.21	79.63	72.29	84.23
	1206_{th}	23.79	77.93	86.45	92.14
	1208_{th}	15.39	81.86	82.23	88.52

Table 5. Speed comparison of between LBP and our method (in seconds)

Method	Average running time per frame		
	Waving trees	Rippling water	Campus
LBP	8.82	9.03	9.17
Ours	6.33	6.73	6.86
Time reduction	28.23%	25.47%	25.19%

[2] D.S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 827–832, 2005.

[3] Z. Zivkovic and F. van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern Recognition Letters*, vol. 27, no. 7, pp. 773–780, 2006.

[4] A. Elgammal, R. Duraiswami, D. Harwood, and L.S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance," *Proc. IEEE*, vol. 90, no. 7, pp. 1151–1163, 2002.

[5] Yaser Sheikh and Mubarak Shah, "Bayesian modeling of dyanmic scenes for object detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1178–1192, Nov 2005.

[6] Liyuan Li, Weimin Huang, Irene Yu-Hua Gu, and Qi Tian, "Statistical modeling of complex backgrounds for foreground object detection," *IEEE Trans. on Image Processing*, vol. 13, no. 11, 2004.

[7] M. Heikkilä and M.Pietikäinen, "A texture-based method for modelling the background and detecting moving objects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 657–662, 2006.

[8] Marko Heikkilä, Matti Pietikäinen, and Cordelia Schmid, "Description of interest regions with local binary patterns," *Pattern Recognition*, vol. 42, no. 3, pp. 425–436, 2009.

[9] Baochang Zhang, Yongsheng Gao, Sanqiang Zhao, and Jianzhuang Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE Trans. on Image Processing*, vol. 19, no. 2, pp. 533–544, 2010.

[10] Sanqiang Zhao, Yongsheng Gao, and Terry Caelli, "High-order circular derivative pattern for image representation and recognition," *Proc. 20th International Conference on Pattern Recognition*, pp. 2246–2249, 2010.

[11] K. Toyama, B. Brumitt, J. Krumm, and B. Meyers, "Wallflower: Principles and practice of background maintenance," *Proc. IEEE International Conferences on Computer Vision*, vol. 1, pp. 255–261, 1999.