DYNAMIC BACKGROUND SUBTRACTION BASED ON SPATIAL EXTENDED CENTER-SYMMETRIC LOCAL BINARY PATTERN

Gengjian Xue, Jun Sun, Li Song

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

ABSTRACT

Moving objects detection in dynamic scenes is a challenging task in many computer vision applications. Traditional background modeling methods do not work well in these situations since they assume a nearly static background. In this paper, a novel operator named spatial extended center-symmetric local binary pattern (SCS-LBP) for background modeling is proposed. It extracts spatial and temporal information simultaneously while has low complexity compared to the local binary pattern (LBP) operator. Then combining this operator with an improved temporal distribution estimation scheme, we propose a new background subtraction method. In our method, each pixel is modeled by a group of adaptive SCS-LBP histograms, which provides us with many advantages compared to traditional ones. Experimental results demonstrate the effectiveness and robustness of our method.

Keywords— Background modeling, object detection, spatial extended center-symmetric local binary pattern, online estimation

1. INTRODUCTION

Moving objects detection in video sequences is one of the main tasks in many computer vision applications, such as industrial automation, transportation, security and surveillance. Its output is as an input to a high-level process, making it a critical part of the system. Background subtraction is an common approach for this task. The performance of background subtraction depends mainly on building and maintaining an adaptive background model. Although many methods have been presented to model the background, it is still a challenging task since the background scenes are usually dynamic in nature, *e.g.* illumination changes, swaying trees, rippling water, flickering monitors.

One of the most widely used approaches for background modeling is the Gaussian mixtures models (GMM) [1]. In this technique, each pixel is modeled independently using a mixture of Gaussians and updated by an online approximation. Many authors have proposed improvements and extensions to this algorithm. In [2], the number of components of the mixture is constantly adapted for each pixel. Lee [3] presented an adaptive learning rate for each Gaussian model to improve the convergence rate. However, the Gaussian assumption for the pixel intensity distribution does not always hold in dynamic scenes and these methods may fail. Another popular method was proposed by Elgammal et al. [4]. This method utilizes a nonparametric kernel density estimation (KDE) technique for representing the background. The probability density function for pixel intensity is estimated directly from the data without any assumptions of the underlying distributions. Sheikh and Shah [5] improved the kernel estimation model by combing both color and location information of pixels. But kernel based methods are memory cost and the performance is good only if the background image does not change too much.

Pixel-based methods introduced above assume each pixel is independent, which restricts their use in dynamic background. In contrast, many methods using spatial information have also been proposed. Recently, the LBP based background modeling method proposed in [6] has attracted great attention. This method models each pixel by a group of LBP histograms and shows promising performance in dynamic scenes. However, LBP operator is not so efficient for background modeling since it is sensitive to noise, produces long histograms and does not consider temporal information. In [7], a spatial-temporal LBP was proposed for background modeling, but the computational load was increased and comparative results with the LBP based method were not presented. Center-symmetric local binary pattern (CS-LBP) which was first used for matching and object category classification [8] is an effective extension to LBP. This operator not only has the property of illumination invariance, but also produces short histograms and be more robust to noise.

In this paper, we first extend the CS-LBP operator from spatial domain to spatial-temporal domain and propose a novel operator named SCS-LBP which extracts spatial and temporal information simultaneously. Then combining the SCS-LBP operator with an improved temporal information estimation scheme, we present a new background modeling approach. Experiments on challenging sequences indicate that our method can achieve high accurate detection in dynamic scenes while reducing the computational complexity compared to the LBP based method.

The rest of the paper is organized as follows: In Section 2, we describe the LBP, CS-LBP operator and the proposed SCS-LBP

978-1-4244-7493-6/10/$26.00 ©2010 IEEE ICME 2010

operator. Background modeling and maintaining method based on SCS-LBP is presented in Section 3. Experimental results and conclusions are presented in Section 4 and Section 5.

2. PROPOSED OPERATOR

2.1. LBP and CS-LBP operator

LBP operator was first designed for texture description [9]. This operator describes each pixel by comparing its value with neighbors. If the neighboring pixel value is higher or equal, the value is set to one, otherwise set to zero. Then the concatenation of binary pattern over the neighborhood converted into a decimal number as a unique descriptor for each pixel:

$$\text{LBP}_{\text{R,N}}(x,y) = \sum_{i=0}^{N-1} s(p_i - p_c)2^i \tag{1}$$

where p_c corresponds to the grey level of central pixel $p_{(x,y)}$ and p_i to the grey levels of N equally spaced pixels on a circle of radius R. The function $s(x)$ is defined as follows:

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

LBP features have proven to be robust to monotonic illumination changes. CS-LBP operator is an effective extension to LBP. Instead of comparing each neighbor pixel with central pixel, it compares the grey levels of pairs of pixels in center-symmetric direction:

$$\text{CS-LBP}_{\text{R,N}}(x,y) = \sum_{i=0}^{(N/2)-1} s(p_i - p_{i+(N/2)})2^i \tag{3}$$

where p_i and $p_{i+(N/2)}$ correspond to grey levels of center-symmetric pairs of pixels.

2.2. SCS-LBP operator

LBP and CS-LBP operators only consider spatial information. When detecting moving objects in video sequences, temporal information is also very important and useful. Aiming at background modeling and subtraction, we propose a novel region operator SCS-LBP which extracts both spatial texture and temporal motion information at the same time as show in Fig.1, the operator is defined as:

$$\begin{aligned} \text{SCS-LBP}_{\text{R,N}}(x,y,t) = & \sum_{i=0}^{(N/2)-1} s(p_{(i,t)} - p_{(i+(N/2),t)})2^i \\ & + f(p_{(x,y,t)} - \bar{\mu}_{(x,y,t-1)})2^{N/2} \end{aligned} \tag{4}$$

$p_{(i,t)}$ and $p_{(i+(N/2),t)}$ stand for grey levels of center-symmetric pairs of pixels in the current frame. R, N and $s(x)$ function are the same as above. The binary function $f(t)$ is used for describing temporal relationship. We set zero to $f(t)$ if current pixel $p_{(x,y,t)}$ is considered as background, satisfying the condition that its value is within 2.5 its standard deviation from its

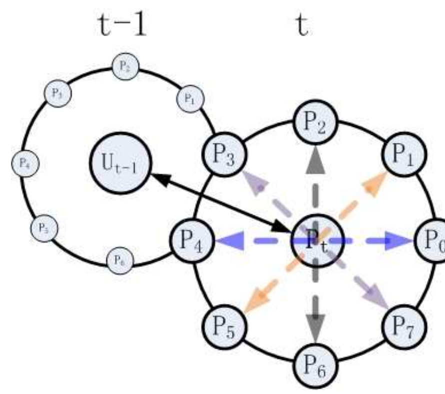

Fig. 1. SCS-LBP operator for a neighborhood of 8 pixels

mean value. Otherwise $p_{(x,y,t)}$ is classified as foreground and $f(t)$ is set to one:

$$f(t) = \begin{cases} 0 & \text{if } |p_{(x,y,t)} - \bar{\mu}_{(x,y,t-1)}| < 2.5 * \bar{\sigma}_{(x,y,t-1)} \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

where $\bar{\mu}_{(x,y,t-1)}$ and $\bar{\sigma}_{(x,y,t-1)}$ are estimated mean value and standard deviation respectively corresponding to pixel $p_{(x,y)}$.

SCS-LBP operator has some advantages for background modeling in dynamic scenes. First, it has the property of invariant to monotonic grey level changes. Next, compared to LBP, it captures better gradient information through comparing pairs of neighbors, making it more discriminative. Once more, it produces short histograms and has low computational complexity. Using the neighbor equals 8 for example, LBP produces $2^8 = 256$ histogram bins while SCS-LBP only yields $2^{(8/2)+1} = 32$ bins. Finally, it can extract spatial and temporal information simultaneously.

3. BACKGROUND MODELING BASED ON SCS-LBP HISTOGRAM

In this section, we propose a novel background modeling approach based on SCS-LBP histogram. The algorithm can be divided into two phases, background modeling, background update and foreground detection. Two assumptions should be noted. One is that the camera is static as in the relevant references. The other is that we model each pixel of the background identically.

3.1. Background modeling

In our method, we select the SCS-LBP histogram as the feature vector for modeling the background. The SCS-LBP histogram for a particular is computed over a circular region of radius R_{region} around it. When building the background, we consider the feature vectors of the pixel over time as a pixel process. The background model for the pixel consists of a group of adaptive SCS-LBP histograms,$\{m_0, m_1, \ldots, m_{K-1}\}$, where K is selected by the user. Each mode histogram has a weight between 0 and 1 so that the weights of all K histograms sum up to one. The bigger the weight, the higher probability of being a background histogram. We denote the weight of the k_{th} histogram as ω_k.

3.2. Model updating and foreground detection

When a new frame is arriving, we first compute the current SCS-LBP histogram for the pixel in the frame. As for temporal information extraction, we adopt an improved estimation method [10] to estimate the mean value and deviation of the pixel. This method has proven more efficient in dynamic scenes.

$$\bar{\mu}_t = (1-\beta)\,\bar{\mu}_{t-1} + \beta I_t \qquad (0 < \beta < 1) \tag{6}$$

$$\bar{\sigma}_t^2 = (1-\beta)\,\bar{\sigma}_{t-1}^2 + \frac{\beta}{2}(I_t - I_{t-1})^2 \tag{7}$$

where I_t and I_{t-1} are current and previous pixel values respectively, β is a learning rate.

Next, we sort the existing K histograms in descending order according to ω_i and select the first B models as the background histograms:

$$\omega_1 + \omega_2 + \ldots + \omega_{B-1} > T_B \tag{8}$$

where $T_B \in (0, 1)$ is a user defined parameter. Then the current histogram is compared with the selected B background histograms using the proximity measure. Here, we choose the histogram intersection as the measure according to [6].

$$\cap(m, h) = \sum_{i=0}^{L} (m(i), h(i)) \tag{9}$$

where i is the histogram bin index, L is the number of histogram bins, m and h are the existing and current histograms respectively. If the similarity is below the threshold T_P for all background histograms, the pixel is consider as foreground. Then, 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 the threshold T_P for at least one background histogram, the pixel is labeled as background and we update the model histogram with the highest proximity value. The best matching model histogram denoted by m_k is adapted with the new data by updating its bins as follows:

$$m_k = (1-\alpha)m_k + \alpha h \tag{10}$$

meanwhile the weight of all model histograms are also updated:

$$\omega_k = (1-\alpha)\omega_k + \alpha M_k \tag{11}$$

where M_k is 1 for the best matching histogram and 0 for the others. α is the learning rate, bigger learning rate means faster adaption.

The procedure of our proposed background modeling algorithm is as the Fig.2.

```
determine parameters and build initial background model
For(each pixel in current frame) do
  ➢ estimate the mean value and deviation of the pixel
    according to Eq.(6)(7)
  ➢ compute the SCS-LBP binary pattern using Eq.(4)
  ➢ build current SCS-LBP histogram h over R_region
  ➢ choose background histograms according to value T_B
    as Eq.(8)
  ➢ compare h with background histogram m using
    histogram intersection measure by Eq.(9)
    If (similarity_value > T_P)
      Cur_Pixel = background
      update the best matching histogram m_k, and
      histogram weights using Eq.(10)(11)
    Else
      Cur_Pixel = foreground
      histogram with lowest weight is replaced with h
      initial weight is given to h
    End
End
```

Fig. 2. Overview of the proposed framework

4. EXPERIMENTS AND DISCUSSIONS

Two video sequences are tested to verify the effectiveness of our method. GMM [1], KDE [4] and LBP [6] methods are employed to compared with the proposed method. Visual comparison and numerical method in terms of Detection rate (DR) and false alarm rate (FAR) are used to evaluate the performance. (TP is the number of foreground pixels correctly detected, FP is the number of background pixels detected as foreground and FN is the number of foreground pixels not detected). Ground truth images are obtained by manually segmented.

$$DR = \frac{TP}{TP + FN} \tag{12}$$

$$FAR = \frac{FP}{TP + FP} \tag{13}$$

The first sequence involving heavily waving trees is from [11]. Examples of detection results are shown in Fig.3. The GMM and KDE methods can not adapt to rapid changing environment, whereas LBP and SCS-LBP methods are robust to such scenes. Table.1 show the corresponding quantitative results.

Fig.4 shows a qualitative comparison of our method with the other methods in dynamic background provided by rippling water. It can be seen that a large number of pixels are misclassified by the GMM and KDE methods. The LBP method eliminates much noise, but cannot give accurate detection. Instead, our proposed approach effectively suppresses those dynamic background pixels and accurately detect the foreground object. Since the target in the sequence is relatively small, the improvement effect is obvious by comparing our approach and LBP method. Quantitative results shown in Table.2 also prove the effectiveness of our method.

Fig. 3. Comparison results on waving trees. The top row shows the original 247^{th}, 249^{th}, 252^{th} and 256^{th} frames. The second shows are corresponding ground truth frames. The third, forth and fifth rows are the results obtained by GMM, KDE and LBP respectively. The last row shows the results obtained by our approach. Note: Morphological operating are not used in the results.

It can been seen that our method and LBP method outperform other two methods in the experiments. Besides, our method has low complexity compared with the LBP method. For the two experiments, these parameter values $\alpha = 0.01$, $T_B = 0.8$, $R = 2$, $R_{region} = 8$, $K = 4$ which are selected according to [6] are the same for both methods. We set $N = 7$ for LBP method and $\beta = 0.005$, $N = 8$ for our method. The histogram bins of each pixel are $2^7 = 128$ for LBP method, whereas our method only produces $2^{(8/2)+1} = 32$ bins which are much shorter than LBP method. Although temporal information extraction adds the computational cost, the total computational time per frame by our method is shorter than LBP method. Table.3 shows the speed comparison of both methods on two sequences. As an example, all algorithms are not optimized and implemented with Matlab7.1 on the computer of 2.4GHz Intel Core 2 Duo, 2GB RAM. The frame size of both sequences is 160 × 120.

5. CONCLUSION

In this paper, we propose a novel spatial-temporal operator named SCS-LBP for background modeling. It can extract spatial texture and motion information simultaneously with low

Table 1. Quantitative comparison of DR and FAR on waving trees sequence

Method		GMM	KDE	LBP	SCS-LBP
DR(%)	247_{th}	69.69	77.58	93.94	93.84
	249_{th}	65.23	79.08	93.64	95.37
	252_{th}	66.16	79.14	95.16	96.70
	256_{th}	63.58	69.40	87.47	91.69
FAR(%)	247_{th}	25.39	33.71	9.35	13.10
	249_{th}	26.49	33.25	8.40	11.23
	252_{th}	23.98	33.42	11.22	12.99
	256_{th}	30.04	40.54	6.51	8.62

Table 2. Quantitative comparison of DR and FAR on rippling water sequence

Method		GMM	KDE	LBP	SCS-LBP
DR(%)	182_{th}	55.96	83.03	54.34	92.32
	186_{th}	56.63	81.93	47.39	86.14
	202_{th}	49.12	80.62	49.34	80.18
	216_{th}	57.69	74.70	84.21	91.70
FAR(%)	182_{th}	63.21	71.54	57.17	16.61
	186_{th}	67.55	77.83	56.38	10.25
	202_{th}	50.77	66.51	49.55	20.35
	216_{th}	66.27	74.15	48.51	17.93

computational cost. Combing this operator with an improved temporal information estimation method, we propose a new background modeling approach. Experiments on challenging videos demonstrate the effectiveness and robustness of our method in dynamic scenes. Since our method contains many parameters, our future work will focus on efficient parameter setting according to different situation.

6. ACKNOWLEDGEMENT

The work was supported in part under National Basic Research Program of China (2010CB731401 and 2010CB731406), 863 National High-Tech Program (2008AA01A318) and National Natural Science Foundation of China (60702044 and 60625103).

Table 3. Speed comparison of between LBP and SCS-LBP methods(in seconds)

Method	Average running time per frame	
	Waving trees	Ripple water
LBP	7.28	7.32
SCS-LBP	5.90	5.78
Time reduction	18.96%	21.04%

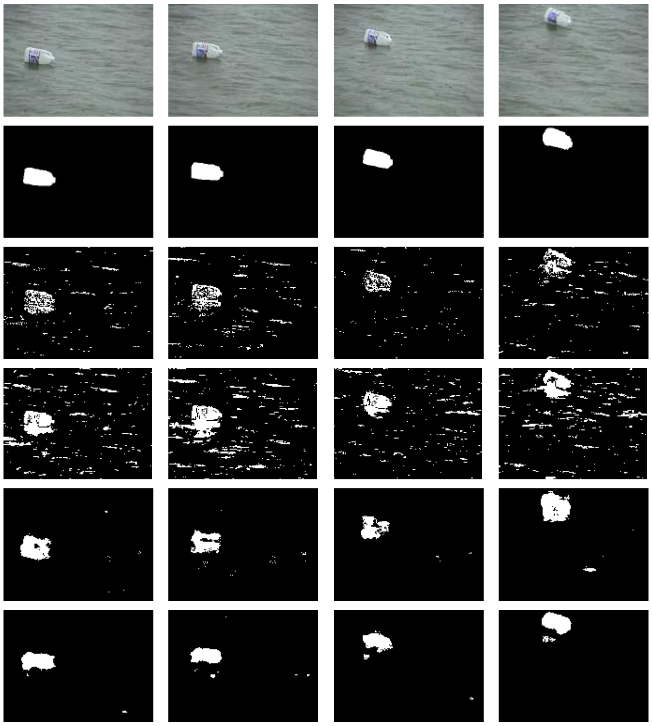

Fig. 4. Comparison results on rippling water. The top row is the original frames named as 182^{th}, 186^{th}, 202^{th} and 216^{th} frames. The second row is corresponding ground truth frames. The third, forth and fifth rows are results obtained by GMM, KDE and LBP respectively. The last row is the results obtained by proposed method. Note: Morphological operating were not used in the results.

7. 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.

[2] 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.

[3] 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, May 2005.

[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] 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.

[7] Shengping Zhang, Hongxun Yao, and Shaohui Liu, "Dynamic background modeling and subtraction using spatial-temporal local binary patterns," *Proc. 15th International Conference on Image Processing*, pp. 1156–1159, 2008.

[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] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[10] Pedro Gil-JimGénez, Hilario Gómez-Moreno, Javier Acevedo-Rodríguez, and Saturnino Maldonado Bascón, "Continuous variance estimation in video surveillances equences with high illuminationc hanges," *SignalProcessing*, vol. 89, no. 7, pp. 1412–1416, 2009.

[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.