FAST AND EFFECTIVE BACKGROUND SUBTRACTION BASED ON εLBP

LingFeng Wang, ChunHong Pan

National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
{lfWang@nlpr.ia.ac.cn, chpan@nlpr.ia.ac.cn}

ABSTRACT

Background subtraction is an important component of many computer vision systems. In this paper, we present a fast and efficient texture-based method for background subtraction in a video sequence. At first, a novel **LBP** (Local Binary Pattern) operator called ε**LBP** is applied to extract the local description of each pixel. Then, the kernel of background subtraction is performed based on two principle phases, i.e. the **background building** phase and the **foreground detection** phase. In the former phase, the probability that a pixel belongs to foreground is calculated based on the difference between the current ε**LBP** and the adaptive mean ε**LBP**. In the later phase, a thresholding operator is applied on the probability to determine whether a pixel can be classified as foreground, and the adaptive mean ε**LBP** is updated by a user-settable learning rate. Finally, our approach is evaluated against several video sequences compared with the traditional **MoG** model. Experiments show that our method is suitable for various scenes and is appealing with respect to robustness.

Index Terms— Background Subtraction,ε**LBP**

1. INTRODUCTION

Background subtraction is a convenient and effective method for detecting foreground objects from a stationary camera. Its mainly depends on the background modeling module. The central idea behind this module is to utilize the visual properties of the scene for building an appropriate representation, that can then be used to classify a new observation as foreground or background.

Existing methods for background modeling can be classified as **predictive** and **statistical**. The **predictive** methods model the scene as a time series and develop a dynamical model to recover the current input based on past observations [1, 2], while the **statistical** methods neglect the order of the input observations and roughly build a probabilistic representation of the observations [3, 4, 5, 6, 7]. A popular **statistical** method is to model each background pixel with a single Gaussian distribution [3]. However, This method does not work well in the case of dynamic natural environments including repetitive motions, i.e. waving vegetation, rippling water, and camera jitter. In [4], the mixture of Gaussians(**MoG**) approach is proposed to solve these complex, non-static backgrounds. Unfortunately, background with fast variations can not be accurately modeled by just a few Gaussians. To overcome the limitations of parametric methods (i.e. single Gaussian in [3], **MoG** in [4]), a nonparametric technique is developed in [5]. This utilizes a general nonparametric kernel density estimation technique for building a statistical representation of the scene background. However, both the parametric methods and nonparametric method may fail when foreground objects have similar color to background, or even when the illumination variations occur due to sunlight changing outdoor and light switching indoor. The main reason is that, these methods only use the pixel color or intensity information to detect foreground objects. To deal with this weak description, [6, 7] use a novel and powerful approach based on discriminative texture features represented by **LBP** [8, 9] to capture background statistics. In [6], each pixel is modeled as a group of adaptive **LBP** histograms that are calculated over a circular region around the pixel, while in [7], each pixel is modeled as a mixture of **LBP** mode by combining the hysteresis update step and the bilateral filter together. The main limitation of those two methods is that both memories and computation costs increase greatly with the increasing of the images resolution.

In this paper, we propose a fast background subtraction method based on the novel **LBP** called ε**LBP** for detecting moving objects. Our method is able to overcome two drawbacks brought by the traditional **LBP** operator, i.e. the neighboring pixels are conditional independent under the center pixel, and it is weakly to measure the difference between the center pixel and its neighborhood. The overview of our method is illustrated in Fig.2. Compared with the **LBP** approaches, our method improves greatly memories and computation efficiency by a simple measurement, which is linearly proportional to the images resolution. And compared with the **MoG** model, the illumination variations and the color similarity between foreground and background are eliminated in some degree as discriminative texture feature ε**LBP** is used in our approach.

The rest of this paper is organized as follows. A brief introduction on the novel texture description ε**LBP** compared with the **LBP** is given in Section 2; Background subtraction method, which is composed by background modeling and foreground detection, is described in Section 3 in detail; experiment results on real data compared with the **MoG** are reported in Section 4; the conclusive remark is addressed at the end of this paper.

2. FROM LBP TO εLBP

LBP [8, 9] is a robustly gray-scale invariant texture primitive statistic. The original version of the operator labels the pixels of an image by thresholding the value of center pixel with the $\mathbf{3} \times \mathbf{3}$ neighborhood and describing the result as a binary number (binary pattern):

$$\mathbf{LBP}_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \quad s(t) = \begin{cases} 1 & t \geqslant 0 \\ 0 & t < 0 \end{cases} \qquad (1)$$

where g_c corresponds to the gray value of the center pixel (x_c, y_c), and g_p to the gray values of P equally spaced pixels on a circle of radius R. The **LBP** operator describes the pixels of an image only by the signs of the differences in a neighborhood, which are interpreted as a P-bit binary number.

This research is sponsored by the National '863' High-Tech Program of China under the grant No. 2009AA012104 and Natural Science Foundation of China under the grant Nos: 60675012, 60873161.

The advantage of the **LBP** operator is its tolerance against illumination changes and its computational simplicity. However, it has two main drawbacks as follows: 1. the neighboring pixels are conditional independent under the center pixel. In fact, the assumption that the neighboring pixels are conditional independent under the center pixel is too strong to give the local description of the center pixel. 2. the measurement between the center pixel and its neighborhood only takes the signs of the differences into consideration. Obviously, this kind of measurement may lead to some wrong results. For example, when all the gray values (the center pixel and the corresponding neighboring pixels) increase or decrease a certain equivalent amount, the result of the original **LBP** does not change. Thus, only using the signs of the differences loses a lot of usable information. In order to overcome two drawbacks, we propose a new ε**LBP** operator as follows, and its comparison with **LBP** is illustrated in Fig.1.

$$\varepsilon\mathbf{LBP}_{P,R}(x_c, y_c) = \sum_{p=1}^{P} s(\frac{\hat{g}_p - \check{g}_p}{g_c} - \varepsilon)2^p \tag{2}$$

where $\hat{g}_p$ and $\check{g}_p$ correspond to the gray value of the clockwise and counter-clockwise neighborhood of g_p. The parameter ε is a noise parameter which makes the **LBP** more stable against noise. The advantages of ε**LBP** compared to **LBP** are from two aspects. Using the difference of $\hat{g}_p$ and $\check{g}_p$ in ε**LBP** operator has an intuitive motivation in that it explodes the conditional independent of neighborhood. While using the ratio of $\hat{g}_p - \check{g}_p$ and g_c in ε**LBP** operator strengthens the measurement the center pixel and its neighborhood.

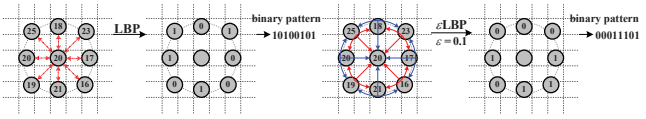

Fig. 1. Overview of **LBP** and ε**LBP** operators for a neighborhood of **8** pixels.

3. BACKGROUND SUBTRACTION BASED ON εLBP

Background subtraction algorithm can be divided into two basic phases: **background modeling** and **foreground detection**. The flow chart of the novel background subtraction method is summarized in Fig.2. A gray value input image is first analyzed by performing ε**LBP** operator. Then, the probability that a pixel belongs to foreground is calculated on each pixel (in the **background modeling** phase). Finally, foreground mask is gained by a thresholding operator and the adaptive mean ε**LBP** is updated (in the **foreground detection** phase) by a user-settable learning rate.

3.1. Background Modeling

In order to eliminate the illumination variations and the color similarity between foreground and background, we choose to utilize texture information when modeling the background and measure the texture by the ε**LBP**. Background modeling consists of two basic steps: ε**LBP** calculating and probability calculating.

The first step of background modeling is ε**LBP** calculating. Compared with the local description of the pixel in [6], which uses the **LBP** histogram computed over a circular region of radius around the center pixel, we only calculate the ε**LBP** of the center pixel

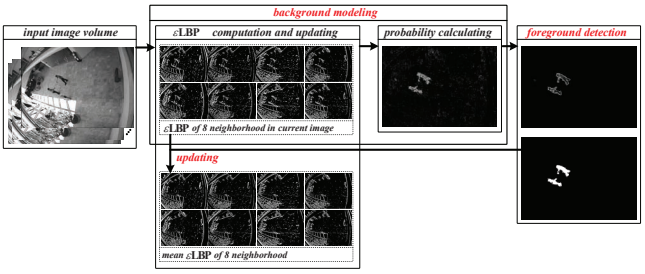

Fig. 2. Overview of the novel background subtraction algorithm.

for the purpose of improving the performing speed. Since all the neighboring pixels around the center pixel are equivalent in ε**LBP** calculating, considering them separately is better than using a single number (the original **LBP** representation) calculated from Eqn.2. In other words, we calculate

$$\pi_p = s(\frac{\hat{g}_p - \check{g}_p}{g_c} - \varepsilon) \qquad p \in \{1, 2, ..., P\} \tag{3}$$

for each neighboring pixel g_p of the center pixel g_c, and then the current ε**LBP** is represented by $\boldsymbol{\pi} = \{\pi_1, \pi_2, ..., \pi_P\}$ instead of Eqn.2. Therefore, each neighboring pixel g_p has their own binary pattern π_p. After performing the ε**LBP** operator, **8** binary images of different neighborhood can be obtained when using the **3** $\times$ **3** neighborhood (Fig.2 illustrates this in detail).

The background model presented in our approach follows the single Gaussian model proposed in [3], which attempts to model each pixel with two Gaussian parameters the mean and the variance. However, different from the single Gaussian model [3], only the mean (called adaptive mean ε**LBP**) is maintained, and the variance is ignored. Since all the neighboring pixels are consider separately, so the adaptive mean ε**LBP** is denoted as $\boldsymbol{\mu} = \{\mu_1, \mu_2, ..., \mu_P\}$, where each μ_p is the mean of the π_p with a series of frames.

The second step of background modeling is probability calculating, which is to calculate the probabilistic that a pixel belongs to the foreground. We define that a pixel belongs to the foreground as foreground probabilistic φ. As all neighboring pixels are considered separately, we can calculate each of them first and then sum up them as the foreground probabilistic φ. Let us denote that the probabilistic that all neighboring pixels belong to foreground as $\boldsymbol{\omega} = \{\omega_1, \omega_2, ..., \omega_P\}$, where each ω_p is the probability that a neighbor g_p belongs to foreground. Because the foreground probability ω_p describes the difference between the current observation (current ε**LBP**) and the model (adaptive mean ε**LBP**), ω_p can be calculated as follows,

$$\begin{aligned} \omega_p &= \frac{\mid \pi_p - \mu_p \mid}{\max\{\mathscr{H}(\mu_p), \delta\}} \\ \mathscr{H}(\mu_p) &= \mu_p \log \mu_p + (1 - \mu_p) \log (1 - \mu_p) \end{aligned} \tag{4}$$

where the parameter δ (we set $\delta = 0.1$) is small positive number that prevents dividing by zero and $\mathscr{H}(\mu_p)$ is defined as entropy of the mean ε**LBP** of the neighboring pixel g_p. The definition of Eqn.4 comes from two folds. First, foreground probability is determined by the different between the current observation and the model (represented in numerator). It means that the foreground probability is liner to the difference. Second, the foreground probability is also determined the certainty degree of the current observation (represented in denominator). We utilize of the entropy of the mean ε**LBP** to describe the certainty degree of the observation. As mentioned above,

the foreground probabilistic φ can be calculated by the sum of ω_p, given by

$$\varphi = \sum_{p=1}^{P} \omega_p \tag{5}$$

As shown in Eqn.3, Eqn.4, and Eqn.5, the computation complexity is quite low as they require simple operations only. The complexity is linear to the number of neighborhood $\mathbf{O}(P)$.

3.2. Foreground Detection

Foreground mask is calculated before updating of the background model (the adaptive mean ε**LBP**). At the foreground mask calculating procedure, a thresholding operator is applied to determine whether a pixel can be classified as foreground, given by

$$m = \begin{cases} 1 & \varphi > T_B \\ 0 & \textbf{else} \end{cases} \tag{6}$$

where T_B is a user-settable threshold. In our experiment, we set $T_B = 10$ when using the $\mathbf{3} \times \mathbf{3}$ neighborhood. The processing of updating (the adaptive mean ε**LBP**) is required after obtaining foreground mask. Following the update approach proposed in [10], which is called selective background update, our ε**LBP** is updated by the new observation,

$$\boldsymbol{\mu} = m\boldsymbol{\mu} + (1-m)\left(\alpha\boldsymbol{\mu} + (1-\alpha)\boldsymbol{\pi}\right) \tag{7}$$

where α is a user-settable learning rate and m is the foreground mask. As illustrated in Eqn.7, if the mask $m = 1$, that is the pixel belongs to foreground, the adaptive mean ε**LBP** $\boldsymbol{\mu}$ is not updated, while if the mask $m = 0$, that is the pixel belongs to background, the adaptive mean ε**LBP** $\boldsymbol{\mu}$ is updated with the learning rate α. The adaptation speed of the background model is controlled by the learning rate parameter α. The bigger the learning rate, the faster the adaptation.

4. EXPERIMENT RESULTS

Three experiments are performed to compare our method with the **MoG** method. In these experiments, the neighborhood size is selected by $\mathbf{3} \times \mathbf{3}$, and the learning rate α is evaluated with $\alpha = 0.1$.

First Experiment: Both numerical and visual methods are used to evaluate our approach compared with the **MoG** model. In this sequence, there is obviously shadow besides the walking people. Fig.3 shows the result of *false negatives* (**FN**), which is the number of foreground pixels that were missed, and *false positives* (**FP**), which is the number of background pixels that were marked as foreground, by the numerical method. As illustrated in Fig.3, the **FN** of our approach is more stable than the **MoG** model, and the **FP** of our approach is much more smaller than the **MoG** model. Fig.4 shows the result of foreground detection result compared with the ground truth by the visual method. As illustrated in Fig.4, the correct foreground detection marked with salmon pink color accounted for the main part of the foreground mask.

Second Experiment: It consists of seven video sequences, each of them represents a different, potentially problematic scenario for background subtraction. We compared our approach with the ground truth and the results are illustrated in Fig.5. Although the illumination variations occur in **Light Switch** and **Time of Day** scenes, and the obvious shadow exists in **Bootstrapping** scene, all the foregrounds can be correctly detected by our approach. The main reason

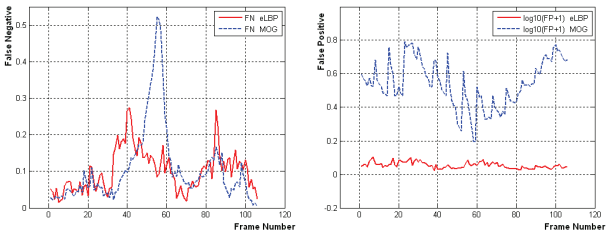

Fig. 3. Comparison results of our method with the **MoG** method. The left and right figures are illustrate the number of (**FN**) and (**FP**) for the two methods, respectively. Note that, in the right figure, the original result data $\mathbf{z}$ are replaced by $\lg(\mathbf{z}+1)$ for the purpose of illustrating the comparison obviously.

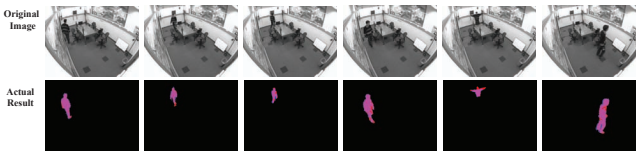

Fig. 4. Background subtraction results of our method compared with the ground truth. The salmon pink color points mark the correct results compared with the ground truth. The red color points mark the **FN**, while the blue color points mark the **FP**.

is that the discriminative texture features represented by ε**LBP** eliminate the effect of illumination variations.

Third Experiment: We compared the performance of our approach to the **MoG** model based on visual expression. As illustrated in Fig.6, our approach is less sensitive on shadow than the **MoG** model, especially when the shadow is strong. Fig.6 gives the comparison results.

We also measured the computational speed of the proposed method. We use a standard PC with a 1.8 GHz processor and 2048 MB of memory, and choose the program language with **Matlab** in our experiments. As illustrated in Fig.7, the process times on middle resolution $\mathbf{360} \times \mathbf{240}$ approximate to **0.0625** (this resolution is frequently used in surveillance system), that is **16** frames per a second. This makes our approach well-suited to systems that require real-time processing, i.e. tracking, detection in surveillance.

5. CONCLUSION

A fast and effective texture based background subtraction method is proposed in this paper. The contributions in this paper are two-fold. First, we improve the original **LBP** operator. The ε**LBP** method are quite fast to compute, which is an important property from the practical implementation point of view, and less sensitive to the illumination variations or the color similarity between foreground and background. Second, applied the ε**LBP** operator in background subtraction. The novel background subtraction method provides us with many advantages and improvements on the the limitation of memories and computation costs in the **LBP** approaches and the sensitive to illumination variations or the color similarity between foreground and background in the traditional **MoG** model. Although there are more advantages in this approach, a disadvantage is that the background image can not be gained, which is the common disadvantage of the texture based method. Fortunately, in many computer vision

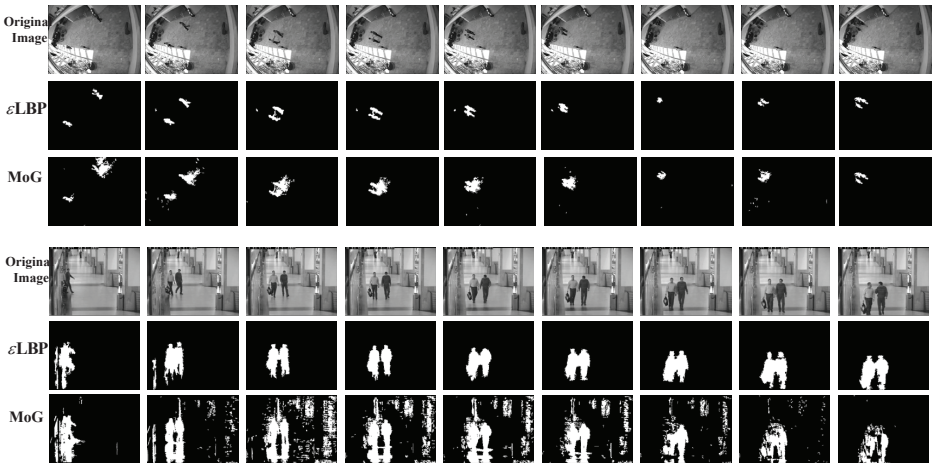

Fig. 6. Background subtraction results of our method compared with **MoG** for the test sequences. The second and third rows contain the corresponding processed frames by the our method and **MoG** method respectively. The images resolution are **320** $\times$ **240** pixels.

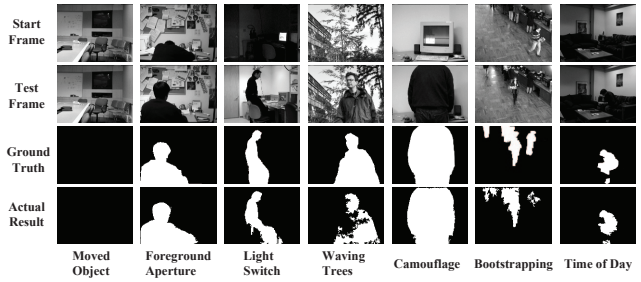

Fig. 5. Background subtraction results of our method for the test sequences. The images resolution are **160** $\times$ **120** pixels.

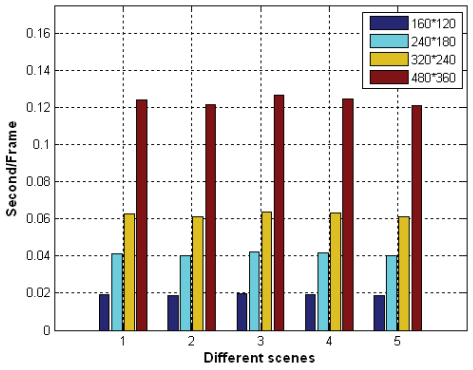

Fig. 7. Four different resolution (**160** $\times$ **120**, **240** $\times$ **180**, **320** $\times$ **240**, and **480** $\times$ **360**) are selected. Meanwhile, for each of the resolution, **5** different scenes are selected.

tasks, i.e. detection and tracking, only the foreground is enough.

6. REFERENCES

[1] Dieter Koller, Joseph Weber, and Jitendra Malik, "Robust multiple car tracking with occlusion reasoning," in *European Conference on Computer Vision(1)*, 1994, pp. 189–196.

[2] Nuria M. Oliver, Barbara Rosario, and Alex Pentland, "A bayesian computer vision system for modeling human interactions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 831–843, 2000.

[3] T.J. Darrell C.R. Wren, A. Azarbayejani and A.P. Pentland, "Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[4] W. Grimson C. Stauffer, "Adaptive background mixture models for real-time tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, vol. 2, pp. 246–252.

[5] D. Harwood A. Elgammal and L.S. Davis, "Nonparametric model for background subtraction," in *European Conference on Computer Vision*, 2000, pp. 751–767.

[6] M. Heikkila and M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 657–662, 2006.

[7] J.-M Jian Yao Odobez, "Multi-layer background subtraction based on color and texture," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[8] M. Pietikaainen T. Ojala 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.

[9] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[10] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russell, "Towards robust automatic traffic scene analysis in real-time," in *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, 1994, vol. 1, pp. 126–131.