

## FAST MOVING OBJECTS DETECTION USING iLBP BACKGROUND MODEL

B. Vishnyakov, V. Gorbatshevich, S. Sidiyakin, Y. Vizilter, I. Malin, A. Egorov

State Research Institute of Aviation Systems (FGUP GosNIIAS), Moscow, Russia –  
 (vishnyakov, gvs, sersid, viz, imalin, aegorov)@gosniias.ru

### Commission III

**KEY WORDS:** background model, LBP descriptor, iLBP descriptor, regression model, video surveillance, object detection.

#### ABSTRACT:

In this paper a new approach for moving objects detection in video surveillance systems is proposed. It is based on iLBP (intensity local binary patterns) descriptor that combines the classic LBP (local binary patterns) and the multiple regressive pseudospectra model. The iLBP descriptor itself is considered together with computational algorithm that is based on the sign image representation. We show that motion analysis methods based on iLBP allow uniformly detecting objects that move with different speed or even stop for a short while along with unattended objects. We also show that proposed model is comparable to the most popular modern background models, but is significantly faster.

### 1. INTRODUCTION

One of the key problems in the intellectual video surveillance systems is fast detection of moving objects. It is usually solved by building a background model of the scene and getting a difference between the built model and the current frame or a group of last frames.

The most common background models are GMM (Gaussian Mixture Model) (Stauffer, 1999) and KDE (Kernel Density Estimator) (Elgammal, 2000). A fast model based on multiple regressive pseudospectra was also proposed (Vishnyakov, 2012). But all these methods, having their own advantages, also have a serious common drawback: they accumulate the information about luminance distribution over a finite period of time in every pixel independently and the relations between neighboring pixels are not taken into account. In recent time, there appeared some papers where authors considered every pixel with a descriptor of its surroundings (Heikkila, 2004; Heikkila, 2006). For example, the textured-based method (Heikkila, 2004) modelled the background with a group of histograms based on local binary patterns. Using LBP histograms helps to avoid labelling some moving background pixels as foreground since it extracts region texture features. However, its detection performance will sharply decline when scenes have strong changes. As further improvement of Heikkila approach, dynamic background modelling and subtraction based on spatio-temporal local binary patterns were introduced in (Zhang, 2008) and modeling pixel process with scale invariant local patterns were introduced in (Liao, 2010) to handle different illumination variations. Such approaches substantially increase quality characteristics of both background models and video analysis algorithms. However, it appears to be almost unacceptable due to its computational complexity, when a real-time processing is needed for multi-camera (not a single camera) setup.

In this work, we propose a new iLBP (intensity local binary patterns) descriptor and build a fast background model on its basis. Together with regressive estimate of the value of an individual pixel, we use a statistical estimate of the LBP descriptor components. This approach allows stabilizing the value of the descriptor and constructing a background model that is robust to lighting conditions changes in the scene and is applicable for the real time multi-camera setup.

### 2. iLBP DESCRIPTOR AND BACKGROUND MODEL

#### 2.1 LBP descriptor

Let us consider the classic LBP descriptor (Ojala, 1994, 1996). Let  $I(x, y)$  be intensity of the current pixel  $(x, y)$ . LBP operator  $LBP(x, y)$  labels pixels of an image by thresholding eight neighborhood of each pixel with the center value:

$$C_{LBP}(x + dx_i, y + dy_i) = \begin{cases} 1: I(x, y) - I(x + dx_i, y + dy_i) \geq 0 \\ 0: I(x, y) - I(x + dx_i, y + dy_i) < 0 \end{cases} \quad (1)$$

and considering the result (Fig. 1) as an eight bit binary number - local binary pattern code:

$$LBP(x, y) = \sum_{i=1}^8 2^{i-1} \cdot C_{LBP}(x + dx_i, y + dy_i), \quad (2)$$

where  $dx_i, dy_i, i = 1..8$  -  $x$ - and  $y$ - coordinate shifts with respect to center pixel  $(x, y)$ .

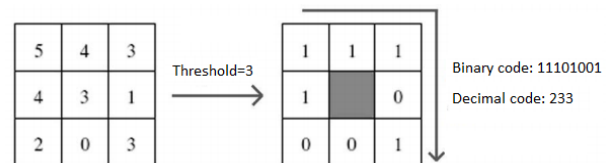


Figure 1. Usual LBP computation.

Comparison of LBP descriptors as binary vectors is calculated using the Hamming distance  $H(\cdot)$ :

$$D_{LBP}(a, b) = H(LBP(a), LBP(b)) \quad (3)$$

Threshold  $tr$  is often used when comparing the pixels to reduce noise influence.

$$C_{LBP}(x + dx, y + dx, tr) = \begin{cases} 1: I(x, y) - I(x + dx, y + dy) \geq tr \\ 0: I(x, y) - I(x + dx, y + dy) < tr \end{cases} \quad (4)$$

## 2.2 iLBP descriptor

However, LBP descriptors with all its advantages have some significant drawbacks. The main drawback is a complete ignoring of intensity information when comparing LBP descriptors. Because of this, there could be a paradoxical situation (wrong pixel comparison result) when intensity values of pixels differ drastically, but their LBP descriptors are identical. On the other hand, it is obvious that within a chosen scene the fact of a local intensity change in the point of interest is very important. To overcome this drawback, we define  $iLBP(x, y)$  descriptor as a collection of  $LBP(x, y)$  descriptor values and intensity  $I(x, y)$  values of the image:

$$iLBP(x, y) = \{LBP(x, y), I(x, y)\} \quad (5)$$

The definition of iLBP descriptor lead us to the following formula for the distance  $D_{iLBP}$  between the iLBP descriptors in the image points  $a$  and  $b$ :

$$D_{iLBP}(a, b) = k |I(a) - I(b)| + H(LBP(a), LBP(b)) \quad (6)$$

where  $k$  is a proportionality factor,  $H(\cdot)$  - Hamming distance between two LBP descriptors.

The proportionality factor  $k$  can be chosen in the range of  $2^{-8}$  to  $2^{-4}$  to match the possible values of the Hamming distance.

## 2.3 iLBP background model

We propose an approach for motion detection using a dynamically changing background model. In contrast to classic methods we consider an image not as a function of intensity, but as a set of iLBP descriptors computed for every point of the image.

Every descriptor contains intensity value and a binary vector, and this dual nature is a problem. The background model can also be considered as containing two independent models, first one corresponding to the binary part (iLBP) and the other to the image intensity part. These models are united at the last stage of processing for motion detection and segmentation of moving regions.

It's convenient to use the regressive model for the "image" part, which has already proved itself very fast and reliable in solving motion detection problems (Vishnyakov, 2012). For the "binary" part we propose using a simple statistical model.

Let us consider  $N$  consequent frames of the video. Let  $T$  be the current frame.

For the image part let us consider accumulator of the regression pseudospectra (Vishnyakov, 2012):

$$M_N(x, y, T) = \alpha \cdot M_N(x, y, T - 1) + (1 - \alpha) \cdot I(x, y, T), \quad (7)$$

where  $I(x, y, T)$  is a  $(x, y)$  pixel intensity at the frame  $T$ ,  $M_N(x, y, T)$  - the value of the  $N$ -frame accumulator in a pixel  $(x, y)$ ,  $N$ -frame means all frames from  $(T - N)$  frame up to  $T$  frame,  $\alpha \in [0, 1]$  - regression parameter.

For the binary part of the scene, point  $(x, y)$  is corresponded to a sequence of LBP descriptors on the considered frames:

$$LBP_N(x, y, T) = \{LBP(x, y, T), LBP(x, y, T - 1), \dots, LBP(x, y, T - N)\} \quad (8)$$

Without intensity components of the descriptors this sequence can be considered as an implementation of stationary random processes  $n_1(T), \dots, n_8(T)$ ,  $n_i(T) = LBP^i(x, y, T)$ . Here  $LBP^i(x, y, T)$  is the bit  $i$  of the LBP descriptor computed for the

point  $(x, y)$  on the frame  $T$ . Every random process corresponds to a pixel in the central  $(x, y)$  pixel's neighbourhood and to a bit in the LBP code (Fig. 2).

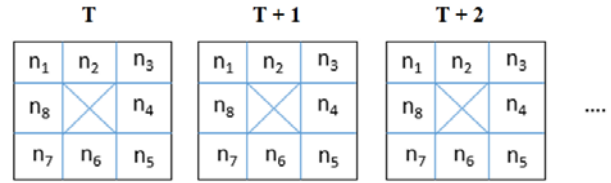


Figure 2. Random processes in relation to the central pixel for several consecutive frames.

Then, for a binary part of a background model, we can use quantile  $\varphi_\alpha^i(LBP_N(x, y, T))$ ,  $i = 1 \dots 8$  of the trajectories of the random processes  $n_1(T), \dots, n_8(T)$ .

$$\varphi_\alpha^i(LBP_N(x, y, T)) = \begin{cases} 1: \sum_{j=T-N}^T LBP^i(x, y, j) \geq N \cdot \alpha \\ 0: \text{otherwise} \end{cases}, \quad (9)$$

where  $N \cdot \alpha$  - quantile level,  $i = 1, \dots, 8$  - bit number.

Thus, the final binary part of the descriptor is put together like (2):

$$\varphi_\alpha(LBP_N(x, y, T)) = \sum_{i=1}^8 2^{i-1} \cdot \varphi_\alpha^i(LBP_N(x, y, T)). \quad (10)$$

For the moving objects detection we propose to use a similar to (6) sum of the differences of independent background models, calculated for various parameters:

$$c_{iLBP}(x, y) = k |M_{N_1}(x, y, T) - M_{N_2}(x, y, T)| + H(\varphi_\alpha(LBP_{N_1}(x, y, T)) - \varphi_\alpha(LBP_{N_2}(x, y, T))), \quad (11)$$

For the regular use,  $N_2$  parameter can be set to 1 frame. However, in general,  $N_2$  can be any number less than  $N_1$ . We recommend setting parameter  $N_2$  equal to  $N_1/2$  or  $N_1/4$  to improve noise filtration.

Main  $N_1$  parameter is a primary accumulator length and can be considered exactly like an accumulator in (Vishnyakov, 2012). If we set  $N_1$  equal to a relatively small number of frames (8...32), moving objects will be detected. If we set  $N_1$  equal to a relatively big number of frames (512...2048), unattended and carried away objects will be detected.

Therefore, this approach allows uniformly detecting objects that move with different speed or even stop along with unattended or carried away objects.

To identify if a  $(x, y)$  pixel belongs to the background or to the foreground we use a simple threshold:

$$(x, y) \in \begin{cases} \text{object} : c_{iLBP}(x, y) \geq C \\ \text{background} : c_{iLBP}(x, y) < C \end{cases} \quad (12)$$

This method of moving objects detection allows combining the strengths of both textural techniques of image comparison and intensities.

## 2.4 iLBP computation remarks

In the practical implementation of the approach described above, a number of significant challenges appear. The first obvious problem is to store a sequence of LBP.

The easiest way is to use 8 integer variables ( $P_1, \dots, P_8$ ), where each  $P_i$  corresponds to its bit descriptor:

$$P_i(\text{LBP}_N(x, y, T)) = \sum_{t=1}^N 2^{t-1} \text{LBP}_t^i(x, y) \quad (13)$$

Then the problem of computing quantiles reduces to the well known population counting problem. This task can be quickly calculated by modern processor commands, for example, `popcnt()`.

Another difficulty is the necessity of calculation of the LBP descriptors for each frame, wherein the most processor time in the calculation of the descriptor is used by paired comparisons.

If reconfigurable or customary processors are available this is not a problem (Boutellier et al., 2012), but for an ordinary PC hardware the nontrivial solution to this problem is a sign representation of the image. With this approach we consider the image as a plurality of paired comparisons (more is 1 and less is 0). In addition, symmetric pairwise comparisons are considered equivalent. Such a representation can be shown as a flat undirected graph (Fig. 3).

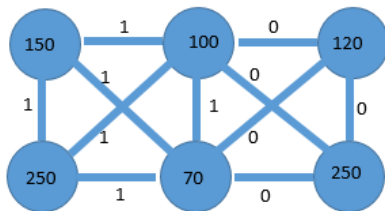


Figure 3. Presentation of image as an undirected graph.

In this graph, each node corresponds to a pixel of the image, and the weights of the edges correspond to the pairwise comparisons. Then the Hamming distance for the two binary parts of the iLBP descriptors is the number of mismatches of the weights  $w$  of the respective edges of the sign representation (accounting that the graph is undirected, you can use a simple rule  $w(a, b) = 1 - w(b, a)$ , where  $a$  and  $b$  are related nodes). In considered case, we do not need to take into account the orientation of the edges, because descriptors are compared in the same points on different frames.

Background model, described in the previous section, can be easily transformed for the sign representation of the image:

$$\varphi_{\alpha}^i(\text{LBP}_N(x, y, T)) = \begin{cases} 1: \sum_{t=1}^N w_t \left( \begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} x + dx_i \\ y + dy_i \end{pmatrix} \right) \geq N \cdot \alpha \\ 0: \text{otherwise} \end{cases} \quad (14)$$

Thus, all the operations described in the previous section are easily transferred to the case of the sign representation. However, for computing, storing and comparing of sign representations we require half as much processor and memory operations. In addition, this representation shows the relationship of this approach to the morphological image analysis, described in (Karkishchenko, 2010). This fact explains the high stability of the algorithm to changes in brightness and external noises.

### 3. TESTING RESULTS

In order to demonstrate the efficiency and effectiveness of the proposed approach we compared our experimental results with the well-known background modeling methods GMM (Stauffer, 1999), KDE (Elgammal, 2000), STLBP (Spatio-temporal local binary patterns) (Zhang, 2008) on the two video sequences ("highway" and "PETS 2006"). Particularity of the "highway" video is the large number of branches and moving

shadows cast by them. Significant challenge for the analysis of the video "PETS 2006" is the large number of moving objects in the background, partially fenced and having low contrast. Quantitative results of the methods of GMM, KDE and iLBP model are shown in Table 1, Table 2, image comparison – in Figure 4. Under false negatives we mean the number of background pixels that were not found. Under the false positives – the number of background pixels that have been considered as a moving object by the algorithm. Ground truth images, processed images, frames per second, true positive, false positive, false negative results for GMM, KDE, STLBP were taken from the video database CDNET (Goyette, 2012) for change detection. Note that all processed images from CDNET were median filtered. iLBP result images were not filtered to show what it is capable off. Achieved iLBP results allow suggesting that the proposed method is not generally inferior to GMM, KDE and STLBP, but greatly exceeds them in computation and gives reasonable moving objects masks in outdoor scenarios.

Frames	787 frame	1627 frame
Original		
Ground truth		
KDE		
GMM		
STLBP		
iLBP		

Figure 4. Detection of moving objects using iLBP background model. Line 1 – frame numbers of "highway" video, line 2 –

corresponding frames, line 3 – ground truth, line 4 – KDE, line 5 – GMM, line 6 – STLBP, line 7 – iLBP.

Method	TP	FP	FN	FPS
GMM	5011645	378316	446344	~42
KDE	5118955	368979	339034	~18
STLBP	5188688	56464	269301	~58
iLBP	4803545	449439	411211	<b>~920</b>

Table 1. Results for GMM, KDE, STLBP and iLBP for the “highway” video. TP - True Positive, FP - False Positive, FN - False Negative.

Method	TP	FP	FN	FPS
GMM	4232121	1154934	596069	~21
KDE	3816139	790623	1012051	~9
STLBP	3727875	838047	1100315	~29
iLBP	3956678	1004934	1200881	<b>~670</b>

Table 2. Results for GMM, KDE, STLBP and iLBP for the “PETS 2006” video. TP - True Positive, FP - False Positive, FN - False Negative.

#### 4. CONCLUSION

The problem of automatic video analysis for the detection and tracking of moving objects is the most significant problem in the field of motion analysis, applied to problems of video surveillance and security systems. In this paper, we proposed a new approach to the problem of background modeling and subtraction based on combination of image intensity and binary information in each pixel. For this purpose, we introduced a new descriptor iLBP and a fast method for iLBP evaluation on a sequence of images when reconfigurable or customary processors are not available and processing speed is crucial. The evaluation results of the proposed approach are given for the publicly available outdoor scenarios. Achieved speed is more than satisfactory while preserving reasonable loss in overall performance.

#### References

Stauffer C. and Grimson W. E. L., 1999. Adaptive background mixture models for real-time tracking. *In Proc. Int. Conf. on Computer Vision and Pattern Recognition*, Vol. 2, IEEE, Piscataway, NJ.

Elgammal A., Harwood D. and Davis, L., 2000. Non-parametric model for background subtraction. *In Proc. Eur. Conf. on Computer Vision, Lect. Notes Comput. Science*. 1843, pp. 751-767.

Vishnyakov B., Vizilter Y., Knyaz V., 2012. Spectrum-Based Object Detection And Tracking Technique For Digital Video Surveillance. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 579-583.

Ojala T., Pietikäinen M., and Harwood D., 1994. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. *Proceedings of the 12th IAPR International Conference on Pattern Recognition*. vol. 1, pp. 582 - 585.

Ojala T., Pietikäinen M., and Harwood D., 1996. A Comparative Study of Texture Measures with Classification Based on Feature Distributions. *Pattern Recognition*. vol. 29, pp. 51-59

Heikkila M., Pietikainen M., Heikkila J., 2004. A texture-based method for detecting moving objects. *Proc. British Machine Vision Conf.*, vol. 1, pp. 187–196.

Heikkila M., Pietikainen M., 2006. A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4): 657–662.

Zhang S., Yao H., Liu S., 2008. Dynamic background modeling and subtraction using spatio-temporal local binary patterns. *Proc. IEEE Int. Conf. Image Process.*, pp.1556 -1559.

Liao S., Zhao G., Kellokumpu V., Pietikainen M., Li S., 2010. Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. *In Proc. Int. Conf. on Computer Vision and Pattern Recognition*. pp. 1301-1306.

Boutellier, J., Lundbom, I., Janhunen, J., Ylimainen, J., Hannuksela J., 2012. Application-specific instruction processor for extracting local binary patterns. *Design and Architectures for Signal and Image Processing (DASIP)*, pp.1-8.

Karkishchenko A., Goncharov A., 2010. Stability investigation of the sign representation of images. *Automation and Remote Control*. Vol. 71, Issue 9, pp. 1793-1803.

Goyette N., Jodoin P.-M., Porikli F., Konrad J. and Ishwar P., 2012. Changedetection.net: a new change detection benchmark dataset. *In Proc. IEEE Workshop on Change Detection (CDW-2012) at CVPR-2012*. pp. 1-8.