Fast motion detection based on cross correlation

PancaMudjirahardjo, JooKooi Tan, Hyoungseop Kim, Seiji Ishikawa
Dept. of Control Engineering, Kyushu Institute of Technology,
1-1 Sensui-cho, Tobata-ku, Kitakyushu-shi, Fukuoka, 804-8550, Japan
E-mail: {panca,etheltan,ishikawa}@ss10.cntl.kyutech.ac.jp, kim@cntl.kyutech.ac.jp
tel: 093-884-3191

Abstract

We present a method of fast motion detection as an abnormal motion based on cross correlation. Since the camera view is not in perpendicular with motion direction, the velocity of motion is not uniform spatially. Instead of object detection directly, we separate an image into several blocks. We calculate the cross correlation of the pixel intensity series in these blocks between a current and a previous frame. The maximum correlation is achieved at certain delay. This delay shows a shift of a similar pattern between the current and the previous frame. To localize an abnormal motion, we employ a hierarchical block size. The performance of the proposed method is experimentally shown.

Keywords: Abnormal motion, cross correlation, delay, hierarchical block size.

1. Introduction

In recent years, abnormal motion detection has attracted great research attention in computer vision. Most current surveillance systems only provide *reactive* security by enabling the analysis of events after the event has already occurred — what is really needed by the security community is *proactive* security to help prevent future attacks.

Many approaches on video event analysis are based on the object trajectories extracted from video. Abnormal events can be detected through a prior learning of normal events or, without a learning process, by analyzing the trajectory result directly.

Jiang et al. [1] used spatial and temporal context and performed frequency-based analysis to detect anomalous video events. The normal observation is modeled by hidden Markov model (HMM). This research detected the anomalous car trajectory on the road from top view. Kiryati et al. [2] recognized an abnormal human behavior from high camera view. Before the detection phase, they included a training phase for normal condition. Baranwal et al. [3] detected an abnormal indoor motion in a static background environment. They trained various motions using radial basis functions networks (RFBN). Park et al. [4] used clustering of motion based on similarity measurement of a feature space. They detected an abnormal motion, especially in a different direction case, from high camera view.

In this paper, we propose a fast motion detection with a camera view not in perpendicular with motion direction, as an abnormal motion among walking motion. We capture a scene from a 2 meter height and more for outdoor scenes, as shown in **Fig. 1**. Due to camera view not in perpendicular with motion direction, motion velocityin the image is not uniform spatially. We need to extend the method in [5,6].

It requires no foreground segmentation, no motion recognition and no object detection. We analyze the object’s velocity through a certain delay where the maximum correlation occurred.

Fig. 1. Scenes for performed experiment

2. Overview of the Proposed Method

A fast motion detection needssome velocity data. In this paper, we provide velocity data from cross correlation of pixel intensity series in the blocks between a current and a previous frame. The maximum correlation is achieved at a certain delay. This delay shows a shift of a similar pattern between the current and the previous frame.

Cross correlation is a standard method of estimating the degree to which two series of data are correlated. Consider two series $x(i)$ and $y(i)$ where i=0,1,2...N-1. The cross correlation r_{xy} at delay d is defined as [7],

$$r_{xy} = \frac{\sum_i \left[(x(i) - \bar{x})(y(i-d) - \bar{y}) \right]}{\sqrt{\sum_i (x(i) - \bar{x})^2} \sqrt{\sum_i (y(i-d) - \bar{y})^2}} \tag{1}$$

where $\bar{x}$ and $\bar{y}$ are the means of the corresponding series.

The range of delay d and thus the length of the cross correlation series can be less than N, for example, to test correlation at short delays only. The denominator in the expression above serves to normalize the correlation coefficients so that $-1 \le r_{xy} \le 1$ holds; the bounds indicating maximum correlation and 0 indicating no correlation. A high negative correlation indicates a high correlation but of the inverse of one of the series.

We describe the proposed method for fast motion detection based on cross correlation as depicted in **Fig.2**.

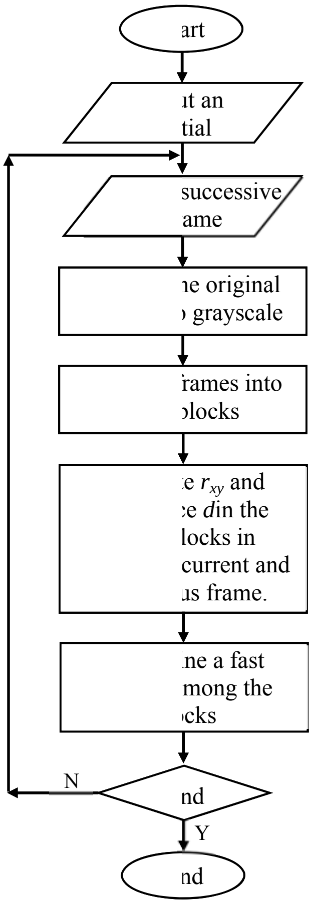

Fig. 2 Overview of the proposed method.

3. Method

In this section, we describe the proposed method in detail.

3.1. *Preprocessing*

First, we need to convert the original successive frames into grayscale images. Subsequently, we separate the images into $m \times n$ blocks. The calculation of cross correlation in the same block between a current and a previous frame gives the information of shift of an object in this block.

3.2. *Calculation of a shift magnitude based on cross correlation.*

As was mentioned in section 2, cross correlation is a standard method of estimating the degree to which two series are correlated [7]. This calculation is one dimensional computation.

As a simple example, consider two rectangular pulses shown in **Fig. 3**, in blue and green. The correlation series is shown in red.

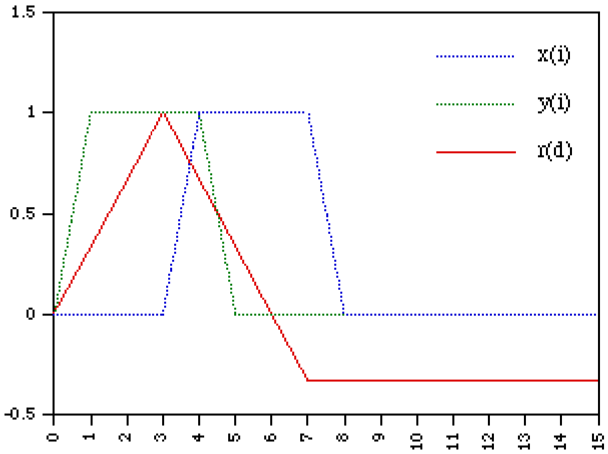

Fig.3. Example of cross correlation

The maximum correlation is achieved at a delay of 3. Considering the equations above, what is happening is that the second series is being slid past the first, at each shift the sum of the product of the newly lined up terms in the series is computed. This sum will be large when the shift (delay) is such that similar structure lines up.

Delay of the maximum correlation can be assumed as a shift of an object. This calculation of shift based on cross correlation is simpler than 2D correlation, because 2D correlation needs a template, then slides it over a specified range to get the maximum value. This calculation is also simpler than a method proposed in [6]. It is more accurate than optical flow based on Lucas-Kanade tracker which relies on feature points.

To provide one dimensional data series, we scan an image pixel as in **Fig. 4**.

To provide one dimensional data series as shown in **Fig. 4**, we perform the below algorithm, in which all the variables are in an integer type:

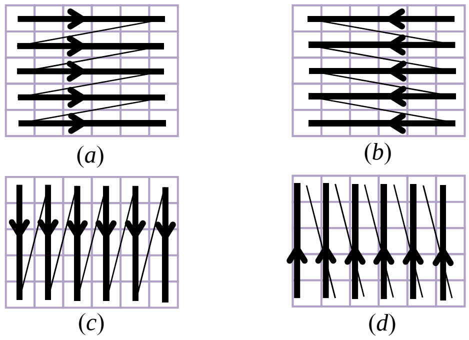

Fig. 4. Arrangement of pixel data to provide one dimensional data series.
(*a*) hor_1 (*b*) hor_2 (*c*) ver_1 (*d*) ver_2

for ($i = 0 : m \times n$)
  hor_1[i] = pixel[i/m][i-(i/m)×m];
  ver_1[i] = pixel[i **mod** n][i/n];
  hor_2[i] = pixel[i/m][(i/m+1)m-1-i];
  ver_2[i] = pixel[(i/n+1)n-1-i][i/m];
end for;

Here m and n are weight and height of an image block, respectively. **mod** is a modulo operation. Pixel[a][b] is a pixel intensity at a^{th} row and b^{th} column of the image block.

Then we calculate the cross correlation between current and previous image frame at the same image block. The maximum correlation is achieved at a certain delay. For example, the maximum correlation between hor_1(t) and hor_1(t-1) occurs at $delay_{\mathrm{hor\_1}}$.: We will get other delays, i.e. $delay_{\mathrm{hor\_2}}$, $delay_{\mathrm{ver\_1}}$, $delay_{\mathrm{ver\_2}}$.

$$delay_{hor\_\alpha} = \arg\max_{d}\left(r_{hor\_\alpha(t),hor\_\alpha(t-1)}\right) \tag{2a}$$

$$delay_{ver\_\alpha} = \arg\max_{d}\left(r_{ver\_\alpha(t),ver\_\alpha(t-1)}\right) \tag{2b}$$

Where $\alpha \in \{1,2\}$. Then the shift magnitude at a current image block is calculated as,

$$|shift| = \sum_{\alpha}\sqrt{delay^2_{hor\_\alpha} + delay^2_{ver\_\alpha}} \tag{3}$$

Let us define the identity of $|shift(i,j)|$, $I_{(i,j)}$, as below,

$$I_{(i,j)} = \begin{cases} 1 & \text{if } |shift(i,j)| > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Here, i and j are block's position at i-th row and j-th column, respectively.

3.3. *Determining a fast motion*

To detect a fast motion in a frame, we calculate the maximum of shift magnitude among the blocks within a frame. We do this search in hierarchical block sizes, as depicted in **Fig. 5**. First, block size $m\times n$ pixels, we will detect a fast motion at *block_fast*1. Second, block size $m\times h$ pixels, we will detect a fast motion at *block_fast*2. Finally, block size $2m\times h$ pixels, we will detect a fast motion at *block_fast*3, where h is image height.

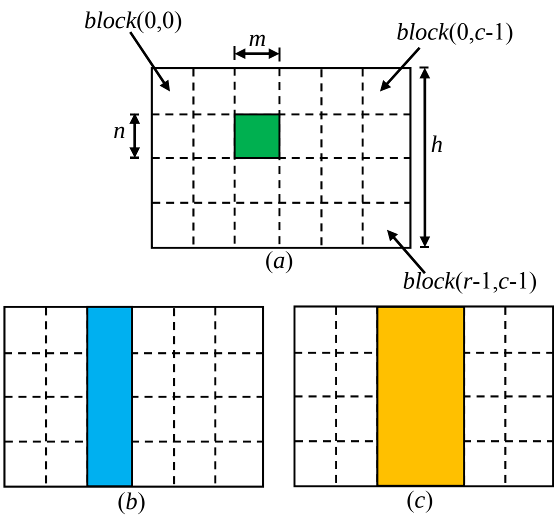

Fig. 5. Hierarchical block size.
(*a*) $m\times n$ (*b*) $m\times h$ (c) $2m\times h$

Operation in $m\times n$ *block size level*. For each block, we have a shift magnitude, $|shift(i,j)|$ defined by Eq. (3),where $i\in\{0,\dots,r-1\}$and $j\in\{0,\dots,c-1\}$. Then a*block_fast*1 is defined by,

$$block\_fast1 = \max(|shift(i,j)|) \tag{5}$$

We record *block_fast*1 position at r_1-th row and c_1-th column.

Operation in $m\times h$ *block size level*. For each column, we have an average of shift magnitude, avg_shift_j, where $j\in\{0,\dots,c-1\}$.

$$f(j) = \sum_{i=0}^{r-1} |shift(i,j)| \tag{6a}$$

$$r_a(j) = \sum_{i=0}^{r-1} I_{(i,j)} \tag{6b}$$

$$avg\_shift_j = \begin{cases} \dfrac{f(j)}{r_a(j)} & \text{if } f(j) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{6c}$$

Then*block_fast*2 is defined as,

$$block\_fast2 = \max(avg\_shift_j) \tag{7}$$

We get *block_fast*2 position at c_2-th column.

Operation in $2m\times h$ *block size level*. For each column, we have an average of shift magnitude, avg_shift_k, where $k\in\{0,\dots,c-2\}$.

$$f(k) = \sum_{j=k}^{k+1}\sum_{i=0}^{r-1} |shift(i,j)| \tag{8a}$$

$$r_b(k) = \sum_{j=k}^{k+1}\sum_{i=0}^{r-1} I_{(i,j)} \tag{8b}$$

$$avg\_shift_k = \begin{cases} \dfrac{f(k)}{r_b(k)} & \text{if } f(k) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{8c}$$

Then*block_fast*3 is defined as**,**

$$block\_fast3 = \max(avg\_shift_k) \tag{9}$$

We get *block_fast*3 position at c_3-th column.

Finally, we will detect a fast motion at c_F column, if$c_1 = c_2 = c_F$ or $c_1 = c_3 = c_F$ or $c_2 = c_3 = c_F$ or $c_1 = c_2 = c_3 = c_F$.

4. Experimental Result

The experimental environment is as follows: Operating system is Windows 7 ultimate; the processor is Intel® core™ i7 CPU 870 @2.93GHz and the used software is Microsoft Visual Studio 2010.

For experiment, we use outdoor scene, as **Fig. 1**, with many people do normal motion (walking) and a person does abnormal motion (running), with video frame rate and the size of a frame 30 fps and 320x180 pixels, respectively.

We set m and n are 20 pixels, so we separate an image frame into 16x9 blocks. For delay, d, we set 10. The experimental results are shown in **Fig. 6**.

Table 1shows the evaluation of performance, where TPR is true positive rate.

$$TPR = \frac{TP}{TP + FN} = \frac{TP}{GT} \times 100\% \tag{10}$$

Here :
TP : true positive, fast motion is detected as fast motion.
FN : false negative, fast motion is detected as normal motion.
GT : ground truth.

Table 1. Evaluation of performance

scene	TPR (%)	Execution time (ms)
1	88.7	228.75
2	91.5	

5. Conclusion

In this paper, we propose a method of fast motion detection in a crowd, with camera view not in perpendicular with motion direction. Instead of object detection and to avoid foreground segmentation, we calculate the cross correlation between the current and the previous frame in the same image block.

To detect a fast motion, we employ hierarchical block sizes. There are three levels of hierarchical block sizes. The location of the fast motion is a column where two or three detected blocks which contain fast motion are overlapping.

As future work, we are going to conduct experiments on the recognition of abnormal motion under stronger occlusion.

References

[1] F. Jiang, J. Yuan, S.A. Tsaftaris, A.K. Katsaggelo, "Anomalous video event detection using spatiotemporal context".*Computer Vision and Image Understanding* 115, pp. 323-333, 2011.

[2] N. Kiryati, T.R. Raviv, Y. Ivanchenko, S. Rochel, "Real-time abnormal motion detection in surveillance video".Proceedings of ICPR. 2008.

[3] M. Baranwal, M.T. Khan, C.W. De Silva, "Abnormal motion detection in real time using video surveillance and body sensors". *International Journal of Information Acquisition.*Vol. 8, No. 2, pp. 103-116, 2011.

[4] M. Park, J. K. Tan, Y. Nakashima, H. Kim, S. Ishikawa, "Detecting human flows on a road different from main flows". Proceedings of AROB. 2011.

[5] P. Mudjirahardjo, J.K. Tan, H. Kim, S. Ishikawa, "Abnormal motion detection in an occlusive environment". Proceeding of SICE Annual Conference 2013, pp. 1398-1402 . 2013.

[6] P. Mudjirahardjo, J.K. Tan, H. Kim, S. Ishikawa, "Fast motion detection in a dynamic background".Proceeding of International Symposium AROB 19th2014, pp. 896-900, 2014.

[7] D. Lyon. "The discrete fourier transform, part 6: cross-correlation", Journal of Object Technology, Vol. 9, No. 2. Pp. 17-22, 2010.

Fig.6. Performance of fast motion detection from various scenes. A detected person with fast motion is indicated by red lines.