

# A Quick Self-Adaptive Background Updating Algorithm Based on Moving Region

QIN Bo, ZHANG Chuangde, FANG Zhenghua, LI Wei

(Computer Science Department, Ocean University of China, Qingdao/China 266071)

## Abstract

Background subtraction is a common way of real-time motion detection. Owing to lacking the right approach to update the background, it was confined in certain aspects. In this paper, we will present a quick self-adaptive background updating algorithm based on moving region, and use label method to de-noise the moving objects. The algorithm can reduce the calculation and accelerate the background subtraction. It also can update the background exactly along with the variance of illumination, and meet the real-time processing. While the label method can de-noise all the noises in the moving objects, the moving objects can be segmented from the background perfectly. The images in our experiment are obtained from the traffic scenes, and the experimental results show that this builds a good foundation for further shadow elimination and object recognition.

**Keywords:** Motion detection; Background subtraction; Background updating; Label

## 1. Introduction

Motion detection based on video is an important research area in Computer Vision. In the real world, a lot of meaningful vision information is contained in the movement. The main part of video image processing is to segment the moving objects from the background quickly and exactly. The real-time motion detection has a broad prospect in many areas, including the Intelligent Transportation System, the Supermarket Surveillance System, and the Automatic Navigation System, etc. Therefore, it has great value in the research and applications.

In recent years, much research both home and abroad has been done on the real-time motion detection. Temporal difference [1], background subtraction [2, 3, 4] and optical flow [5] are three main approaches developed for motion detection. The approach based on optical flow depends on the fact that the motion of moving objects entails intensity

changes in magnitude such that the intensity changes are important cues for locating moving objects in time and space. However, their relationship is not unique because temporal changes can be generated by noise or other external factors like illumination drift due to the weather changing. Moreover, the methods based on optical flow are not computationally efficient. Temporal difference is a simple method for detecting moving objects in a static environment, but the results depend heavily on the visual homogeneity as well as the speeds of the moving objects.

Most typical approaches to motion detection are based on background subtraction. Background subtraction method based on standard intensity may work well if the illumination is constant. Actually any change in illumination will significantly affect the intensity. Under such circumstances, the traditional background subtraction, whether moving object detection or image segmentation, is difficult to figure out that the changes of intensities are due to illumination or object motion, because it uses a stationary reference background image.

Therefore many improved methods have been brought up. In reference [6], a background subtraction method based on edge detection is described. But some motion information may be wiped off if the background edge and the motion edge have overlapping domains. In reference [7], a moving object recognition method is described that uses an adaptive background subtraction technique to segment vehicles from the background. This method cannot be used when the moving objects in the successive video images have overlapped regions. Considering the need of real time, the detection effect can be largely improved if a reasonable background updating mechanism can be brought forward, which will make good use of the advantage of the background subtraction method.

To solve the above-mentioned problems, we present a background subtraction method for motion detection. It contains two parts, the self-adaptive background updating based on moving region and the noise eliminating based on the label method. According to only updating background of the moving

region, it can reduce the calculation and accelerate the background subtraction. Using the label method to denoise in the moving object, the moving objects can be segmented from the background exactly. Many experiments have been done to evaluate the performance of the proposed approach. The results show that the method can detect moving objects accurately under various lighting conditions, and also can satisfy the real-time demand.

The paper is organized as follows: first describe the determination of self-adaptive threshold based on Gaussian model. And then we will present the quick self-adaptive background updating method and discuss the noise elimination method based on label. Finally we present the experimental results and draw the conclusions.

## 2. Determining the adaptive threshold based on Gaussian model

The key of the image difference method is the selection of threshold. Some researchers propose to use a certain threshold, obtained by empiricism in the whole system. However, if the illumination changes largely, this method will cause so much noise to breakdown the whole system. Reference [8] presents a self-adaptive threshold method based on gray histogram. It runs as follows: starting from the pixel value corresponding to the peak of the histogram. We search toward increasing pixel intensities for a location on the histogram that has a value significantly lower than the peak value (mostly 5%~10% of the peak value). This can be used to discard temporal variations due to illumination, but some edge information of moving objects cannot be detected when the gray value of the moving object's edge is close to the background.

Reference [9] supplies an adaptive threshold method based on Gaussian model. The experimental results show that it is not sensitive to illumination, and also can detect the whole object region in any circumstances. The threshold selection is derived from the assumption that the histogram generated from the intensity difference between the two consecutive frames contains three values combined with additive Gaussian noise. We introduce the image obtained by Color Difference Model  $I_D$  to the adaptive threshold method based on Gaussian Model. The color image difference model is defined as follows:

$$I_D[i][j] = \max\{|R_b[i][j] - R_k[i][j]|, |G_b[i][j] - G_k[i][j]|, |B_b[i][j] - B_k[i][j]|\} \quad (1)$$

Where  $R_b[i][j]$ ,  $G_b[i][j]$ ,  $B_b[i][j]$  represent the

current background pixel's red, green, and blue color component, and  $R_k[i][j]$ ,  $G_k[i][j]$ ,  $B_k[i][j]$  represent the current frame pixel's red, green, and blue color component. Experimental results show that the difference image on the color space contains fewer small blots and isolated points than that on the gray level.

The mixture probability density function of the difference model  $I_D$  is

$$P(d) = p_1(d) + p_2(d) + p_3(d) \quad (2)$$

Which is, for the Gaussian case,  $p_1(d)$ ,  $p_2(d)$ ,  $p_3(d)$ , is defined as:

$$p_i(d) = \frac{\omega_i}{\sqrt{2\pi}\delta_i} \exp\left[-\frac{(d - \mu_i)^2}{2\delta_i^2}\right] \quad i=1,2,3 \quad (3)$$

$\omega_i$  is the population proportion,  $\mu_i$  is mean value of the three difference levels,  $\delta_i$  is the variance about the means, and  $d$  is the pixel value of each point in the  $I_D$ .

Let  $ave(d)$  be the histogram probability of a difference value( $d$ ), which is defined as:

$$ave(d) = \frac{1}{n} sum(d), d \in [0, 255] \quad (4)$$

$sum(d)$  is the function of counting the number of the pixels  $d$  in the  $I_D$  and  $n$  is the total number of pixels. The threshold value is determined by a fitting criterion defined as:

$$e = \frac{1}{n} \sum_{d=0}^{255} [P(d) - ave(d)]^2 \quad (5)$$

The best fit between the data and the Gaussian model is found by minimizing the mean squared error between the mixture density  $P(d)$  and the histogram probability of a different value  $ave(d)$ . The  $d$ , whose value can minimize the value of  $e$ , is set as the new threshold.

## 3. Quick self-adaptive background updating

Updating the background in real-time is the key of the motion detection. In the real world, any changes of illumination or weather will affect the intensity of the video image greatly. So the background images must be updated in time. In this paper, a quick self-adaptive background updating algorithm is presented as Fig. 1. The approach consists of three steps: (1) Extract Background Model(BM) using the selective averaging method. (2) Extract Moving Region(MR) with the binary object mask image. (3) In the moving region,

take the difference between Current Image(CI) and Current Background(CB) image to determine which area should be updated or not.

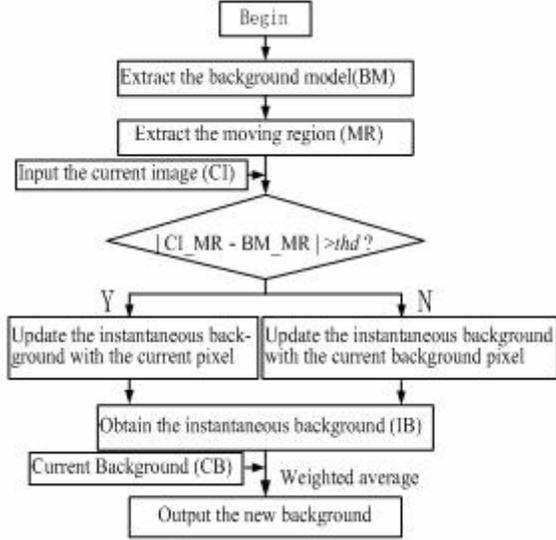


Fig. 1: Flow chart of the background updating algorithm.

### 3.1. Extracting the background model

Extracting the background model is the first step in the preprocessing phase. The background image, only be calculated once, can be used in the following phase. As the video image sequences have the feature of stationary background region, we adopt the selective averaging method [10] to extract the background. Defined as follow:

$$BM[i][j] = \frac{\sum_{m=1}^N PriImg_m[i][j]}{N} \quad (6)$$

Where  $BM[i][j]$  represents the pixel value of the point (i, j) of the background,  $PriImg_m[i][j]$  represents the pixel value of the point (i, j) of the  $m$ th frame of the preprocessed image,  $N$  is the number of the selective image.

The experiment demonstrates that a good background model can be extracted by selective averaging about 100 images.

### 3.2. Extracting the moving region

In this paper a traffic video image is divided into the stationary region (SR) and moving region (MR), and the background updating calculations only occur in the moving region. In this way, the updating calculation can be reduced greatly. Calculations are done to get the difference image between the preprocessed image and the formerly obtained background image. The difference image is thresholded to give a binary object

mask. And the value of *threshold* is determined by the above function (5). Defined as:

$$ImgDif_m[i][j] = |PriImg_m[i][j] - BM[i][j]| \quad (7)$$

$$E_m[i][j] = \begin{cases} 0 & ImgDif_m[i][j] \leq threshold \\ 1 & ImgDif_m[i][j] > threshold \end{cases} \quad (8)$$

Where  $ImgDif_m[i][j]$  represents the  $m$ th difference image, and  $E_m[i][j]$  represents the  $m$ th binary object mask.

We extract the moving region with the  $n$  frames binary object mask image.  $MR[i][j]$  be defined as:

$$MR[i][j] = \begin{cases} 0 & \bigcup_{m=1}^n E_m[i][j] = 0 \\ 1 & others \end{cases} \quad (9)$$

The moving region is the region where the  $MR[i][j]$  value is 1. For example, the white region in Fig. 2 is the moving region.



Fig. 2: The moving region

### 3.3. Self-adaptive background updating

Before background updating, the pixels need to be classified as foreground and background. Background updating uses the background pixels from the current image to modify the current background. The specific background updating process is described as follows.

First, we define the current difference image as:

$$CurImgDif_t[i][j] = |CI_t[i][j] - CB_t[i][j]| \quad (10)$$

Where  $t$  is initialized as 1, when  $t$  equals to 1,  $CB_t[i][j]$  is equal to  $BM[i][j]$ .  $CurImgDif_t[i][j]$  represents the current difference image.

Second, the current difference image is thresholded to give a binary object mask.

$$D_t[i][j] = \begin{cases} 0 & CurImgDif_t[i][j] \leq thd \\ 1 & CurImgDif_t[i][j] > thd \end{cases} \quad (11)$$

Where  $thd$  is the threshold of the current image, and is also determined by function (5).  $D_t[i][j]$  is the binary object mask which is used as a gating function that decides which image to sample for updating the background.

Third, at those locations where the mask is 0 (corresponding to the background pixels), the current image is sampled. At those locations where the mask is 1 (corresponding to foreground pixels), the current background is sampled. The result is called Instantaneous Background (IB).

$$IB_t[i][j] = \begin{cases} CI_t[i][j] & D_t[i][j] = 0 \\ CB_t[i][j] & D_t[i][j] = 1 \end{cases} \quad (12)$$

Finally, the new background is set to be the weighted average of the instantaneous background and the current background.

$$CB_{t+1}[i][j] = \alpha * IB_t[i][j] + (1 - \alpha) * CB_t[i][j] \quad (13)$$

$\alpha$  is a weight factor, assigned to the current and instantaneous background, and affects the update speed. We want the update speed to be fast enough to quickly capture the changes in illumination, but slow enough for the momentary changes (due to, say the AGC of the camera being activated) not to persist for an unduly long amount of time. The weight has been empirically determined to be 0.1. We have found that this gives the best tradeoff in terms of update speed and insensitivity to momentary changes.

#### 4. Quick noise elimination based on label Method

The most employed noise elimination approaches are mathematics morphology, median filter and wavelet de-noising. With these methods to de-noise, there are still some spots in the image, especially in the moving objects. Actually the most part of a binary image is the background, for example the white region of the image in fig. 3 and most of the white region is conjoint, only a few parts are isolated which are the so-called spots. In order to wipe off the spots in the moving objects, the label method is proposed to eliminate the noise quickly. Described as follows:

(1) Set the boundary label as 2.

(2) From point (1, 1) to point (m-2, n-2), where m is the length of the image and n is the width, set the black point label as 1, the white point is 2 if any of these points' label is 2, which lie on the top-left, top and left of this white point, and the other white points are 4, a part of the label is illustrated in fig.3 (a).

(3) From point (m-1, n-1) to point (0,0), when the point with label 4, modify it to 2 if the label of any of these points is 2, which lie on the right, left-bottom, bottom and right-bottom of this point, else modify it to 1. The modified label is illustrated in fig. 3(b).

In the end, the region with the label of 2 is the background, while the region with 1 is the moving object. Now we reach the aim of denoising. This method has small calculation and can meet the need of real-time processing.

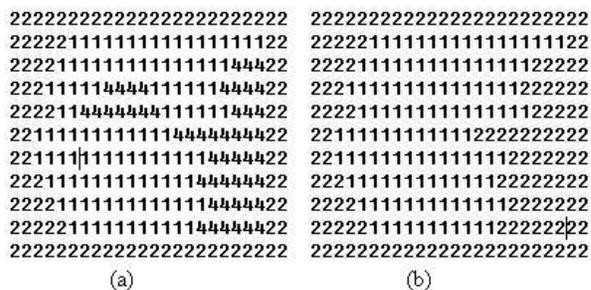


Fig. 3: (a) a part of the label (b) the modified label

The experiment results show the label method can wipe off all the noises in the moving objects. Fig.4 (a) is the binary object mask image after morphological de-noising [11]. There are still some spots in the vehicles. Fig.4 (b) is the binary object mask image after label method de-noising, the spots in the vehicles are all wiped off.

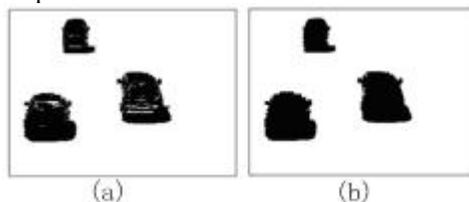


Fig. 4: (a) the binary object mask image after morphological de-noise. (b) the binary object mask image after label method processing.

#### 5. Experimental results

To detect the effect of the proposed algorithm, we use the following traffic video image sequences, with the size of 320\*240, for testing. Fig.5 shows the detection results. (a) is the background model extracted from 100 frames image. (b) is a frame of traffic image. (c) is the result of the vehicles detection. (d) is the updated background image.

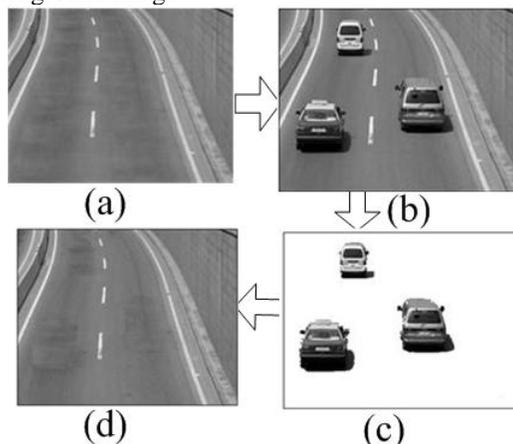


Fig. 5: (a) the background model (b) a traffic image (c) the vehicles detected result (d) the new updated background

We have conducted our experiments on a personal computer with an INTEL Pentium IV 2.93 GHz CPU using Microsoft Visual C++ 6.0. The computation

time per frame ranges from 0.085 to 0.135s, depending on the image quality and the number of vehicles. For most applications, this can be considered real-time. And from the figure above we can see that the moving objects can be segmented from the background perfectly.

The results show that the algorithm can meet the need of real-time processing and detect the moving objects exactly and robustly.

## 6. Conclusion

The quick self-adaptive background updating method based on the moving region can update the background image correctly along with the change of illumination or weather. It can also reduce the calculation and accelerate the background subtraction. The noise elimination using label method can wipe off all the spots in the moving objects, and we can segment the moving object effectively. The results establish foundation for further shadow elimination or objects recognition.

## 7. References

- [1] PL Rosin, T. Ellis, "Image Difference Threshold Strategies and Shadow Detection," *Proc. of the 6th British Machine Vision Conference*, pp.347-356, 1995.
- [2] Jong Bae Kim, Hang Joon Kim, "Efficient region-based motion segmentation for a video monitoring system," *Pattern Recognition Letters*, pp.113-128, vol. 24, num.1-3, 2003.
- [3] M. Seki, T. Wada, H. Fujiwara, and K. Sumi, "Background subtraction based on cooccurrence of image variations," *Proc. Of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 65-72, vol.2, 2003.
- [4] A. Elgammal, D. Harwood, L. Davis, "Non-parametric model for background subtraction," *Proc. Of European Conference on Computer Vision*, Ireland: ACM IEEE, pp.751-767. 2000.
- [5] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, "Statistical and knowledge-based moving object detection in traffic scene," *Proc of IEEE Int'l Conference on Intelligent Transportation Systems*, pp.27~32, Oct. 2000.
- [6] Stauffer C, Grimson W. "Adaptive background mixture models for real-time tracking," In *Proc IEEE Conference on Computer Vision and Pattern Recognition* . pp.246~252, vol.2,1999.
- [7] Surendra Gupte, Osama Masoud, F. K. Martin Robert, and P. Nikolaos, "Detection and Classification of Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, pp.37-47, March vol.1, num.3, 2002.
- [8] H. Nariman, M. Alireza, B. Neil. "Automatic thresholding for change detection in digital video," *Proc of SPIE - The International Society for Optical Engineering, Washington: Society of Photo-Optical Instrumentation Engineers*, pp. 133-142, vol.4067, num. I, 2000.
- [9] J. Kittler, J. Illingworth, "Minimum error thresholding," *Pattern Recognition*, pp. 41-47, vol.19, num.1, 1986.
- [10] Siyal M.Y, Fathy M, "Image Processing Techniques for Real-Time Qualitative Road Traffic Data Analysis," *Real-Time Imaging*, pp.271-278, May, 1999.
- [11] G. Louverdis, M.I. Vardavoulia, I. Andreadis, and Ph. Tsalides, "A new approach to morphological color image processing," *Pattern Recognition*, pp.1733-1741, vol.8, num.35, 2002.