A Multi-Layer Background Subtraction Based on Gaussian Pyramid for Moving Objects Detection

Xuemei Xu, Zhengyu Huang, Qiaoyun Guo, An Li, Qin Mo

Department of Physics and Electronics, Central South University, Changsha, China

xuxuemei999@126.com; zhengyuhw@yahoo.com.cn

Abstract- **In this paper, a real-time multi-layer background subtraction based on Gaussian pyramid is proposed for moving object detection. The proposed method models background on two levels: region analysis in the high-resolution level with averaging background model and pixel analysis in the low-resolution level with hierarchical non-parametric kernel density estimation method. The new method has lower time and space complexities and is more effective than Elgammal's method. Meanwhile, time factor is introduced to refine foreground, and a novel background updating strategy is proposed to adapt to the changes in the scene. Experiment results on both the public video database and our own video database show that the proposed approach has good accuracy and speed, especially against drastic camera shaking.**

Keywords- Background Subtraction; Kernel Density Estimation; Motion Detection; Gaussian Pyramid

I. INTRODUCTION

The detection of unusual motion is the first stage in many automated visual surveillance applications. It is easier for human beings to distinguish moving objects from background than that for computer. In fact, the scene is complex and changing. Background in an outdoor scene may suffer situations such as illumination changing, camera shaking, and objects entering or leaving from background. To deal with these situations and achieve a better performance, it is important to develop a robust detection algorithm with high sensitivity. Meanwhile, the algorithm should have low complexity in real-time application.

In order to solve the scene change problem, several promising methods have been proposed for moving object detection[1-6]. Among them, background subtraction method provides the most complete feature data. For background subtraction method, we need first build a representation of scene background, and then subtract the reference image from the current image. During the detection process, we should update the background model timely owing to the complex change of background scene. It is important that background model could tolerate these kinds of changes, which becomes invariant to adapt to the changes.

The adaptive background subtraction is modeled by averaging the images over time and updated by a running average [7]. The method is less complex and more effective when objects move continuously and the illumination is invariant or changes slowly. But it cannot deal with the situation that moving objects in the scene stop suddenly and then turn into background, and is intolerable to complex scene that contains moving backgrounds like waving tree branches.

Fortunately, statistic model which contains parametric and non-parametric statistics can model complex and non-static backgrounds. Mixture of Gaussians (MOG) [8] can reliably deal with slow lighting, multi-modal distributions and long-term scene changes. However, the assumption that backgrounds are multi-mode Gaussians is not always true. And, if background has high frequency variations, this method fails to achieve sensitive detection. In addition, a trade-off problem exists in MOG.

To solve these problems, Ahmed Elgammal et. al. [9] presented a novel non-parametric background model without using any assumption on the underlying distribution. They utilized general non-parametric kernel density estimation to build a statistical representation of background. The method is quickly adaptive to changes in the detection process and can achieve sensitive detection of moving objects in a messy scene. However, this method is computationally expensive and needs abundant memory to store sufficient sample of recent intensity values. In the literature [10], Ahmed Elgammal introduced the fast Gauss transform algorithm for efficient computation. But there are some drawbacks affecting applications for statistical pattern recognition.

In this paper, we present a multi-layer background subtraction based on Gaussian pyramid. We delaminate the scene into four classes with the former three consisting of moving pixels and the last one with one of static pixels: dynamic background, foreground, and background, which is transformed from foreground and static background. The multi-layer background subtraction is based on two processes: region analysis and pixel analysis. We first make a region analysis to distinguish static pixels from moving pixels with averaging background method, and obtain motion region which contains true foreground and other false foregrounds like dynamic background. Second, we make pixel analysis to eliminate the false foreground. We present a novel hierarchical non-parametric kernel density estimation built on a Gaussian pyramid. The detection is executed in the low-resolution level of Gaussian pyramid and we use the nearest neighbor interpolation to obtain motion region of high-resolution level which contains true foreground and some noises caused by interpolation. Then we take the intersection of the results between region analysis and pixel analysis as true foreground. Finally we introduce time factor to distinguish foreground from background that is transformed from foreground, and further refine the true foreground. In a word, our method combines the advantages of both averaging background method and non-parametric kernel estimation method.

The plan of the paper is as follows. In Section II we explain the region analysis based on averaging background model in detail. In Section III, hierarchical non-parametric kernel density estimation based on Gaussian pyramid is presented. In Section IV, we evaluate the performance of the algorithm and discuss the experimental results. Finally, we summarize the results and indicate future directions.

II. REGION ANALYSIS BASED ON AVERAGING BACKGROUND MODEL

In an intelligent traffic monitoring system, cameras are usually fixed on the roadside poles. This paper focuses on the monitoring system that uses a single camera. In the real monitoring, the scene is complex and changing, and the following factors will affect the performance of background subtraction both in the procedures of model building and model updating: gradual and sudden changes in sunshine, global background changes due to camera shaking caused by wind, and local background changes due to rippling water or waving tree branches, and changes between background and foreground, for example, a car parked at the wayside for a long time should be considered as part of the background.

In this paper, we introduce time factor to handle the transformation between background and foreground. Let $T_{fg}(x_t)$ represent the time duration when pixel x_t is consistently considered as foreground. Through the observation of the historical intensity values, we divide the pixels detected as foreground into true foreground and background. If $T_{fg}(x_t)$ is less than the threshold T_{delete}, it is considered as foreground, otherwise as background transformed from foreground.

After analyzing characteristics of different changes in the scene, we delaminate the scene into four classes: static background containing unchanging or slowly changing background, dynamic background like rippling water or waving tree branches, and background transformed from foreground. We utilize different methods to deal with different layers in order to model the background effectively and update the model timely. For example, we adopt a fast updating strategy for the background that is transformed from foreground and use a normal method for both static background and dynamic background. Our task is to find an excellent method to delaminate the scene.

As averaging background method is less complex and more effective in situations that objects move continuously and the illumination of sunshine is invariant or just changes slowly, we model static background with the method and update the model with a running average. Let $I_n(x)$ represent the intensity value of pixel x at time $t=n$. Let $B_n(x)$ represent the intensity value of the current background model of pixel x at time $t=n$. $Th_n(x)$ is a threshold describing a statistically significant intensity change of pixel x. $B_n(x)$is initialized by averaging background method and updated by different ways on the basis of different layers. $Th_n(x)$ is initialized by a pre-determined non-zero value and written in Formula (1). β is the updating speed and usually set to 0.003.

$$Th_{n+1}(x) = (1-\beta)Th_n(x) + 5\beta(|I_n(x) - B_n(x)|) \quad (1)$$

In the detection procedure, we decide whether the pixel is moving or stationary by subtracting the background model $B_n(x)$ from the current image $I_n(x)$. If Condition (2) is satisfied, the pixel is stationary, otherwise the pixel is moving.

$$|I_n(x) - B_n(x)| < Th_n(x) \quad (2)$$

We need a further process as will be described in Section 3 to decide whether the moving pixel belongs to dynamic background or foreground. After the further process, if $T_{fg}(x)$ of the pixel is larger than the threshold (usually, the threshold is set to 80), the pixel belongs to background transformed from foreground. Thus, we adopt a fast updating strategy shown as Formula (3), which can deal with the situation that moving objects stop suddenly and turn into background. Otherwise, we use a normal strategy in Formula (4) to update the background pixels into background model.

$$B_{n+1}(x) = (1-\beta)I_n(x) + \beta B_n(x) \quad (3)$$

$$B_{n+1}(x) = (1-\beta)B_n(x) + \beta I_n(x) \quad (4)$$

With low time and space complexities, the averaging background method can handle the situation that objects move continuously and the illumination is invariant or just changes slowly. Although this method does not work well in the complex scene, Motion region which contains true foreground with perfect shape and other false foregrounds like dynamic background is obtained. In Section 3, we will eliminate the false foreground to obtain perfect foreground.

III. PIXEL ANALYSIS BASED ON KERNEL DENSITY ESTIMATION

A. Non-parametric Kernel Density Estimation

Compared with parametric statistics, non-parametric statistics can model the real background distribution better. Non-parametric kernel density estimation can achieve accurate estimation of the density function depending only on recent information, and avoid inevitable errors in parameter estimation. Elgammal [9] has specifically described this method. In this section we will give a simple review and then present a novel hierarchical non-parametric kernel density estimation based on Gaussian pyramid. After that, the method using brightness distribution of neighborhood pixels to suppress the false detection due to camera shaking will be described. Finally, we will introduce our model updating strategy.

Let x_1, x_2,$\cdots$, x_N represent a recent sample of intensity values for pixel x. Using this sample, the underlying probability density function (PDF) of pixel x_t can be computed by

$$p(x_t) = \frac{1}{N}\sum_{i=1}^{N} K_\delta(x_t - x_i) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\delta}K(\frac{x_t - x_i}{\delta}) \quad (5)$$

where K is a kernel function, and δ is the bandwidth.

After kernel function and bandwidth are chosen, PDF depends only on recent information from the sequence. So this method allows us to estimate the density function more accurately. In Formula (5), the estimation result is influenced by the bandwidth more than the kernel function. If K is chosen as Gaussian kernel function and we assume that different color channels are independent, then Formula (5) becomes:

$$p(x_t) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi}\delta_j}e^{-\frac{(x_{t_j}-x_{i_j})^2}{2\delta_j^2}} \tag{6}$$

Where d represents the channels, δ_j represents the kernel bandwidth for the jth color channel. Then we can decide whether pixel x_t belongs to foreground or background according to the value of $P(x_t)$. It is worth mentioning that Formula (6) can be used in any color space to effectively suppress shadows. If we use a color space that can separate color information from lightness information, we can easily handle the situation where brightness value is greater or less than a certain limit.

In order to suppress the false detection caused by small movements that are not represented in the background like camera shaking and waving tree branches, correlation between pixels is used in the background modeling. Considering the background distributions in a small neighborhood of the detected foreground pixel, we can decide whether this pixel that just happens to move through a small displacement really belongs to foreground or background.

Let $N(x)$ represent the neighborhood pixels of pixel x and pixel $y \in N(x)$. Then we estimate the probability of pixel x in the distribution of $N(x)$. If there is a pixel y that demonstrate pixel x belonging to background, we should classify pixel x into background. The method is described in the following,

For m=1 to M (M represents the number of the neighborhood pixels of pixel x_t) **do**

i. Estimate the probability of pixel x_t in the distribution of y_m

$$p(x_t \mid y_m) = \frac{1}{N}\sum_{i=1}^{N}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi}\delta_j}e^{-\frac{(x_{t_j}-(y_m)_{i_j})^2}{2\delta_j^2}}$$

ii. If $p(x_t \mid y_m) < th_1$,

$Mask(x_t) = 1$, x_t is a foreground pixel

Otherwise,

$Mask(x_t) = 0$, x_t is a background pixel

Break;

End for

B. Hierarchical Non-parametric Kernel Density Estimation Based on Gaussian Pyramid

Though kernel density estimation method proposed by Elgammal has a lot of advantages, the time and space complexities are high and will affect performance of the algorithm. So we propose a hierarchical method based on Gaussian pyramid and only estimate the pixel probability in the low-resolution level. In the Gaussian pyramid, the resolution reduces gradually from the bottom layer to top Layer. To produce high layer in the Gaussian pyramid from low layer of the pyramid, we first convolve low layers with a Gaussian kernel and then remove every even numbered row and column. We can find that the area of each image is exactly one-quarter of that of the predecessor, as shown in Fig. 1. Compared with Elgammal's method, our method processes only one-quarter of the pixels and thus is more effective. Through estimating the probability in the low-resolution level, we can obtain accurate foreground. We use nearest neighbor interpolation with the detection result to obtain motion region in the high-resolution level. When the image is enlarged within a certain limit (usually under three times), the nearest neighbor interpolation produce an excellent result. Due to the magnification restrictions, we have to limit the layers of Gaussian pyramid and usually construct a Gaussian pyramid of two or three levels which are enough to produce good results.

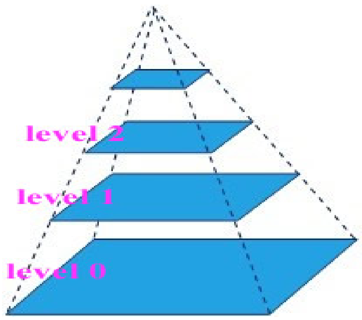


Fig. 1 Gaussian pyramid of a frame

In the process of suppressing the false detection, we find that the size of the neighborhood influences the result of suppression. A much bigger neighborhood produces a better result but with a larger amount of calculation. As we usually choose only 4 neighbors or 8 neighbors, only the false detection caused by small movements is suppressed. If a pixel moves with a larger displacement (e.g. more than one pixel) due to camera shaking to a large extent, the previous method cannot work properly unless we choose a much bigger neighborhood. However, this may cause a large amount of calculation. Fortunately, we can solve this problem with the Gaussian pyramid. The larger displacement in the high-resolution level will become a smaller one in the low-resolution level after down sampling.

The picture as shown in Fig. 2 is a sub-plot of the Gaussian pyramid. Both Pixels P and Q are in the high-resolution level. The distance between P and Q is two pixels. Pixel Pup and Qup are the corresponding pixels in the low-resolution level. From Fig. 2 we can see that pixel Pup is one of the 4 neighbors of Qup. If P has the same original location as Q which belongs to background, P moves to present location. Obviously P will be detected as a foreground, and even if we use the correlation between pixels in the 4 neighbors, we cannot suppress the false detection. But in the low-resolution level, Pup will be classified into background through the correlation with Qup.

The hierarchical non-parametric kernel density estimation based on Gaussian pyramid fills the gap of Elgammal's method, and it not only reduce the amount of calculation but also improve the performance of kernel density estimation.

After detection, Background model needs to be timely updated to adapt to changes in the scene. We present a method that combines selective update and blind update to achieve better update for history samples. This method inherits the advantage of selective update mechanism. In normal circumstances we update the background model based on the detection result. A new pixel is added into the model in a first-in first-out manner only when it is classified as a background pixel. In order to avoid deadlock situations, we adopt a blind update at set intervals.

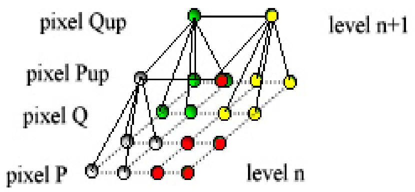


Fig. 2 A subgraph of the Gaussian pyramid shows the neighborhood relationship in the low-resolution layer

C. A Multi-Layer Background Subtraction

Since motion region in the high-resolution level produced by nearest neighbor interpolation contains true foreground and noise resultant from interpolation, we have to filter the noise which is not easy to eliminate by using morphology method or various noise filtering methods because the noise is not independent. The noise may mix with true foreground and become a whole. Thus, we propose a multi-layer background subtraction which combines the results of region analysis and pixel analysis. We take the intersection of the results between region analysis and pixel analysis as true foreground to achieve better detection result.

The whole process of the proposed approach is described as follows,

Step 1 Initialization: construct region model and pixel model for the background. Acquire *N* frames,

1. region model: at each pixel, calculate the mean intensity of these *N*. If the image sequence is in color, each of the R, G, and B components will be calculated.
2. pixel model: first construct a Gaussian pyramid of two levels for each frame. In the low-resolution level, estimate the kernel width as Elgammal described and build a pre-calculated table for the kernel function values.

Step 2 Foreground detection: acquire a new frame

1. In the high-resolution level, use background subtraction to obtain rough motion region (represented by *RM*). If a pixel does not satisfy Formula (2), it belongs to motion region.
2. In the low-resolution level, for every pixel, whether a pixel belongs to foreground is based on two conditions:
   - probability $p(x_t) \le th_1$
   - $Mask(x_t) = 1$

   Thus, accurate foreground in the low-resolution level is obtained (represented by *AF*).
3. Use nearest neighbor interpolation with *AF* to obtain motion region in the high- resolution level (represented by *IAF*).
4. Find the foreground (represented by *FG*) from *RM* and *IAF*

$$FG = \{x_i \mid x_i \in RM \cap x_i \in IAF\}$$

If $x_i \in FG$

$$T_{fg}(x_i) = T_{fg}(x_i) + 1$$

Else

$$T_{fg}(x_i) = 0$$

Step 3 Refine the foreground: turn the foreground pixel whose duration is long into background,

$$FG = FG - \{x_i \mid x_i \in FG \cap T_{fg}(x_i) > T_{delete}\}$$

Step 4 Update the model

1. region model: update averaging background method and threshold as Section 2 describes.
2. pixel model: update the samples.

Step 5 Return to Step 2 for the next frame.

IV. EXPERIMENTS

In this section we describe a set of experiments to evaluate the performance of four algorithms: multi-layer background subtraction (MBS, proposed), averaging background subtraction (ABS [11]), MOG, and non-parametric kernel density estimation (NPK). All of the experiments are made on a standard PC with a 2.2GHz processor and 1GB memory. We implement our experiments on PETS2001 video database with a resolution of 384×288 and our own video database with a resolution of 320×240, respectively, to analyze three features of different algorithms: real time property, accuracy and robustness. The video PetsD2TeC2 from PETS2001 video database is taken with sunshine change. The video QS_street from our own video database is taken with slight camera shaking and video QS_street_shake with obvious camera shaking. In our experiments, the significant parameter values of the four algorithms are listed in Table I.

TABLE I THE SIGNIFICANT PARAMETER VALUES OF THE FOUR ALGORITHMS

Method	Parameter Values
ABS	*N*=35 (frame number for building averaging background model)
MOG	*K*=7 (number of Gauss distributions), *T*=0.7(the minimum portion of the data that belongs to background), *Scale*=3.5 (a match is defined as a pixel value within *Scale* standard deviations of a distribution),α=0.05 (learning rate)
NPK	*N*=50 (the number of samples for each pixel), *W*=100(time size window for sampling), th_1=10e-8(a global probability threshold for a pixel to be considered a foreground pixel)

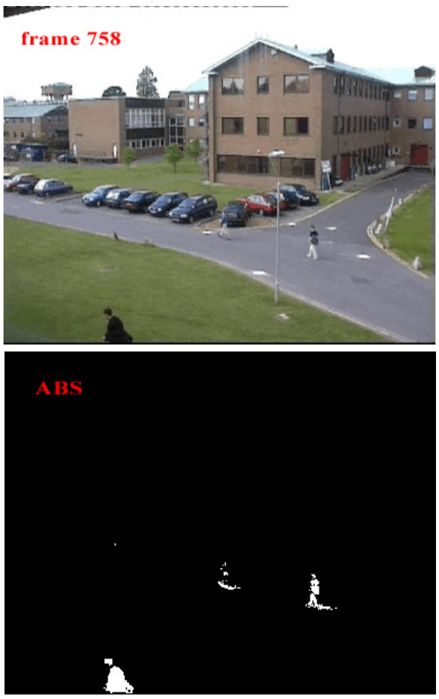
MBS	n=2(the layers of Gauss pyramid), N_size=8(the neighborhood of a pixel), T_{delete}=100(the duration threshold), N=50 (the number of samples in pixel analysis), W=100(time size window for sampling in pixel analysis), th_1=10e-8(a global probability threshold in pixel analysis), β=0.003(updating speed of averaging background model)

The objective of the first experiment is to compare the real time property of the four algorithms listed above. In order to ensure the accuracy of the results, we do not do any extra processing. Table Ⅱ shows the average costing time per frame of every algorithm on different videos. From the results we can see the proposed approach is faster than MOG and non-parametric kernel density estimation, so MBS is effective and practically useful in real-time applications.

TABLE II AVERAGE COSTING TIME (S/FRAME)

	ABS	MOG	NPK	MBS
PetsD2TeC2	0.000017	0.047826	0.046176	0.017689
QS_street	0.000009	0.031414	0.034912	0.014438
QS_street_shake	0.000011	0.031316	0.034709	0.014238

In the second experiment, we choose PetsD2TeC2 to test the algorithms. The first half of the video PetsD2TeC2 containing 2822 frames is normal without camera shaking and sunshine change, but the remaining half is complex due to sunshine change. We choose two representative frames: frame 758 and frame 2383 in the video. As shown in Fig. 3, there are three pedestrians in frame 758. We put names of the algorithms on the respective result images. Because frame 758 is a normal frame without sunshine change, all of the four algorithms produce satisfying results except several false positives due to shadow.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

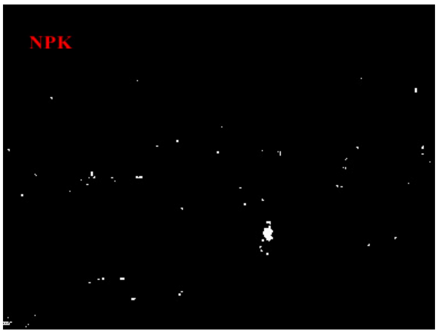
Fig. 3 Detected results on a normal frame without sunshine change: (a) original frame- frame 758, (b) the result of ABS, (c) the result of MOG, (d) the result of NPK, (e) the second layer of Gauss pyramid, (f) detected result on the low-resolution level , (g) the result of MBS(our method)

However, frame 2383 contains obvious sunshine change. In Fig. 4 the detected result of ABS shows that illumination change leads to poor detection: road and house are detected as foreground, and MOG has more false positives. Although NPK adopts a selective update mechanism and performs better than MOG, some false negatives also exist. In our method, we take normal and fast updating strategies in the high-resolution level and adopt a method that combines selective update and blind update in the low-resolution level.

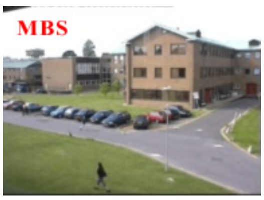
(a)

(b)

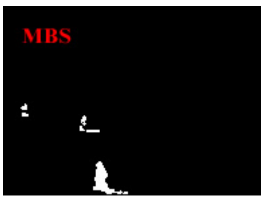
(c)



(d)



(e)



(f)

(g)

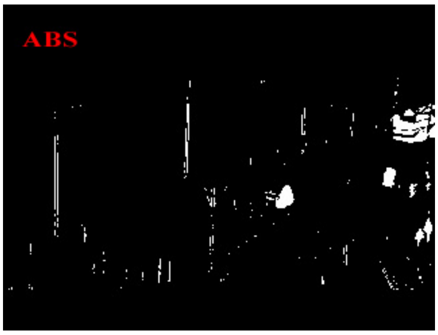
Fig. 4 Detected results on a frame that contains obvious sunshine change: (a) original frame- frame 2383, (b) the result of ABS, (c) the result of MOG, (d) the result of NPK, (e) the second layer of Gauss pyramid, (f) detected result on the low-resolution level , (g) the result of MBS(our method)

From the figures, we can see that our method achieves better performance and is more robust to gradual illumination change, but it is not adaptive to drastic change of sunshine caused by moving clouds. In the future work, it will be necessary for us to build a more stable model.

In the third experiment, we choose the video QS_street_shake to test the algorithms and the objective is to compare the robustness against camera shaking among the four algorithms. As the former 100 frames of QS_street_shake contain obvious shaking, we take frame 65 which contains three moving objects: a bus, a car and a pedestrian as an example.

From the detected results in Fig. 5 we can see ABS and MOG are more sensitive to camera shaking. Some utility poles are falsely detected as foreground. NPK has a better result due to the high detection performance of the non-parametric model, but some strange noise appears. In our method, we suppress camera shaking in the low-resolution level using the neighbors of a pixel. The larger displacement in the high-resolution level will be a smaller one in the low-resolution level after down sampling. From the result we can see that our method is more robust against camera shaking.

(a)



(b)

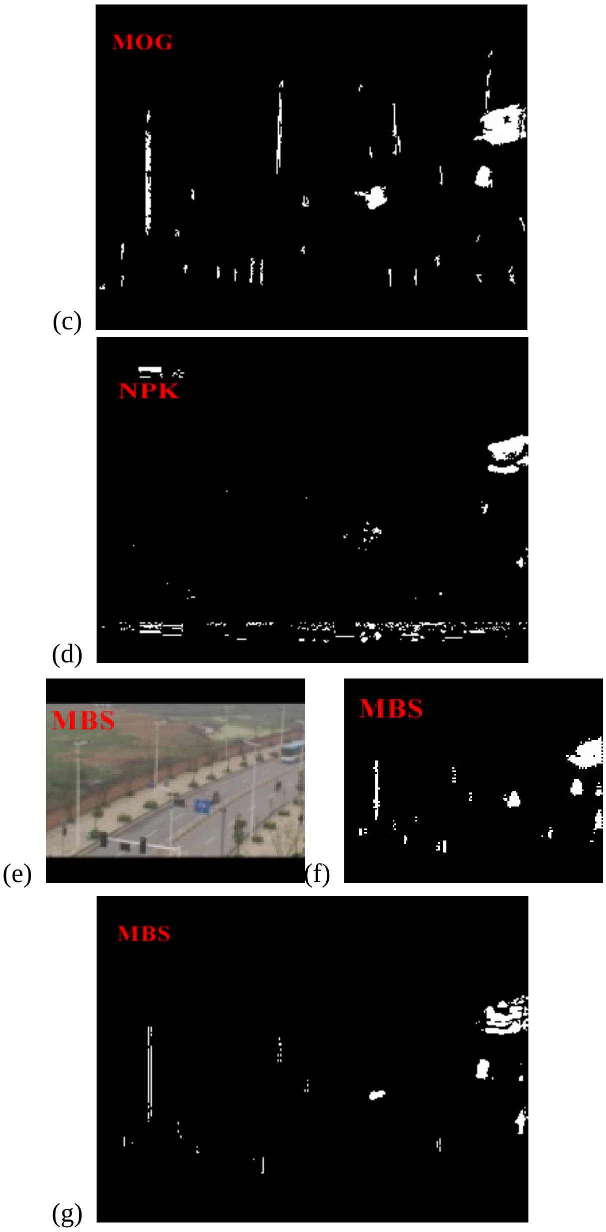


Fig. 5 Detected results on a frame that contains obvious camera shaking:

(a) original frame- frame 65, (b) the result of ABS, (c) the result of MOG, (d) the result of NPK, (e) the second layer of Gauss pyramid, (f)detected result on the low-resolution level, (g) the result of MBS(our method)

The objective of the fourth experiment is to quantitatively evaluate the detection performance. We choose a real transportation video with drastic camera shaking from our video database. We introduce accuracy (ACC) and false positive rate (FPR) to evaluate the detection performance, and use 31 frames of the traffic scene video (every five frames from Frame 239 to Frame 389) to generate the ACC curve and FPR curve as follows. From the results we can see that our method has lower FPR and higher ACC against drastic camera shaking.

$$ACC = \frac{\text{Number of true detected pixels}}{\text{Number of total pixels in a frame}}$$

$$FPR = \frac{\text{Number of false positive detected}}{\text{Number of total pixels in ground truth}}$$

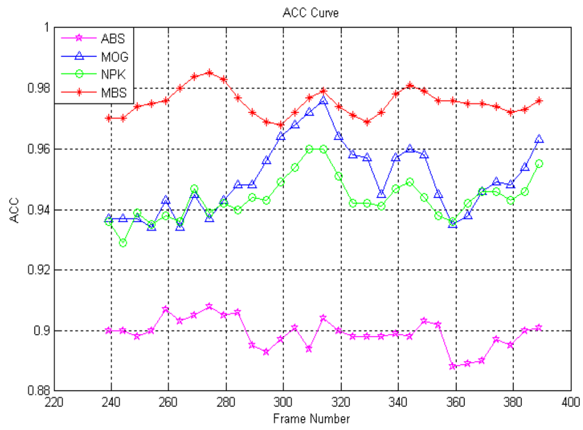


Fig. 6 ACC curves of the four algorithms

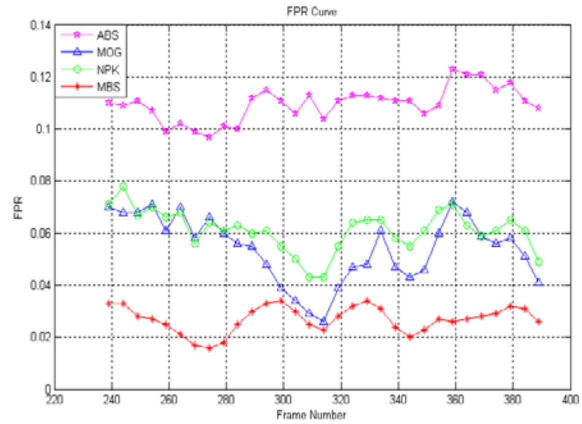


Fig. 7 FPR curves of the four algorithms

From these experiments we notice that our method achieves sensitive detection for different videos and is more adaptive to illumination change and camera shaking with good real-time property.

V. CONCLUSIONS

Moving object detection is a crucial step in surveillance applications and the performance of background subtraction will affect higher-level video applications. In this paper, a multi-layer background subtraction based on Gaussian pyramid is proposed. The proposed method models background on two levels: region level in the high-resolution level and pixel level in the low-resolution level, which use averaging background model and hierarchical non-parametric kernel density estimation method respectively. The new method has lower time and space complexities and is more effective than that of Elgammal's method. It can suppress the false detection resultant from drastic camera shaking (e.g. more than 4 pixels) because of the Gaussian pyramid. Meanwhile, time factor is introduced to refine foreground, which is adaptive to changes introduced into the scene background(for example, if a car is parked in the scene or if a person stays stationary in the scene for an extended period, it should be consider as background, rather than foreground). Then a novel background updating strategy both in region analysis and pixel analysis is proposed respectively to adapt to the changes in the scene. Experiments both on the public video database and our own video database show that the proposed approach handle scenes containing moving background or illumination variations successfully, and achieve robust detection with accuracy and speed for different types of videos.

In our future work, we will build a more stable model to adapt to drastic change of sunshine resultant from moving clouds. And more researches are needed to improve robustness against bad weather like rain and snow.

REFERENCES

[1] Z. Xiang and Y. Jie, "The analysis of the color similarity problem in moving object detection," Signal Processing, 89(4), 2009, pp: 685-691.

[2] Z. Li, P. Jiang and H. Ma, "A model for dynamic object segmentation with kernel density estimation based on gradient features," Image and Vision Computing, 27, 2009, pp: 817-823.

[3] Y. Z. Liu, H. X. Yao and W. Gao, "Nonparametric Background Generation," Journal of Visual Communication and Image Representation, 18(3), 2007, pp: 253-263.

[4] A. Bruhn, J. Weickert and C. Schnörr, "Lucas/Kanade Meets Horn/Schunck: Combining Local and Global Optic Flow Methods," International Journal of Computer Vision, 61(3), 2005, pp: 211–231.

[5] M. Piccardi, "Background subtraction techniques: a review," 2004 IEEE International Conference on Systems, Man and Cybernetics, 4, 2005, pp: 3099 – 3104.

[6] Z. Zivkovic, "Efficient adaptive density estimation per image pixel for the task of background subtraction," Pattern Recognition Letters, 27(7), 2006, pp: 773-780.

[7] T. Kanade, R. Collins, A. Lipton, P. Burt, and L. Wixson, "Advances in cooperative multisensor video surveillance," Proceedings of the 1998 DARPA Image Understanding Workshop, 1, 1998, pp: 3–24.

[8] C. Stauffer and W. E. L Grimson., "Adaptive Background Mixture Models for Real-time Tracking," CVPR, 2, 1999, pp: 246-252.

[9] A. Elgammal, D. Harwood and L. Davis, "Non-parametric Model for Background Subtraction," Computer Vision., 43, 2000, pp: 751-767.

[10] A. Elgammal, R. Duraiswami and L. S. Davis, "Efficient Non-parametric Adaptive Color Modeling Using Fast Gauss Transform," CVPR, 4, 2001, pp: 563-570.

[11] LO B P L, VELASTIN S A. "Automatic congestion detection system for underground platforms" [C], Proc of 2001 Int Symp on Intell Multimedia, Video and Speech Processing. 2000: 158-161.

Xuemei Xu received the Doctor degree in materials physics and chemistry from Central South University in 2005, major in the interface properties and mechanism of organic light-emitting devices. Currently, Main research areas are the mechanism of polymer light-emitting diodes, photonic crystal, photochromic polymer materials, digital image processing, signal integrity, pattern recognition and computer vision.

Zhengyu Huang is studying on Electronic Engineering in the department of Central South University dating from 2010. Currently, his research areas include digital image processing, pattern recognition and computer vision.