Moving Objects Detection and Segmentation Based on Background Subtraction and Image Over-Segmentation

Yun-fang Zhu
College of Computer and Information Engineering
Zhejiang Gongshang University,Hangzhou, Zhejiang, China
Email: zhuyf@zjgsu.edu.cn

***Abstract*—Moving objects detection is a fundamental step in many vision based applications. Background subtraction is the typical method. Many background models have been introduced to deal with different problems. The method based on mixture of Gaussians is a good balance between accuracy and complexity, and is used frequently by many researchers. But it still cannot provide satisfied results in some cases. In this paper, we solve this problem by introducing a post process to the initial results of mixture of Gaussians method. An over-segmentation based on color information is used to segment the input frame into patches. The goal of segmentation is to split each image into regions that are likely to belong to the same object. After moving shadow suppression, the outputs of mixture of Gaussians are combined with the color clustered regions to a module for area confidence measurement. In this way, two major segment errors can be corrected. Finally, by connected component labeling, blobs with too small area are filter out, and the contour of moving objects are extracted. Experimental results show that the proposed approach can significantly enhance segmentation results.**

***Index Terms*—moving objection detection; background subtraction; mixture of Gaussians; color clustering**

I. INTRODUCTION

Moving Objects segmentation is a fundamental and critical task in many vision based applications, such as automated visual surveillance, human-machine interface, and very low-bandwidth telecommunications. A common approach is to perform background subtraction, which identifies moving objects from the difference between the current frame and a reference frame (which often called "background model"). The background model must be representation of the scene with no moving objects and must be kept regularly updated because for some cases, the background is changing when time passes by. Such as view captured by an outdoor surveillance camera, the background is different when sun-light or weather is different.

With respect to the state of the art[1-3], a wide variety of approaches performing background subtraction have been developed. A good review for these methods can be found in [4]. Referring to the conclusions of [4], Mixture of Gaussians [5-7] and Kernel density estimation (KDE) [8] can model well the background pdf in general cases and provide higher accuracy compared to other reviewed methods. If speed is concerned, they both have a const complexity. But KDE has a much higher memory requirement (in order of a 100 frames). So in the real applications, Mixture of Gaussians is the most frequently used method, as witnessed by the huge amount of literature on it.

However, this method also suffers from slow learning at the beginning [9], incapability of identifying moving shadows from the objects casting them [6,10] and unsatisfied results in some cases. These drawbacks can be seen clearly with preliminary experimental results with the datasets provided by PETS2009 [11], as illustrated in Fig 1. Note that the video of the background model is not a literal visualization. It's simply a weighted sum of all components, whether they're part of the background model or not.

From the results we can find that due to low rate of background updating, moving shadows, and possible influence by noise, the performance is poor. Though efforts have been imposed by many researchers to improve the algorithm in different senses, we have to say that a comprehensive physical model of the background is really difficult to develop. Therefore, a good post-processing may be more suitable in general cases. In this paper, a novel color clustering based post-processing method is proposed, and will be discussed in details in the following sections.

II. THE PROPOSED METHOD

Flow chart of the proposed method is given in Fig 2. Inputting a video frame, background subtraction based on mixture of Gaussians is used to segment the moving objects from the static background. Meanwhile, the background model is updated. For the reasons mentioned before, the initial results maybe contain many flaws and must be corrected. In this work, a method based on color clustering and area confidence measurement is proposed. First, the moving shadows are eliminated from the initial results. And the input frame is segmented into patches

Project supported by a grant from the National Natural Science Foundation of China (No. 61002043), the Science Fund for Young Scholars of Zhejiang Gongshang University (No.Q09-23), Science and Technology Planning Project of Zhejiang Province, China (No. 2010C31108).

doi:10.4304/jsw.6.7.1361-1367

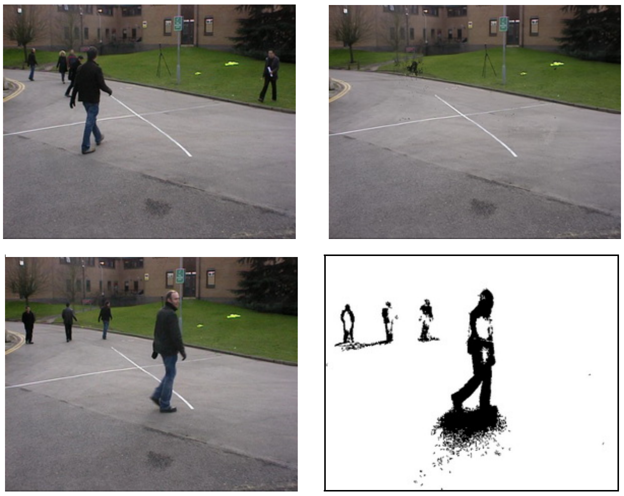

Figure 1. preliminary result on the dataset of PETS2009[11]. upleft : first frame in dataset of S3 (Multiple Flow 12-43). upright: the estimated background in the time of frame 206. lowleft: current frame (frame 206). lowright : the extracted moving regions by mixture of Gaussians implemented in OpenCV[12]

with the similar color by over-segmentation algorithm. Then, these two outputs are combined for confidence measurement based on area analysis. Two major segment errors can thus be corrected. Finally, connected component labeling is used to find the contour of moving objects.

A. Background Subtraction

In the model of Mixture of Gauss [5-7], the background is not a single frame without any moving objects. Rather, the background model is parametric. Each pixel location is represented by a number (or mixture) of Gaussian functions that sum together to form a probability distribution function F,

$$F\left(\mathrm{x}_N\right)=\sum_{j=1}^{k} w_j\cdot\eta(\mathrm{x}_N;\theta_j) \tag{1}$$

where w_j is the weight parameter of the j^{th} Gaussian component. $\eta(\mathrm{x}_N;\theta_j)$ is the Normal distribution of j^{th} component, which can be represented by,

$$\eta(\mathrm{x}_N;\theta_j)=\eta(\mathrm{x};\mu_j,\Sigma_j)=\frac{1}{(2\pi)^{\frac{D}{2}}\left|\Sigma_j\right|^{\frac{1}{2}}}e^{-\frac{1}{2}(\mathrm{x}-\mu_j)^T\Sigma_j^{-1}(\mathrm{x}-\mu_j)} \tag{2}$$

where μ_j is the mean, $\Sigma_j=\sigma_j^2\mathrm{I}$ is the covariance of the j^{th} component.

The Gaussians are ordered by the value of w_j/σ_j ,then the first B distributions are chosen as the background model, where

$$B=\arg\min_b\left(\sum_{k=1}^{b} w_j>\mathrm{T}\right) \tag{3}$$

The threshold T is the minimum prior probability that the background is in the scene.

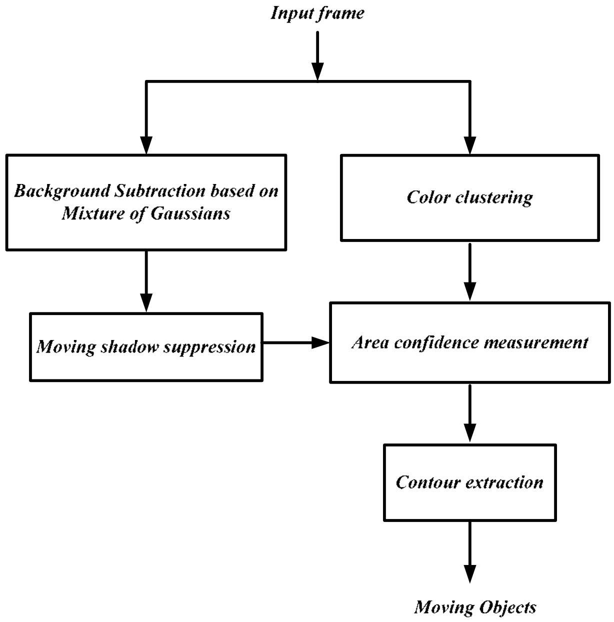

Figure 2. block diagram of the proposed method

To determine if a new pixel is part of the background, we compare it with the existing B Gaussian distributions in turn. If the pixel value is within a scaling factor of a background distribution's standard deviation, it is considered part of the background. Otherwise, it's foreground.

The method of mixture of Gaussians is a good balance between accuracy and complexity, and is used and improved frequently by many researchers. Development in Mixture of Gaussian include a sensitivity analysis of parameters[13], improvements in complexity and adaptation[6,7,14,15], and an extension to construct a panoramic background [16].

B. Moving shadow suppression

As mentioned before, the basic implementation of mixture of Gaussians could not identify the moving shadows casted by the objects. With background subtraction, all the moving points of both objects and shadows are detected at the same time. Moreover, shadow points are usually adjacent to object points and shadows and objects are often merged into a single blob.

Though many color spaces based methods[17] can separate chromatic and illumination components, and maintain a stable chromatic model regardless of the influence of brightness, and thus will improve the performance of shadow detection. But the transform between color spaces will add additional computational burden. Under the consideration of saving computational cost, we adopt the RGB space based method proposed by Horprasert in [6].

The basic idea in [6] is that shadow has similar chromaticity but lower brightness. For a given observed pixel value I_i , a brightness distortion α_i , and a color distortion CD_i , is calculated by

$$\alpha_i=\arg\min_{\alpha_i}\left(\mathrm{I}_i-\alpha_i E_i\right)^2 \tag{4}$$

$$CD_i = \|I_i - \alpha_i E_i\| \quad (5)$$

where E is the expected chromaticity line. α_i equals 1 if the brightness of the given pixel in the current frame is the same as in the background image. α_i is less than 1 if it is darker, and greater than 1 if it becomes brighter than the expected brightness. Then, the criteria for shadow pixels simply becomes,

$$\begin{cases} \tau_\alpha < \alpha_i < 1 \\ CD_i < \tau_{CD} \end{cases} \quad (6)$$

In (6), τ_α and τ_{CD} are predefined thresholds. We use $\tau_\alpha = 0.7$ and $\tau_{CD} = 5$ in our experiments. Fig 3 illustrates the results after the moving shadows have been suppressed.

Compared with result in lowright of Fig 1, there are some noise points in Fig 3. This is because we recalculate the foreground points using the background image.

Figure 3. results after moving shadow suppression

C. Color Clustering

From the result illustrated in Fig 3, we can see that the resulting contours of moving objects have been drawn roughly. But if we inspect the result carefully, it can be seen that there are at least two kinds of false segmentation lying near the contours of moving objects. The first is that background areas are falsely categorized to moving objects. The second is on the contrary.

The reasons behind it may possibly be that the updating rate of background is not fast enough so the background model is not clean enough to extract the moving objects, and it may also be caused by image noise. These false segmentations will certainly degrade the accuracy of further processes, such as objects tracking, and be even worse when objects are close to each other. More over, some of these errors, which connecting with moving objects, can't be eliminated by general processing, such as smoothing, de-noising and erosion-dilation based morphologic operations.

In order to solve this problem without adding too much computational burden, we proposed a novel color clustering based method as a post-processing to correct the false segmentations in the initial results

Color based image segmentation is a process of dividing an image into different regions such that each region has homogeneous color. It is an important operation in many applications of image processing and computer vision, and has been extensively studied [18].

With color based image segmentation, it can provide relatively complete boundaries of objects. Experiences tell us that in most cases, neighboring pixels with similar colors should belong to the same objects, but the reverse-deduction may not be true. So the goal of segmentation is to split each image into regions that are likely to belong to the same object. These regions or segments should be as precise as possible to distinguish the foreground objects from the background areas.

There are many algorithms existing in the literature, for the sake of real-time characteristic, in this paper, we have tried two methods: K-mean algorithm implemented in OpenCV[9] and the method of over-segmentation[19,20]. The main difference between these two methods is the size of segment. The effect of using large segment is that it may straddle more than two objects or between the object and background area. It is undesirable. On the other hand, if the segment is too small, it may not provide sufficient information to distinguish the object from the other object or background.

The use of over-segmentation strikes a good balance between providing segments that contain enough information for distinguishing and reducing the risk of a segment spanning multiple objects or over the background and the foreground area.

The segmentation algorithm has two steps[19]. First, the image is smoothed using a variant of anisotropic diffusion. The purpose of smoothing is to remove image noise. Then, the image is segmented based on neighboring color values.

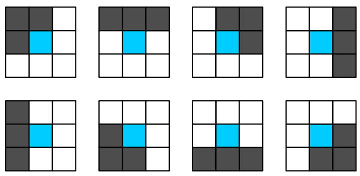

Figure 4. eight directions used for neighboring averaging

The smoothing algorithm iteratively averages along one of the eight directions as shown in Figure 4. The direction is determined by which direction has the minimum sum-of-absolute-differences (SAD) in color from the center pixel.

After smoothing, each pixel is assigned its own segment. Two neighboring 4-connected segments are merged if the Euclidean distance between their average colors varies by less than a threshold (in [19], the value is 6).

If the segment is too small, it will be merged with their most similarly colored neighbors. And if the segment is too large, it will also be divided. For more details, please refer to [19]. A result of the over-segmentation algorithm can be seen in Figure 5. For comparison, images with the averaged color value per segment of K-means and over-segmentation are also illustrated in Figure6.

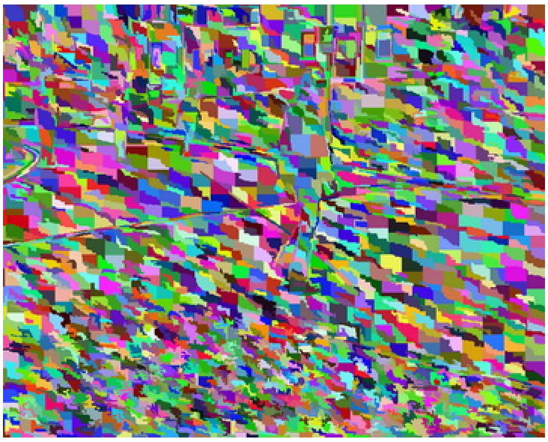

Figure 5. over-segmentation result

Figure 6. images with the averaged color value per segment of K-means and over-segmentation. Up: result of over-segmentation; Low: result of K-means.

Form the results comparisons in Figure 6, we can see that the result of over-segmentation looks more naturally. The reason is that the number of segments in the over-segmentation result is much larger than that of K-means. Though we can enforce the K-means algorithm to cluster more colors, but the price of time-consuming will increase greatly. On the contrary, the over-segmentation algorithm can run very fast, it is important for video surveillance applications.

D. Area confidence measurement

In order to combine the results of background subtraction and color clustering together, we adopt a strategy of area confidence measurement to deal with it. The sketch diagram is given in Fig 7.

Suppose I_f represents the result given by the method of mixture Gaussians, and shadow suppression has also been done on it. I_f is the output as a binary template, in which pixels belonging to moving objects and background are respectively marked with 0 and 1. The clustering results are denoted by I_c.

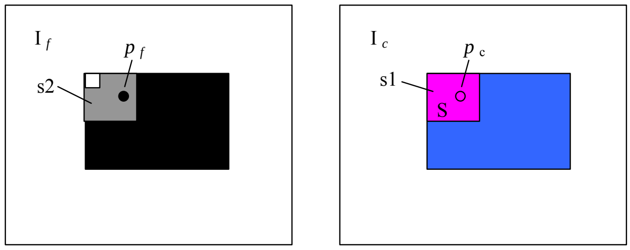

Figure 7. sketch diagram of area analysis

The algorithm starts by scanning I_f, for each point p_f, which belonging to moving objects: $\{p_f \mid p_f \in I_f, p_f = 0\}$, finds its corresponding point p_c in I_c (which means pixel on the same coordinate). Then takes p_c as a seed, searches the points in 8-connected neighbor within the same color patch around p_c. The resulting region is S, whose area is s_1. Returns to I_f, for every point in S, counts the number of pixels whose corresponding point in I_f is 0, and then gets s_2 (the areas corresponding to s_1). s_2 is definitely less than or equal to s_1. Define the ratio k as area confidence by,

$$k = \frac{s_2}{s_1} \tag{7}$$

Compares k with two thresholds: T_1 and T_2 ($0 < T_1 \le T_2 < 1$) .

If $k > T_2$, then the whole corresponding area of S in I_f will be redefined as moving objects.

Otherwise, if $k < T_1$, marks the corresponding area of S in I_f with 1, and is redefined as background pixels.

This step will end when all points marked with 0 in I_f have been scanned and updated. The results are illustrated in Fig 8. Comparing these results with Fig 3, we can find that most of the false segmentations have been corrected.

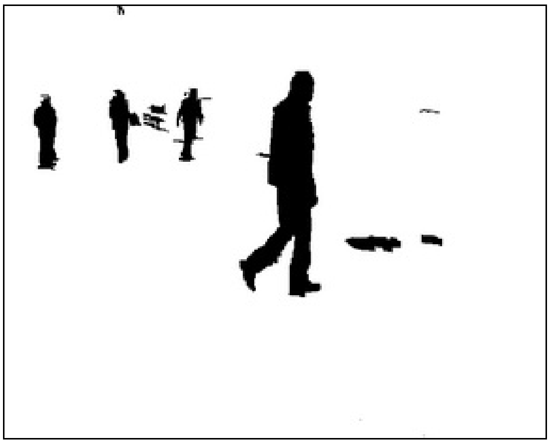

Figure 8. results after area confidence measurement

E. *Contour extraction*

The final step in the proposed method is to extract the contours of moving objects. In this step, a technique of connected component labeling is adopted. It is a basic task in many image processing and computer vision applications. A lot of techniques have been proposed in the past. Though, sophisticated method like snake [21] can provide very good results. But the computational complexity is also high.

For the sake of convenience, we adopt the *cvFindContours* implemented in OpenCV[12]. All connected regions are given a unique label. Each connected region is called a blob. Each blob is then filtered by its area so that only blobs with its area larger than a predefined threshold T_{size} remain. Each remaining blob is correspond to an independent target, and will be used in the further process such as pedestrians tracking algorithm. Figure 9 gives the final results of the proposed method.

III. Experinmental Resuls

In order to prove the effectiveness of our method, experiments have been done with more input frames.

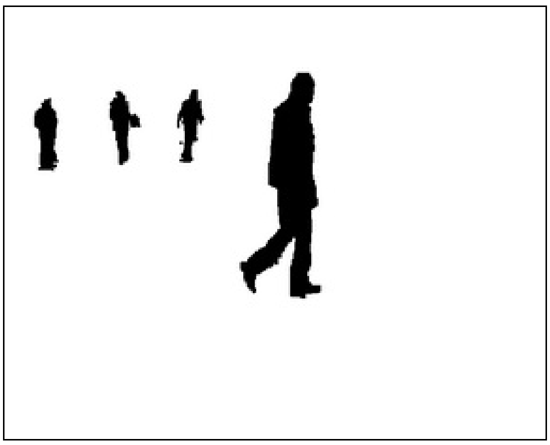

Figure 9. final results after contour extraction

In Fig 10, we give the comparison of final results achieve by k-means and over segmentation of original fig 6. From the two images, we can find some parts of moving objects by k-means are not detected. This is because these parts are converged with background in clustering result, such as the head of the people in red rectangle. If the clustering blocks are small enough, this flaws will disappear.

Fig 11 shows the results of same dataset provided by PETS2009 [11]. We put the original images in first and fourth row, the segmentation results in second and fifth row, the third and sixth row are final segmentation results, from which we can find the moving objects can be correctly detected and the contours of them can be accurately extracted.

We also compare the speed for detecting and segmenting of a frame using the two clustering method. And the result is given in table 1. And all the experiments is making in a computer, whose CPU is Athlon 5200+ and memory is 3G. Test environment is VC6.0.

Table 1 Time comparison of two color clustering

image size	K-means	Over segmentation
720*576	2.457323	1.231228
360*288	0.632535	0.319930

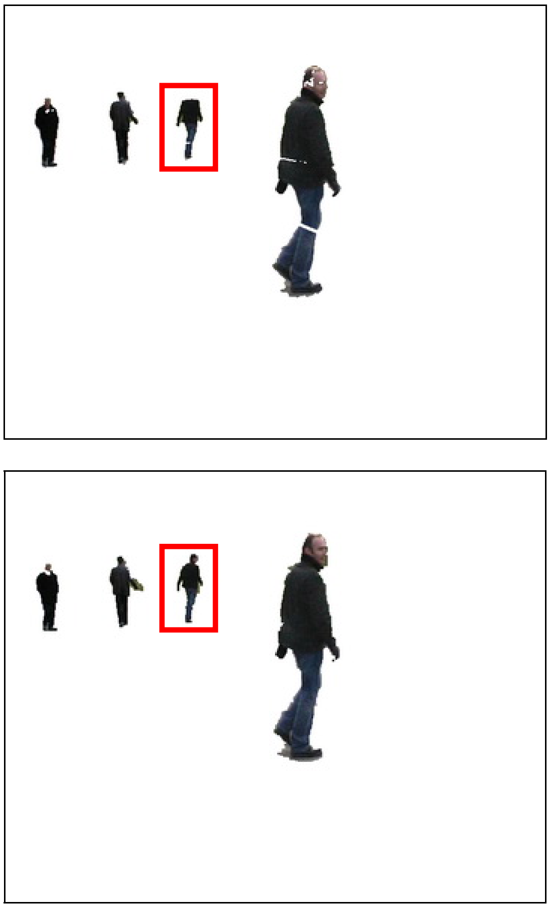

Fig 10 the upper image is the final result given by kmeans, the lower is given by over segmentation

IV. CONCLUSION

Moving objects detection and segmentation is a fundamental step in many applications based on vision. Mixture of Gaussians is the frequently used method to subtracting moving objects from background. But its results are not good enough in some cases. In this paper, a post-processing method is proposed to solve this problem. The results with more complete boundaries provided by the color clustering is used to verify the outputs of mixture of Gaussians, and thus two possible false segmentations can be corrected effectively. Moving shadow suppression and small region filter are also adopted. Using these methods, the results can be greatly improved. Experiments have been done to prove the effectiveness of our work. As a general post-process procedure, the proposed method can also be used for other background subtraction related methods and the results can be used in next step-moving objects tracking.

ACKNOWLEDGMENT

This work was supported in part by grants from the National Natural Science Foundation of China (No. 61002043), the Science Fund for Young Scholars of Zhejiang Gongshang University (No.Q09-23), Science and Technology Planning Project of Zhejiang Province, China (No. 2010C31108).

REFERENCES

[1] C Niu, Y Liu, Moving object segmentation in the H. 264 compressed domain, Lecture Notes in Computer Science,vol 5995,2010:645-654.

[2] W Wang, J Yang, W Gao,Modeling Background and Segmenting Moving Objects from Compressed Video, IEEE Transactions on Circuits and Systems for video technology, Vol. 18, No. 5, 2008:670-681.

[3] Jens Klappstein, Tobi Vaudrey, Clemens Rabe, Andreas Wedel at el,Moving Object Segmentation Using Optical Flow and Depth Information,Lecture Notes in Computer Science, Vol 5414,2009:611-623

[4] Piccardi M. Background subtraction techniques: a review. IEEE International Conference on Systems, Man and Cybernetics, 2004, vol.4: 3099- 3104.

[5] Stauffer C, Grimson W.E.L. Adaptive background mixture models for real-time tracking. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Ft. Collins, 1999: 246-252.

[6] Horprasert T, Harwood D, Davis L S. A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection. Proceedings of IEEE ICCV' 99 Frame-Rate Workshop, 1999, pp.1-19.

[7] KaewTraKulPong P., Bowden R. An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection. In Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, Sept 2001, Pages:1-5.

[8] Elgammal A., Harwood D., Davis L. Non-parametric Model for Background Subtraction. in Proc. 6th Eur. Conf. Computer Vision, vol. 2, 2000, pp. 751-767.

[9] Dar-Shyang Lee. Effective Gaussian mixture learning for video background subtraction. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(5):827 - 832.

[10] KaewTraKulPong P., Bowden R. An improved adaptive background mixture model for real-time tracking with shadow detection. in Proceedings of the 2nd European Workshop on Advanced Video-Based Surveillance Systems, Sept. 2001.

[11] http://www.cvg.rdg.ac.uk/PETS2009/a.html

[12] Intel Open Source Computer Vision Library. URL http://www.intel.com/research/mrl/research/opencv/.

[13] Gao X., Boult T., Coetzee F., Ramesh V. Error analysis of background adaption. in Proceedings IEEE conference on computer vision and pattern recognition, 2000, vol.1, pp. 503-510.

[14] Power P. W., Schoonees J. A. Understanding background mixture models for foreground segmentation. in Proceedings Image and Vision Computing, 2002, pp:267-271.

[15] Lee D.S., Hull J., Erol B. A Bayesian framework for gaussian mixture background modeling. in Proceedings of IEEE International Confererence on Image Processing, 2003, pages:973-976

[16] Mittal A., Huttenlocher D. Scene modeling for wide area surveillancd and image synthesis. in Proceedings IEEE conference on computer vision and pattern recognition, 2, pp. 160-167, (Hilton Head Isand, SC), June 2000.

[17] Cucchiara R., Grana C., Piccardi M., Prati A., Sirotti S. Improving shadow suppression in moving object detection with HSV color information. Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE, 25-29 Aug. 2001 Page(s):334 - 339.

[18] Lucchese L., Mitra S. K. Color image segmentation: A state-of-the-art survey. in Proc. Indian National Science Academy(INSA-A), vol. 67, A, New Delhi, India, Mar. 2001, pp. 207–221.

[19] C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, Richard Szeliski. High-quality video view interpolation using a layered representation. Proceedings of ACM SIGGRAPH 2004, Pages: 600 – 608.

[20] C. Lawrence Zitnick, Sing Bing Kang. Stereo for Image-Based Rendering using Image Over-Segmentation. International Journal of Computer Vision, 2007, 75(1):49–65.

[21] Michael Kass, Andrew Witkin, Demetri Terzopoulos. Snakes: Active contour models. International Journal of Computer Vision. 1988, 1(4) : 321-331.

Yunfang Zhu was born in Jingzhou, Hubei province, China, on February 22, 1980. She received her Ph.D. degree in Information Science and Electronics Engineering in Zhejiang University, China, in 2006.

She was an image algorithm engineer in Omnivision, shanghai from 2006 to 2008. Now she is a lecturer in the college of Computer Science and Information Engineering, Zhejiang Gongshang University. Her result interests are image processing, computer vision and video processing.

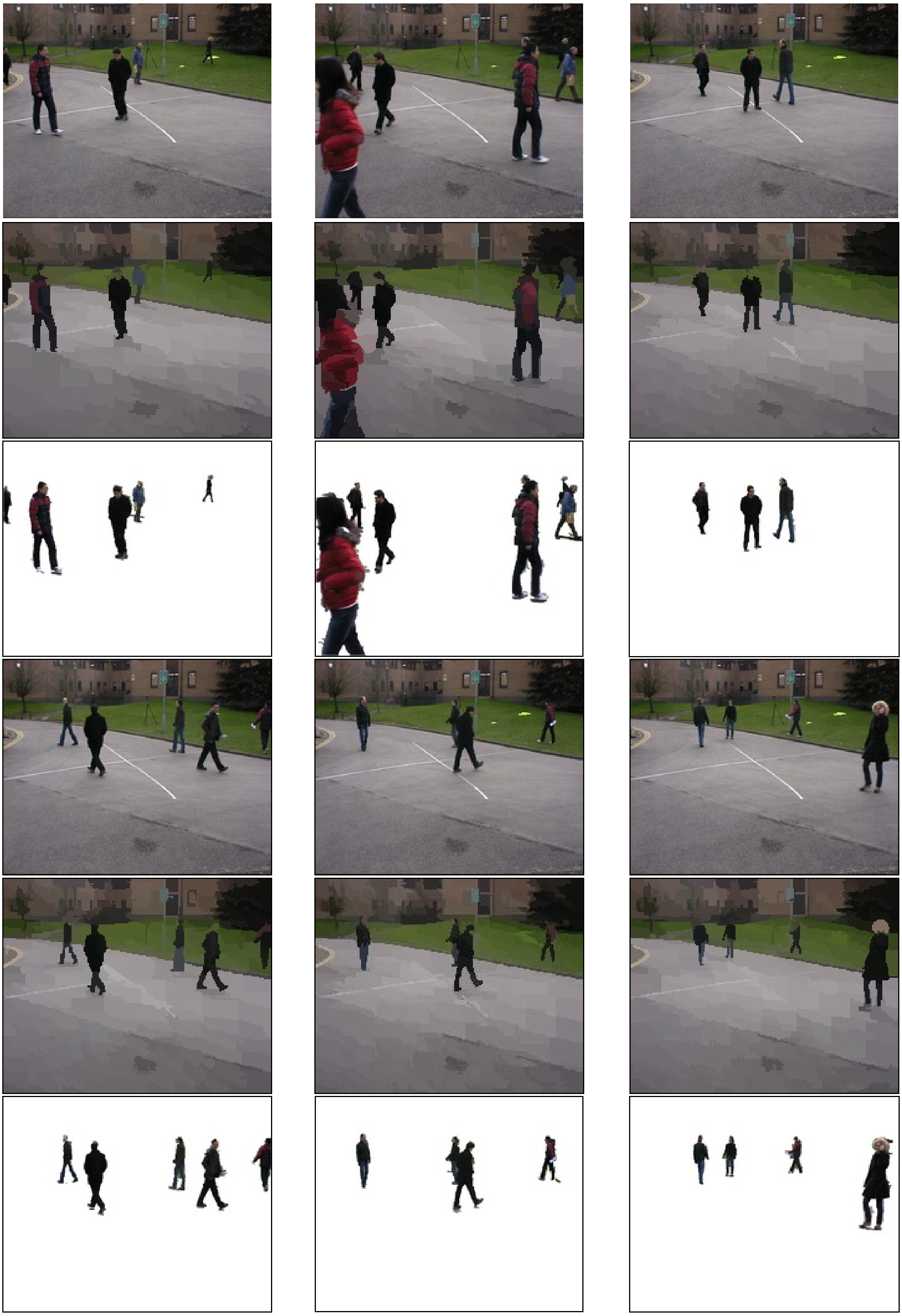

Figure 11. experinmental results of PETS2009. First and forth row: input frame 678, 702,717,753,766,791; second and fifth row: segmentation results; third row and sixth row: segmentation result by the proposed method