EGMM: AN ENHANCED GAUSSIAN MIXTURE MODEL FOR DETECTING MOVING OBJECTS WITH INTERMITTENT STOPS

Huiyuan Fu, Huadong Ma and Anlong Ming

Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia,
Beijing University of Posts and Telecommunications, Beijing 100876, China
fu_hui_yuan@yahoo.com.cn, {mhd, minganlong}@bupt.edu.cn

ABSTRACT

Moving object detection is one of the most important tasks in intelligent visual surveillance systems. Gaussian Mixture Model (GMM) has been most widely used for moving object detection, because of its robustness to variable scenes. However, to the best of our knowledge, existing GMM based methods can not detect moving objects which gradually stop and keep still state for a while. In this paper, we present an Enhanced Gaussian Mixture Model, called EGMM, to handle this problem. We integrate an Initial Gaussian Background Model (IGBM) and an extended Kalman filter based tracker with GMM, to enhance its performance. Experimental results show that our EGMM based method has a lower miss rate at the same false positives per image comparing to GMM based method for moving pedestrian detection, and it also has a higher detection rate for abandoned object detection comparing to GMM based method.

***Index Terms*—** Surveillance, object detection, pedestrian detection, Gaussian mixture model, extended Kalman filter

1. INTRODUCTION

Moving Object detection is a critical task in intelligent visual surveillance. It is the foundation for so many typical vision based applications, such as object tracking and identification. Although it is so important, moving object detection in complex environments is still far from being completely solved.

GMM [1] has been most widely used for detecting moving objects, due to its robustness to lighting changes, and long-term scene changes, etc. Owing to the outstanding performance of GMM on moving object detection, a number of extended methods based on GMM have been proposed [2, 3, 4, 5]. However, existing GMM based methods can not detect the still objects from moving state for a while (in this paper, we can also call this instance as detecting moving objects with intermittent stops), so it will affect many detection based applications. For example, GMM method can not do well for detecting abandoned objects, because the abandoned object is also from moving state to still state. At the same time, GMM method can not work well in a moving pedestrian detection task, due to some people may keep still from moving state for a while.

In this paper, we propose an Enhanced Gaussian Mixture Model, called EGMM, to address this problem. First, we carefully analyse the model of GMM, and find out the reason why it will bring the problem. This is because the distinguish for background and foreground only rely on the match between every pixel's gray and existing Gaussian distributions at this pixel. However, whether these Gaussian distributions belong to background model or foreground model depends on their normalized weight, and the rule for updating weights in GMM will ultimately result in this problem. Then, we present our solution. In order to solve this problem, we not only combine the IGBM with GMM, but also incorporate an extended Kalman filter [6] based tracker into our scheme when the gray value of this still object is similar to the gray value of the background. In this way, we can retain the still object from moving state to still state all-time. Details of our EGMM method can be seen in Section 3. Because most of GMM based methods have not sloved this important problem, we compare our proposed method with traditional GMM method to evaluate its performance for moving object detection with intermittent stops. We compare our proposed EGMM method with GMM method on several data sets. Moreover, we evaluate the EGMM method in a practical moving pedestrian detection experiment and an abandoned object detection experiment. Experimental results demonstrate that our proposed EGMM method can address the problem of detecting moving objects with intermittent stops effectively.

The rest of this paper is organized as follows: Section 2 describes some approaches relevant to GMM method. In Section 3, the proposed method is presented in detail. Experimental results and analysis are described in Section 4. Finally, Section 5 concludes the work.

The research reported in this paper was supported in part by the Natural Natural Science Foundation of China under Grant No. 60833009, No. 60903072, and No. 61003280; the National Science Fund for Distinguished Young Scholars under Grant No. 60925010; the National 973 Project of China under Grant No. 2011CB302700.

978-1-61284-350-6/11/$26.00 ©2011 IEEE

2. RELATED WORK

The most widely used approach for moving object detection is based on GMM [1], which is introduced by Stauffer and Grimson. In this method, each pixel is modeled by using a separate Gaussian mixture, and it will be continuously learnt by an online approximation. Numerous practical applications have demonstrated that it is very useful for detecting moving objects in dynamic change scenes. So it has attracted many researchers to enhance this method further from adaptability, computational complexity, and detection quality, etc.

In order to remove undesirable subtraction due to shadow, automatic white balance, and sudden illumination changes, Zeng et al. [2] designed a two-stage background and foreground classification algorithm based on the previous Gaussian mixture background models (GMMs). First, based on the normalized color and brightness gain information, an adaptive classifier is applied to the foreground pixels in a pixel-wise manner. Then, they group the remaining foreground candidate pixels into regions and compare the corresponding background regions to check if they are foreground regions.

In order to improve the convergence rate, Lee [3] presented an adaptive learning rate for each Gaussian model without affecting the stability. Moreover, a Bayesian framework used to distinguish the most likely background Gaussians and generate an intuitive representation of the background was also incorporated.

In order to improve the computational time of GMM method, Shimada et al. [4] proposed an approach through reducing the number of concurrent models for a pixel by merging. They presented that their algorithm can automatically change the number of Gaussians in each pixel. And they proposed a multi-stage method to improve the detection quality.

3. EGMM FOR MOVING OBJECT DETECTION

3.1. Detecting objects with moving state

In our method, we also use GMM [1] based method to detect the generally dynamic moving objects. Because GMM is robust to dynamic changes, such as lighting changes. We first establish the basic background model by using GMM for detecting dynamic moving objects, and then add the IGBM and the tracker into GMM for detecting still objects from moving state.

In GMM, the recent history of each pixel is modeled by a mixture of K Gaussian distributions. The probability of observing the current pixel value is defined as follows:

$$\mathrm{P}(\mathrm{X}_t) = \sum_{i=1}^{\mathrm{K}} \mathrm{w}_{i,t} \times \psi(\mathrm{X}_t, \mu_{i,t}, \theta_{i,t}) \tag{1}$$

where X_t represents the gray value of current pixel, K is the number of distributions, $\mathrm{w}_{i,t}$ is an estimate of the weight of the i^{th} Gaussian in the mixture at time t, $\mu_{i,t}$ is the mean value of the i^{th} Gaussian in the mixture at time t, $\theta_{i,t}$ is the covariance matrix of the i^{th} Gaussian in the mixture at time t, and ψ is a Gaussian probability density function as follows:

$$\psi(\mathrm{X}_t, \mu, \theta) = \frac{1}{(2\pi)^{\sqrt{|\theta|}\frac{n}{2}}} e^{f(t)} \tag{2}$$

where $f(t)$ is defined as follows:

$$f(t) = -\frac{1}{2}(X_t - \mu_t)^T \theta^{-1} (X_t - \mu_t) \tag{3}$$

Thus, the distribution of recently observed values of each pixel in the scene is characterized by a mixture of Gaussian distributions. The gray value of a new pixel will be represented by one of the major components of the mixture model and used to update the model. At first, all K gaussian distributions are seen as background model. In time t, if there is none of K distributions matching the current pixel, the current pixels Gaussian model will replace one of above K distributions whose weight is the lowest, and all of the weights will be changed. If there is one of K distributions matching the current pixel, its weight will increase. Iteratively, initial K gaussian distributions will be departed into background model and foreground model according to their weight. The number of background model at time t can be defined as follows:

$$B = \arg\min_b \left(\sum_{k=1}^{b} w_k > W\right) \tag{4}$$

where W is a predefined threshold which means the minimum weight representing for the background. The current pixel will be classified to foreground or background by comparing to the K gaussian distributions. If it matches current first B distributions, it is a background pixel, otherwise it is a foreground pixel. In this way, we can update the background model and foreground model dynamically according to the dynamic changes.

3.2. Analysis on detecting still objects from moving state

As we can see in Eq. (4), the value of B decides which Gaussian distributions are chosen as the background model and the others are chosen as the foreground model. We find that w_k (k=1,...,K) is the typical factor, and it can be defined as follows:

$$w_{k,t} = w_{k,t-1} + \alpha(M_{k,t} - w_{k,t-1}) \tag{5}$$

where α is the learning rate and $M_{k,t}$ is 1 for the model which matched and 0 for the remaining models.

Eq. (5) is the rule for updating weights of K different Gaussian distributions in GMM method. Due to this rule, we can not accurately detect the still objects from moving state.

Because at first, all distributions are seen as background model. When a new object is appeared to move, the pixel's Gaussian distribution in this object can not match them (assume that this time is T). So the pixel's distribution will add into the current model with an initially high variance, and low prior weight. According to Eq. (4), it is a foreground model, we call it I. In time T + 1, the current pixel will match the foreground model I, and the model I's weight will increase according to Eq. (5). But the first background model's weight is high enough to be fit for Eq. (4), so the moving object is represented as foreground.

For the same reason, when a moving object is gradually to stop, it will also be detected as foreground. However, if a moving object has stopped and keep still for some time, the new Gaussian model will be generated. Its weight will increase gradually with time accumulating, and it will become a background model finally. Thus, the still object will be fused into the background. This is why traditional GMM can not detect the still objects from moving state robustly.

3.3. Detecting still objects from moving state

Based on above analysis, we find that the weight updating rule in Eq. (5) is a key factor. The rule is suitable for moving objects, but it will make still objects from moving state be fused into the background. In order to solve this problem, we propose our EGMM based method. In our method, we combine the IGBM and an extended Kalman filter based tracker with GMM. First, IGBM can help to retain the still objects from moving state when the gray of still object is not similar to the gray of background. Meanwhile, the tracker in our scheme can detect robustly when the gray of still object is similar to the gray of background.

Considering that all distributions are background model in the beginning of GMM method, we assume there are total K single Gaussian distributions in GMM, so we can use these K initial Gaussian distributions to construct a new background model. We call it IGBM. Because it is composed of total single Gaussian distributions from the initial mixture of Gaussian distributions, it can retain the most abundant information of the original background model. Gaussian distributions can be represented by two key parameters μ and σ, so IGBM can be defined as follows:

$$\begin{cases} \mu_{i,t}(IGBM) = \mu_{i,t}(GMM) \\ \sigma_{i,t}(IGBM) = \sigma_{i,t}(GMM) \end{cases} \quad s.t. \quad i = 1, ..., K \tag{6}$$

where $\mu_{i,t}$ and $\sigma_{i,t}$ are the mean value and standard deviation of the i^{th} distribution of IGBM and GMM at time t, respectively. $\sigma_{i,t}$ is defined as follows:

$$\sigma_{i,t} = \sqrt{1-\delta} \times \sigma_{i,t-1} + \sqrt{\delta \times (X_t - \mu_t)^T \times (X_t - \mu_t)} \tag{7}$$

where X_t is the current pixel coordinate, δ represents the updating speed, and μ_t is defined as follows:

$$\mu_{i,t} = (1-\varsigma) \times \mu_{i,t-1} + \varsigma \times X_t \tag{8}$$

where ς represents the updating speed too. Although we have constructed the IGBM, we also need to update it when the current pixel fits for its distribution due to dynamic scene changes. The updating process can be seen in Eq. (7) and Eq. (8), respectively.

When the gray of still object is not similar to the gray of background, the difference between the Gaussian distributions of still objects and IGBM is obvious. Suppose at time t, the pixel $P(x, y)$'s Gaussian distribution of still objects from moving state just has the biggest weight, its mean value is $\mu_{\max,t}(P(x, y))$. The current updated initial Gaussian distribution's mean value is $\mu_{i,t}(IGBM)$. Motivated by [1] that a match is defined as a pixel value within 2.5 standard deviations of a distribution, so if $\mu_{\max,t}(P(x, y))$ does not match all of K single Gaussian distributions in IGBM, the pixel is regarded as a part of a still object. Whether the pixel belongs to a still object or not can be defined as follows:

$$\varphi_t(x, y) = \begin{cases} 1 \text{ if } |\mu_{\max,t}(P(x, y)) - \mu_{i,t}(IGBM)| > 2.5\sigma_{i,t} \\ 0 \quad \text{otherwise} \end{cases} \quad s.t. \quad i = 1, ..., K \tag{9}$$

where $\sigma_{i,t}$ means the standard deviation of IGBM.

But when the gray of still object is similar to the gray of initial Gaussian background, it is difficult to distinguish them according to Eq. (9). In this instance, we will incorporate an extended Kalman filter based tracker into our scheme. Extended Kalman filter is always used to deal with nonlinear prediction problems. In our framework, the motion of moving objects are random, and their sizes change largely, so it is a nonlinear problem.

The main process of our tracker is described as follows:

- To take GMM algorithm to get the binary foreground frames from the original images.
- To detect each rectangle blob in the current foreground frame, then we record every blob centroids coordinate as $P'(x, y)$.
- To predict each blob centroids coordinate $Q(x, y)$ in the next frame by using extended Kalman filter, and suppose that the initial prediction $Q(x, y)$ equals to $P(x, y)$.
- To detect every blob centroids coordinate $P(x, y)$ in the next frame. So we can get the final blob centroids coordinate R(x, y) as follows:

$$\mathrm{R(x, y) = P(x, y) + \xi(Q(x, y) - P(x, y))} \tag{10}$$

  where ξ is defined as follows:

$$\xi = \begin{cases} 0 \text{ if } \exists P(x, y) \\ 1 \text{ otherwise} \end{cases} \tag{11}$$

- Finally, to process all foreground frames in this way iteratively.

It means that $P(x, y)$ is always existed when objects are moving. We assume that ξ is 0, so we do not use prediction module in Eq. (10). But when some objects are keeping still for a while from moving, we can not detect them since one frame during this time. In this circumstance, we will use prediction module. We record this frame and its previous frame as $I(t)$ and $I'(t)$, respectively. In frame $I'(t)$, we assume that each blob centroid's coordinate is $A(x, y)$, so the value of $Q(x, y)$ equals to $A(x, y)$ now. In frame $I(t)$, $P(x, y)$ is not existed, so the value of ξ equals to 1, and the value of R(x, y) equals to $Q(x, y)$ according to Eq. (11). In the next frame of $I(t)$, $Q(x, y)$ will not change according to the theory of extended Kalman Filter for predicting. As we can know, $Q(x, y)$ will keep the same value as long as the objects are keeping still. So in this situation, the value of ξ is always 1, and $R(x, y)$ will equal to $Q(x, y)$ all time. In this way, we can get the correct detection results on still objects.

Although we can straightly take the tracking method to solve the problem, the consuming time is much more than solving this problem only in the background modeling phase. This will affect the real-time application. However, merely depending on background modeling is not able to address our problem in some environments. So in our scheme, we take a balance strategy. We combine the modeling method and tracking method together. When the moving object is just entering the monitor region, we compare the pixel's Gaussian distribution with current updated IGBM according to Eq. (9). If a large portion of points in the moving object fit for IGBM, we use the tracking method to tackle it. Otherwise we use the model comparison method based on IGBM. This strategy can help us to achieve a well effect balance.

4. EXPERIMENTAL RESULTS

4.1. Experiment overview

Our objective is to compare the performance of EGMM and GMM for moving object detection, especially on detecting moving objects from moving state to still state. In order to evaluate our algorithm, three experiments are conducted. In the first experiment, we compare EGMM and GMM from running the algorithms directly. In the second experiment, we compare them in a practical moving pedestrian detection task. In the third experiment, we use EGMM method on an open data set [7] to testify its performance. In our experiments, we capture our own data set for testing by an AXIS-215- PTZ camera (see Fig. 1) from a practical surveillance scene. Moreover, we also compare them on two well-known open data sets: PETS 2006 [7] and PETS 2000 [8]. All of our experiments are conducted on an Intel E7500, 3 GHz dual core processor with 2GB RAM.

4.2. Comparison on different data sets

In the first experiment, we directly compare EGMM and GMM algorithms under the same scene. In order to compare their performance on detecting still objects from moving state, we not only capture a practical video (30 minutes) by

Fig. 1. (a) The AXIS-215-PTZ camera on the top left; (b) The surveillance scene captured by the camera.

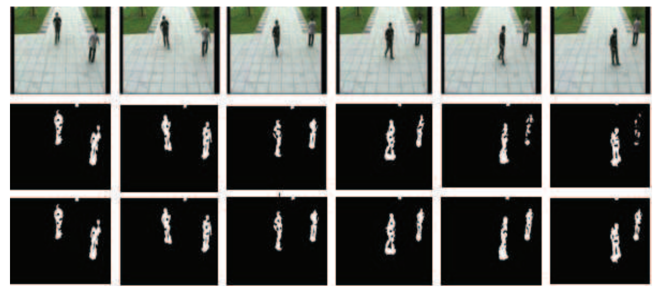

Fig. 2. Retain still objects (humans) from moving state in the binary foreground. Top: original frames; Middle: GMM method corresponded frames to the top frames; Bottom: EGMM method corresponded frames to the top frames. EGMM method is able to retain the still people, while GMM method is not. This sequence is from the camera. The binary result is through morphology dilate for a better view.

the AXIS-215-PTZ camera shown in Fig. 1, but also run them in the same video cropped from the open data set PETS 2000 [8]. These two test videos both contain the instance that some objects just keep still from moving state for a while. The experimental results show that EGMM method can retain the non-moving objects (humans or vehicles) in the foreground (as seen in the bottom line of Fig. 2 and Fig. 3), but the still objects will be vanished when using GMM method (as seen in‘ the middle line of Fig. 2 and Fig. 3).

4.3. Comparison on moving pedestrian detection

In the second experiment, we evaluate our proposed EGMM method in a moving pedestrian detection task. A practical video (50 minutes) captured from the camera (as shown in Fig. 1) is used. It contains lots of instances that some moving objects just keep still from moving state for a while.

HOG-LBP [9, 10] based method is one of the states-of-the-art approaches for pedestrian detection, it combines the advantages of Histogram of Oriented Gradients (HOG) [11] and Local Binary Pattern (LBP) [12, 13] together. It both detects pedestrians based on feature, so it can work on moving people and still people. However, its processing speed is so slow that can not be applied in real-time task, due to that it scans whole of the image every time (see Fig. 6).

In this experiment, we use GMM and EGMM as a filter before processing HOG-LBP method. First, we assume that

Fig. 3. Retain still objects (vehicles) from moving state in the binary foreground. Top: original frames; Middle: GMM method corresponded frames to the top frames; Bottom: EGMM method corresponded frames to the top frames. EGMM method is able to retain the still vehicles, while GMM method is not. This sequence is from PETS 2000 [8]. The binary result is through morphology dilate for a better view.

all people are moving at first (because the paper focuses on solving the problem of detecting moving objects from moving state to still state), so we can filter out the moving object foreground. Then, we only process HOG-LBP method in the filtered foreground. Thus, the consuming time for detecting moving people is greatly decreased. As seen in Fig. 6, EGMM+HOG-LBP method has an approximate speed to GMM+HOG-LBP method, but it is much more fast than HOG-LBP method.

However, EGMM+HOG-LBP method is more robust than GMM+HOG-LBP method. When some moving people are stopping and standing for a while, GMM algorithm can not detect them. In Fig. 4, a person is standing still from moving (in the left and top of Fig. 4), GMM will not detect him/her in the foreground (in the middle and top of Fig. 4). In this way, following HOG-LBP method can not correctly detect him/her (in the right and top of Fig. 4). Comparing to GMM, EGMM can retain the still person in the foreground, and following HOG-LBP method can detect all people right (as seen in the bottom line of Fig. 4).

To quantify the EGMM performance, we use Detection Error Tradeoff (DET) curves [14], plots of miss rate versus false positives per image (FPPI). The lower miss rate means better detection performance on the same FPPI. We evaluate EGMM+HOG-LBP, GMM+HOG-LBP and HOG-LBP detectors, respectively. Fig. 5 shows the performance comparison, and we can see that EGMM+HOG-LBP method outperforms the other two methods all the time. This experiment demonstrates that EGMM is not only better than GMM for robust moving pedestrian detection (see Fig. 5), but also a positive complement for real-time detection (see Fig. 6).

4.4. Comparison on abandoned object detection

In the third experiment, we compare the GMM and EGMM methods by detecting the abandoned luggage. In general, a luggage is keeping moving state with its owner, however, it

Fig. 4. The comparison between GMM and EGMM algorithms as filters for moving pedestrian detection. Top: GMM is used; Bottom: EGMM is used. EGMM can filter out all people (including still people from moving state), but GMM can not (such as the middle pedestrian of the picture). The binary result is through morphology dilate for a better view.

will change to still state when it is abandoned. It is meaningful to remind the owner if the abandoned luggage can be detected. We use open PETS 2006 [7] data set to evaluate the algorithms, because it is a specific data set for detecting abandoned objects. The typical scene in PETS2006 is a subway. As seen in Fig. 7, a man with a bag on his shoulders is moving with another man at first (corresponding to the left frame of Fig. 7). Then, he forgets to take his bag away (corresponding to the middle frame of Fig. 7). EGMM and GMM algorithms are run on this scene, respectively. GMM method will fuse the still bag into background (corresponding to the right frame of Fig. 7), so it can not detect the abandoned bag. But EGMM method will also work in this instance (as seen in the bottom line of Fig. 7, the detected objects are labeled in green rectangle, and the abandoned luggage is labeled in white ellipse).

5. CONCLUSIONS

In this paper, we present a new model, EGMM, for moving object detection. In our scheme, we combine the IGBM and the extended Kalman filter based tracker together. Our proposed method can detect the still objects from moving state for a while, which enhances the performance of traditional GMM. Through a large set of experiments, we show that EGMM can work well for robust moving object detection.

6. REFERENCES

[1] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 246–252, 1999.

[2] H. C. Zeng and S. H. Lai, "Adaptive foreground object extraction for real-time video surveillance with lighting variations," *IEEE International Conference on Acous-*

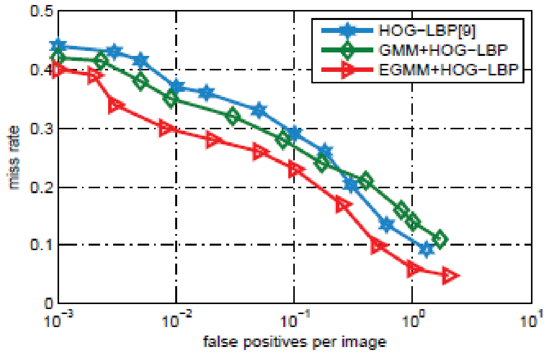

Fig. 5. The comparison of detection accuracy between three approaches for moving pedestrian detection. EGMM+HOGLBP method is more accurate than GMM+HOG-LBP method and HOG-LBP [9] method alltime.

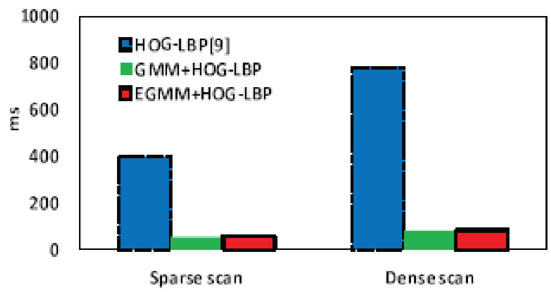

Fig. 6. The comparison of detection speed per image between three approaches for moving pedestrian detection. EGMM+HOG-LBP method has an approximate speed to GMM+HOG-LBP method, but it is much more fast than HOG-LBP [9] method.

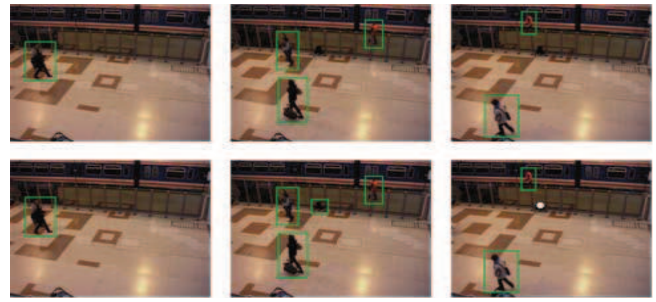

Fig. 7. The comparison between GMM and EGMM algorithms for abandoned luggage detection. Top: GMM is used; Bottom: EGMM is used. EGMM can detect the abandoned luggage (the white ellipse labeled), but GMM can not detect.

tics, Speech, and Signal Processing, vol. 1, pp. 1201–1204, 2007.

[3] D. S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 27, pp. 827–832, 2005.

[4] A. Shimada, D. Arita, and R. Taniguchi, "Dynamic control of adaptive mixture of gaussians background model," *IEEE International Conference on Video and Signal Based Surveillance*, pp. 20–24, 2006.

[5] P. K. Atrey, V. Kumar, A. Kumar, and M. S. Kankanhalli, "Experiential sampling based foreground/background segmentation for video surveillance," *IEEE International Conference on Multimedia and Expo*, pp. 1809–1812, 2006.

[6] X. Kai, C. L. Wei, and L. D. Liu, "Robust extended kalman filtering for nonlinear systems with stochastic uncertainties," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 40, pp. 399–405, 2010.

[7] http://www.cvg.rdg.ac.uk/PETS2006/, "Pets 2006: Performance evaluation of tracking and surveillance," Jun 2006.

[8] http://www.cs.berkeley.edu/ flw/tracker/pets2000.html, "Motion tracking: Pets 2000 sequence,," 2000.

[9] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," *IEEE International Conference on Computer Vision*, vol. 1, pp. 1–8, 2009.

[10] C. B. Zeng, H. D. Ma, and A. L. Ming, "Fast human detection using mi-svm and a cascade of hog-lbp featrues," *IEEE International Conference on Image Processing*, pp. 3845–3848, 2010.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.

[12] Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou, "Discriminative local binary patterns for human detection in personal album," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[13] D. T. Nguyen, Z. Zong, P. Ogunbona, and W. Li, "Object detection using non-redundant local binary patterns," *IEEE International Conference on Image Processing*, pp. 4609–4612, 2010.

[14] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 304–311, 2009.