Histogram-based Foreground Object Extraction for Indoor and Outdoor Scenes

Mandar Kulkarni*
IPCV Lab, Electrical Engg. Department
Indian Institute of Technology Madras
Chennai, India
mandareln40@gmail.com

ABSTRACT

Extracting foreground objects is an important task in many video processing/analysis systems. In this paper, we propose a technique for foreground object extraction, under static camera condition. In our approach the spatial histogram of a single background image is modeled as Mixture of Gaussians and this model is updated after every few frames. To extract the foreground, input frames are compared with current background frame model and foreground pixels are classified according to intensity differences. To mitigate the errors caused due to movement of the background objects (e.g tree leaves in outdoor scenes), we also incorporate optical flow in an efficient manner. We demonstrate performance of our approach on various indoor and outdoor scenes.

1. INTRODUCTION

Foreground extraction is an important task in many computer vision applications. In this paper, we propose an method which models the histogram of an initial background frame by the mixtures of Gaussians. Generally, a natural background includes large objects such as trees, road, floor, buildings, walls etc., each of which contains pixels with similar intensity values, but whose intensities differ considerably from each other. Hence, the histogram of the background frame containing multiple objects, is usually multi-modal and can be approximated by the Mixture of Gaussians [7]. The number of Gaussians is determined by the number of objects present in the background. We also update the background histogram model at regular intervals to adapt to illumination variations over time. We use the Expectation Maximization (EM) algorithm to find maximum likelihood parameters of every Gaussian component.

To detect the foreground objects, we compare input frame with the current background histogram model. Pixels showing higher intensity deviations than background pixels, are classified as foreground objects. The threshold for foreground classification is computed from the current background model. We also account for the fact that if a classified foreground object remains stationary for long time, its corresponding pixels are re-classified as background. To improve the results under significant background motion, we also incorporate optical flow efficiently in our framework. We provide various qualitative and quantitative results on indoor and outdoor scenes.

An illustration of our approach is shown in Fig. 1. Note the different intensities of the background objects such as the road, building etc. These differences show up in the multi-modal histogram (Fig.1 (b)) where blue line indicates histogram of background frame and red line indicates Gaus-

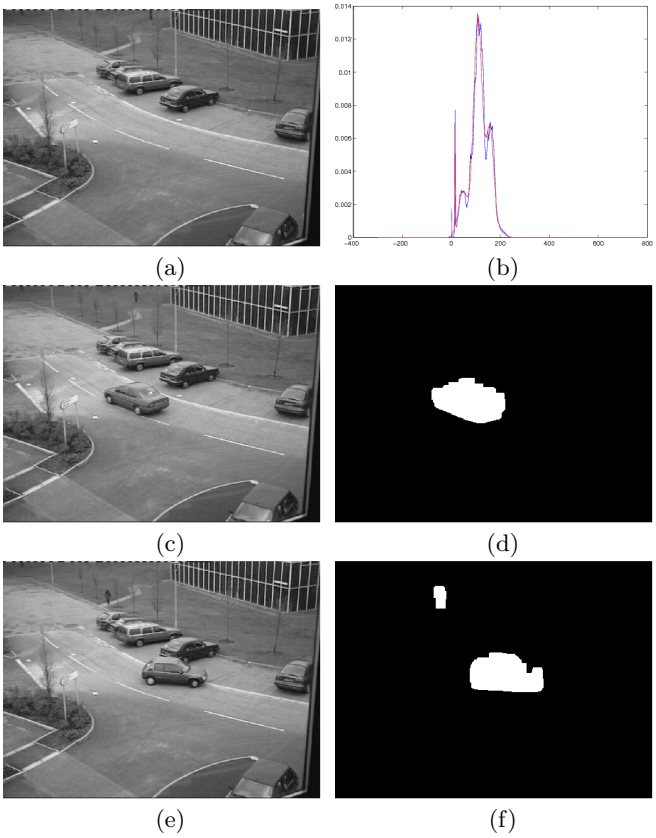

Figure 1: Outdoor scene with less motion in the background: (a) Background frame and (b) its histogram with the GMM fit. (c,e) Input frame. (d,f)Extracted foreground.

*Corresponding author

ICVGIP '10, December 12-15, 2010, Chennai, India

sian approximation of the histogram. The foreground extraction results for the scene are shown in Figs.1 (d,f) for the corresponding frames in Figs.1 (c,e).

1.1 Related Work

Various techniques exist in literature for foreground object extraction. In the methods based on direct frame differencing [8, 11], a difference between consecutive frames is computed and pixels in the difference frame above threshold are classified as foreground pixels. In the approximate median [12], the running estimate of the median is incremented by one if the input pixel is larger than the estimate, and decreased by one if smaller.

A popular framework of background scene modeling, which is also closely related to our work, uses GMM modeling for pixels [4, 9, 10, 14]. In these methods, the background model is learned over time for each pixel in the frame. The input frame pixels which are not following the model are termed as the foreground pixels. Extentions to the GMM methods also exist, such as the adaptive GMM approach [18], where the number of Gaussians assigned to each pixel are updated over time. In all the methods based on GMM for background modeling, each pixel of the frame is modeled by generally 3 to 5 Gaussians. In [3], based on color change at each pixel, reference image model is created. Then depending on the threshold calculated from the model, foreground pixels are classified. This method uses color images and reqiures training data for background modeling. On the other hand, in our method, we work with gray scale images and except the initial background frame, we do not assume any training data. The GMM model is established in a *temporal* sense by considering the intensity variation at a pixel over time. Unlike these, in our approach, we model the *spatial histogram* of background frame by mixture of Gaussians. Indeed, as stated earlier, authors in [7] show that such spatial histograms for natural scenes are usually multi modal and can be modeled as mixture of Gaussians. This spatial GMM modeling is an important distinction between our approach and the standard temporal GMM-based approaches. In the latter, a set of background frames is required for training to find the means, variances and the weights of the Gaussian components at every pixel. On the other hand, our approach needs practically no training data as we model the spatial histogram of a *single* pure background frame. Moreover, in standard GMM techniques, the typical variances of the Gaussians, which model the intensity variation for a pixel, are quite low. Hence the small illumination changes or noise can be misclassified as foreground points. In our approach the Gaussian variances are relatively large since our histogram modeling involves all pixels which span objects with considerably varying intensities. Hence misclassification due to small illumination changes or noise is considerably less. Hence, our approach is more robust to noise and illumination fluctuations.

The works in [5, 6, 1, 15, 16], primarily use optical flow for foreground extraction. However, such optical flow based methods are less accurate and sensitive to noise [17]. Moreover, as these methods use the flow computation on complete images, they are computationally quite complex. Our approach incorporates computing optical flow only for the pixels which are classified as foreground by the primary histogram modeling approach. This considerably reduces computation required for estimating optical flow vectors.

2. THE PROPOSED METHOD

We begin with selecting a pure background frame (with no foreground objects) from the video. In case it is not available it can be obtained by an temporal averaging/median operation on the first few consecutive frames (generally 20-30) of the video.

As mentioned earlier, generally the background scene involves large objects such as trees, road, floor, buildings, walls etc. Each of these scene elements have nearly uniform or smoothly varying intensities, yielding similar intensity values. We observe that normalized histograms of every such background object can be approximated by a Gaussian distribution. Hence, the histogram of the background frame can be modeled as mixture of Gaussians, each of which corresponds to a particular background object.

An example of an outdoor background image and its histogram is shown in Fig.2(a) and 2(b), respectively. Note that the histogram is multi modal and can be approximated by mixture of Gaussians.(Fig.2(d)). Also, observe that the mixture of Gaussians fits quite well to the underlying histogram.

The number of Gaussians to be fitted depends on number of major background objects. These can be found by finding number of prominent peaks in the smoothed histogram (shown in Fig. 2(c)). Finding the prominent peaks is carried out in the following manner. We convolve the background histogram with a large smoothing kernel (typically with a window size of 15) to suppress large scale variations. To further reduce the local variations, we then convolve the previous result with a smaller smoothing kernel (typically with a window size of 5). We take the first and second difference of resultant histogram. The points where first difference is nearly zero and second difference is negative is defined as a prominent peak. We can notice six prominent peaks in the smoothed histogram in Fig. 2(c). Hence histogram is modeled by six Gaussians as shown in Fig. 2(d).

2.1 Computing mixture of Gaussians for the background histogram

To estimate the mixture of Gaussian fit to the histogram, we use the expectation maximization (EM) algorithm. EM finds the maximum likelihood parameters of every Gaussian component given the number of Gaussians to be fitted and an initial estimate of the Gaussian parameters. For faster convergence of the EM algorithm, we use prominent peak positions of the histogram as the initial estimates for the means.

Let Y be the background image and Φ be the set of the parameters of K Gaussians. The probability density function (p.d.f.) for intensity of pixel i can be expressed as a weighted mixture of these K Gaussian components as

$$p(Y_i = y|\Phi) = \sum_{i=1}^{K} G(y, \mu_k, \sigma_k)c_k \quad (1)$$

where

$$G(y, \mu_k, \sigma_k) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp \frac{(y-\mu_k)^2}{2\sigma_k^2} \quad (2)$$

where μ_k, σ_k and c_k indicate the mean, standard deviation and weight of the k^{th} Gaussian component, respectively. Here we have $\sum_{i=1}^{K} c_k = 1$.

The expectation-maximization algorithm (EM) [2] is a

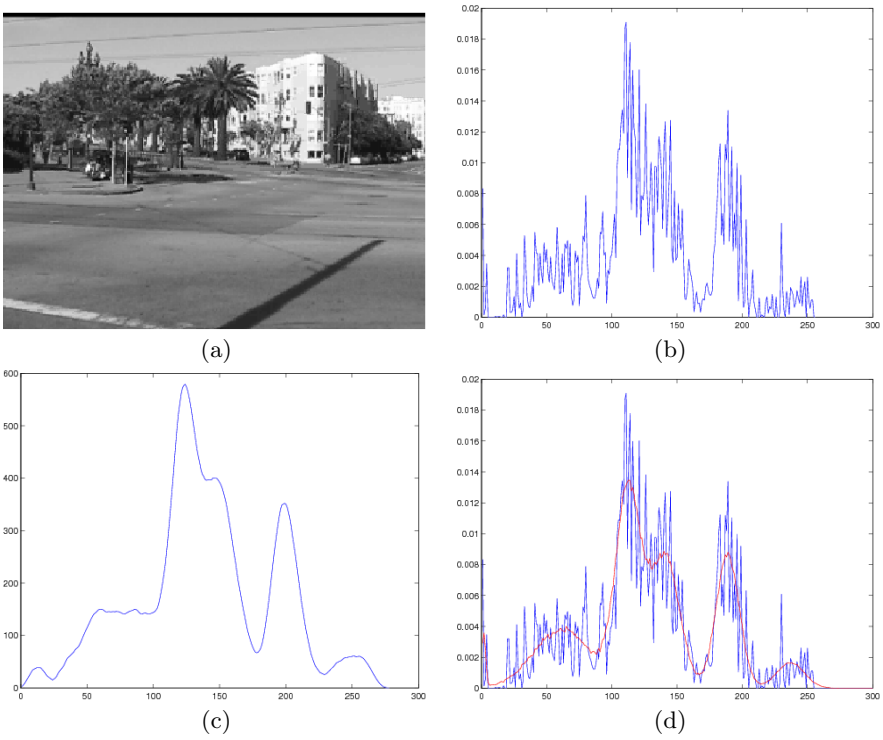

Figure 2: (a) Background frame and (b) its histogram. (c) Smoothed histogram to find the number of dominant peaks. (d) Fitting mixture of Gaussians to the histogram in (b).Blue plot indicates the actual histogram,Red plot indicates its GMM approximation.

general technique for finding maximum likelihood parameter estimates in problems with hidden data. Hidden data in this case is class assignment of each pixel of background frame. We will denote hidden data by Z. The EM tries to find maximum likelihood parameter estimates by first estimating class assignment based on current parameter estimates. The estimated complete data (observed and hidden data) are then used to estimate the parameters through maximizing the likelihood of the complete data. EM involves two steps in its operation.

- **Expectation step or E-step:** Calculate the estimate $p^{(m+1)}$, also known as soft class assignment Z, from the observed data Y and current parameter estimate $\Phi^{(m)}$. For the Gaussian case, equation of soft assignment for m^{th} iteration becomes

$$p_{ij}^{(m+1)} = \frac{G(y_i, \mu_j^{(m)}, \sigma_j^{(m)}) c_j^{(m)}}{\sum_{k=1}^{K} G(y_i, \mu_k^{(m)}, \sigma_k^{(m)}) c_k^{(m)}} \tag{3}$$

  where p_{ij} denotes the probability of i^{th} pixel getting assigned to j^{th} Gaussian component.

- **Maximization step or M-step:** Calculate the maximum likelihood parameters $\Phi^{(m+1)}$ for the current estimate of the complete data $(y, p^{(m+1)})$.

$$\mu_j^{(m+1)} = \frac{\sum_{i=1}^{n} y_i p_{ij}^{(m+1)}}{\sum_{i=1}^{n} p_{ij}^{(m+1)}} \tag{4}$$

$$(\sigma_j^{(m+1)})^2 = \frac{\sum_{i=1}^{n} (y_i - \mu_j^{(m+1)})^2 p_{ij}^{(m+1)}}{\sum_{i=1}^{n} p_{ij}^{(m+1)}} \tag{5}$$

$$c_j^{(m+1)} = \frac{\sum_{i=1}^{n} p_{ij}^{(m+1)}}{n} \tag{6}$$

where n denotes the total number of pixels in the image.

EM algorithm iterates between E-step and M-step and converges to maximum likelihood parameter estimate Φ for observed data Y. It generally takes about 10 iterations to find the maximum likelihood parameters.

2.2 Extracting foreground objects

After fitting the mixture of Gaussians to the background histogram, we form the mean image M where $M(x, y)$ denotes the mean value of the Gaussian component to which that particular pixel is assigned depending on its intensity value. Thus, the background frame is segmented, in which number of segmented regions is equal to the number of Gaussians fitted to histograms.

For the next incoming frame we compute, at every pixel, the difference in its intensity value with the corresponding pixel in the mean image. This difference is then compared with a threshold depending on which the pixel is classified as foreground.

An important aspect in foreground classification is the choice of threshold. Intuitively, since we are using the intensity *difference from the mean image* as a decisive factor in the classification, one may infer that the threshold should be related to the standard deviations (σ_ks) of the Gaussians of the model. However, since the background regions correspond to Gaussians with different σ_ks, the thresholds for pixels belonging to different background regions will differ.

(a) (b)

Figure 3: (a)Input frame. (b)Result without optical flow.

To simplify matters, we use a single threshold value as $3\sigma_a$, where σ_a is the average of the standard deviations of all the Gaussian components in the model. This threshold value was found out empirically, which gives the best result in all the cases.

If the difference value is more than the $3\sigma_a$, the pixel is classified as foreground. Given the input frame I and the mean image M, the foreground image F is the extracted as

$$\begin{aligned} F(x,y) &= 255 && \text{if} |I(x,y) - M(x,y)| \geq 3\sigma_a \\ F(x,y) &= 0 && \text{otherwise} \end{aligned} \quad (7)$$

To handle illumination variations over time, a new background frame is obtained after every 10 minutes in video. We initialize the EM iterations for this new background frame with the present values of the Gaussian parameters. If a particular pixel is classified as foreground for consecutive 300 frames, it is labeled as background and made zero. This addresses the scenario where an moving object becomes and remains stationary for a long time and hence needs to be classified as background.

3. INCORPORATING OPTICAL FLOW

If the background pixels undergo motion resulting in large intensity variations at their locations, they may be falsely classified as foreground. Such a scenario may occur in presence of tree leaves, computer displays etc. Fig. 3 shows the example of such a case where some background pixels corresponding to tree leaves are classified as foreground. To improve the result, we incorporate optical flow in our framework [13]. As motion of the background pixels is much lesser than that of the foreground pixels, flow velocity magnitude of pixels can be used to mitigate the false detections.

Estimating optical flow on complete images is often computationally expensive. However, since we are interested to reduce false classification of background pixels as foreground, we compute the flow velocities only at previously classified foreground pixels, which are much less than the total number of image pixels. Thus our approach is much more efficient than computing the flow over the complete image. Moreover, to speed up the flow velocity computation we use the method of cumulants [13].

The optical flow applies to data which is smoothly continuous and hence differentiable. Problems arise when images are not continuous but contain distinct foreground objects with crisp edges. Frequently, pixel displacements between successive images may be quite large. Hence, the algorithms, which assume small pixel motion, result in large errors in flow computation. To overcome this problem, we smoothen the input images using Gaussian blurring. As a result, the edges becomes more differentiable and it extends the effective scope of the objects. We blur the images with Gaussian kernel. In our experiments we observed that kernel of size 15x15 with variance 1 gives better results.

Given the flow vectors of the foreground pixels, points having velocity less than a *velocity threshold* are classified as background points and set to zero. The velocity threshold is kept sufficiently small to avoid mis-classification of the true foreground points.

4. RESULTS

To validate our approach, we carried out real experiments on images of various indoor and outdoor scenes. We used the videos from PETS 2001 and PETS 2006 data sets, respectively. In addition, we also experimented on some indoor and outdoor videos acquired in our campus. Quantitative analysis of the results is also given. We find out the actual number of foreground pixels present by selecting the foreground objects from the image manually. We then give number of pixels found by our approach. If number of pixels found are greater than actual number of pixels, it indicates the false detection. If number of pixels found are less than actual number of pixels, it indicates the true rejection. It can be noted that our results matches closely with the true values.

4.1 Results for indoor scenes

Table 1: Indoor scene results(Fig.4)

Frame	Actual	Detected
Fig.4 (a)	27720	27945
Fig.4 (c)	4230	4195
Fig.4 (e)	21619	21706
Fig.4 (g)	27580	27643

Indoor scenes have relatively low illumination variation, noise and background motion as compared with outdoor scenes. Due to negligible movement in the background, we achieve satisfactory results for foreground extraction using only the method described in section 2. Figs.4 (b,d,f,h) show the result of this approach on indoor scenes. It can be observed that the method is able to extract the foreground object with good accuracy and extracted objects have sharp boundaries. Results are equally good with single and multiple objects. Table 1 shows the quantitative analysis of the outputs.

4.2 Results for outdoor scenes

Table 2: Outoor scene results (Fig.1)

Frame	Actual	Detected
Fig.1 (a)	14329	14060
Fig.1 (c)	13418	12988

Fig.1 shows example of outdoor scene which have less background motion. Hence first approach based on the background histogram modeling provides satisfactory results. Fig.1 (a,c) shows the frames and Fig. 1 (b,d) shows its corresponding output. However, for outdoor scenes with moving background objects, using only this approach yields some false

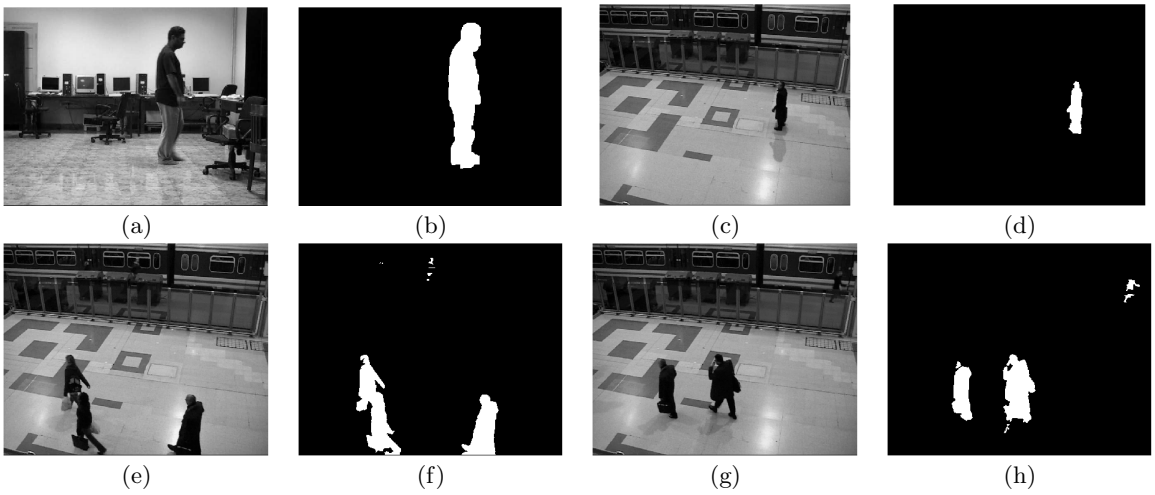

Figure 4: Indoor scene: (a,c,e,g) Input frame. (b,d,f,h) Extracted foreground.

classification of the foreground objects. Fig. 5 shows such an example. In this scenario, the false detection using only the first approach is shown in Fig.5 (b,e) for the frames in the Fig.5 (a,d) respectively. As mentioned earlier, to improve the results in such cases, the optical flow based method is incorporated. As a result, the false classification largely mitigated as can be seen in Fig. 5(c,f). Note that our complete method extracts foreground objects with good localization and yields considerably low misclassification. Table 3 indicates the corresponding quantitative analysis. It can be observed that applying optical flow makes the results more accurate. In Table 3, column 'Before' indicates the number of foreground points detected before application of optical flow and column 'After' indicates the number of foreground points detected after application of optical flow.

Table 3: Effect of Optical flow(Fig.5)

Frame	Actual	Before	After
Fig.5 (a)	7164	9854	7285
Fig.5 (d)	1782	5551	1674

We also compared our approach with standard approaches such as frame differencing [8, 11], approximate median [12] and GMM [14, 4]. The results of the comparison for an outdoor scene (Fig. 6 (a)) are shown in (Figs. 6(b-f)). It can be observed that in our output (Fig. 6(f)) the foreground is extracted with fairly low misclassification as compared to other approaches (Figs. 6(b-e)). The approaches which achieve low misclassification perform poorly in localization (e.g. Fig. 6(b,e)), and those with good localization also suffer from high false detection (e.g. Figs. 6(c,d)). Our approach fares well in this trade-off as compared to others. Table 4 shows the quantitative analysis.

We show some more comparisons with the GMM approach (Fig. 7), which is arguably the most widely used approach for background modeling. Two frames from the outdoor scene video are shown in Figs. 7(a,d). Unlike the frame, for which the results are shown in Fig. 6, these frames also have sharp illumination variations and hence provide an interesting case for comparison. The corresponding outputs for GMM and our approach are shown in Figs.7(b,e) and Figs. 7(c,f) respectively. Note that the GMM approach shows many misclassification. Some of these are possibly due to the sharp illumination variation. Our approach, on the other hand shows more robustness to illumination changes and our results show much lower misclassification as compared to the GMM approach. Quantitative results in Table 5 shows that our approach is more robust to illumination changes than the GMM approach.

Table 4: Comparisons with different methods(Fig.6)

Frame	Method	No. of foreground points
Fig.6 (a)	Actual	**11582**
Fig.6 (b)	Optical flow	16456
Fig.6 (c)	Frame difference	15116
Fig.6 (d)	Approximate median	15228
Fig.6 (e)	GMM	601
Fig.6 (f)	Our Method	**11748**

Table 5: Handeling of sharp illumination changes(Fig.7)

Frame	Actual	GMM	Our method
Fig.7 (a)	14150	28693	12582
Fig.7 (d)	8056	15987	7547

5. CONCLUSION

In this paper we proposed a novel histogram based approach for foreground object extraction for indoor and outdoor scenes. In our approach we modeled the histogram of the background frame by the Mixture of Gaussian and derive a mean background image using this model. The foreground objects were classified according to the intensity deviation with this mean background image. To adapt to the illumination variations with time, we update the histogram model at regular intervals. To avoid false detection due to the motion of background pixels, we incorporate optical flow in our framework in an efficient manner. Our results show that this

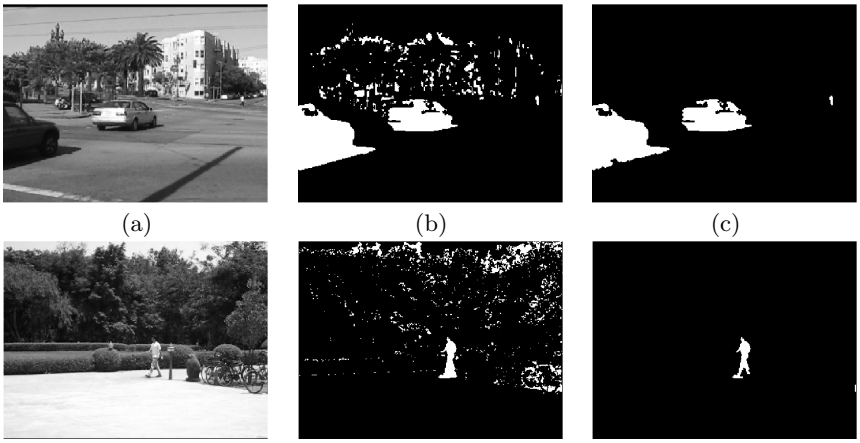

Figure 5: Outdoor scene with significant background movement: (a)Input frame. (b)Result without optical flow. (c)Result with optical flow.

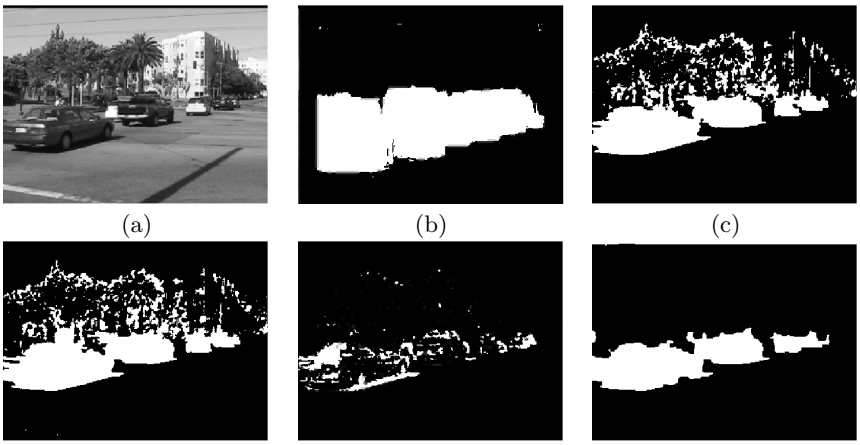

Figure 6: Comparison with standard approaches: (a) Input frame. (b)Optical flow [13]. (c)Frame difference. [8] (d)Approximate median.[12] (e) GMM. [4](f) Our method.

approach is able to extract foreground with good fidelity in indoor as well as outdoor scenes.

6. REFERENCES

[1] A. G. Bors and I. Pitas. Optical flow estimation and moving object segmentation based on rbf network. *IEEE Trans. On Image Processing*, 7(5):693–702, 1998.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society (Series B)*, 39(1):1–38, 1977.

[3] Dongpyo;Woontack. A background subtraction for a vision-based user interface. *Proceedings of ICICS-PCM*, 1:263–267, 2003.

[4] N. Friedman and S. Russell. Image segmentation in video sequences: a probabilistic approach, 1997. Proc. 13th Conf. Uncertainty Artificial Intelligence.

[5] T. Hirai. Detection of small moving objects by optical flow. *11th International Conference on Pattern Recognition*, 2:474–478, 1992.

[6] Y. Huang. Optical flow field segmentation and motion estimation using a robust genetic partitioning algorithm. *IEEE Trans. On Pattern Analysis and Machine Intelligence*, 17(12):1177–1190, 1995.

[7] Z. K. Huang and D. H. Liu. Unsupervised image segmentation using em algorithm by histogram. *Springer Berlin / Heidelberg*, 4681:1275–1282, 2007.

[8] R. Jain and H. Nagel. On the analysis of accumulative difference pictures from image sequences of real world scenes. 1(2):206–213, 1979. IEEE Trans. Pattern Analysis and Machine Intelligence.

[9] K. P. Karmann, A. Brandt, and R. Gerl. Using adaptive tracking to classify and monitor activities in a site, 1990. Time Varying Image Processing and Moving Object Recognition.

[10] D. S. Lee. Effective gaussian mixture learning for video background subtraction. *IEEE Trans. Pattern analysis and machine intelligence*, 27(5):827–832, 2005.

[11] L. Li and M. Leung. Integrating intensity and texture

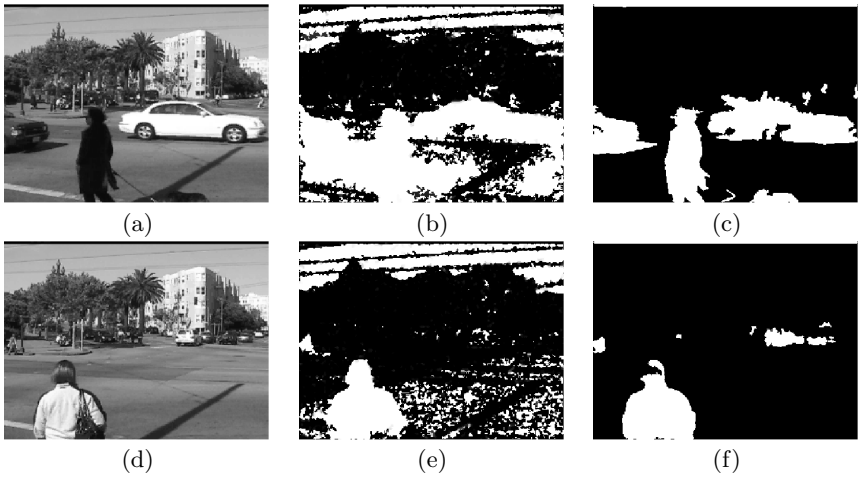

Figure 7: More comparisons with the GMM approach: (a,c) Input frames. (b,e) Corresponding outputs for the GMM approach and (c,f) for our method.

differences for robust change detection. *IEEE Trans. Image Processing*, 11(1):105–112, 2002.

[12] N. McFarlane and C. Schofield. Segmentation and tracking of piglets in images. *Machine Vision and Applications*, 8(3):187–193, 1995.

[13] M. Peura and H. Hohti. Motion vectors in weather radar images, 2004. Proceedings of the 7th International Winds Workshop, Helsinki, Finland.

[14] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. *Proc. IEEE CS Conf. Computer Vision and Pattern Recognition*, 2(1):246–252, 1999.

[15] Y. C. Y. and S. Oe. A new gradient-based optical flow method and its application to motion segmentation. *26th Annual Conference of the IEEE Industrial Electronics Society*, 2:1225–1230, 2000.

[16] H. Yalcin, M. J. Black, and R. Fablet. The dense estimation of motion and appearance in layers. *Conf. on Computer Vision and Pattern Recognition Workshop (CVPRW)*, 11:165, 2004.

[17] D. Zhou and H. Zhang. Modified gmm background modeling and optical flow for detection of moving objects. *IEEE International Conf. on Systems, Man and Cybernetics*, 3(5):2224–2229, 2005.

[18] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. *Proc. International Conf. on Pattern Recognition*, 2(1):28–31, 2004.