Performance Evaluation of Various Foreground Extraction Algorithms for Object detection in Visual Surveillance

Sudheer Reddy Bandi [1], A.Varadharajan [2], M.Masthan [3]
[1, 2, 3] Assistant Professor, Dept of Information Technology, Tagore Engineering College, Chennai,

Abstract
Detecting moving objects in video sequence with a lot of moving vehicles and other difficult conditions is a fundamental and difficult task in many computer vision applications. A common approach is based on background subtraction, which identifies moving objects from the input video frames that differs significantly from the background model. Numerous approaches to this problem differs in the type of background modeling technique and the procedure to update the model. In this paper, we have analysed three different background modeling techniques namely median, change detection mask and histogram based modeling technique and two background subtraction algorithms namely frame difference and approximate median. For all possible combinations of algorithms on various test videos we compared the efficiency and found that background modeling using median value and background subtraction using frame difference is very robust and efficient.

Keywords—Background Subtraction, Visual Surveillance, Threshold Value, Camera jitter, Foreground Detection, Change Detection Method, Mean.

I. Introduction
Video surveillance or visual surveillance is a fast growing field with numerous applications including car and pedestrian traffic monitoring, human activity surveillance for unusual activity detection, people counting, and other commercial applications. In all these applications extracting moving object from the video sequence is a key operation. Typically, the usual approach for extracting the moving object from the background scene is the background subtraction which provides the complete feature data from the current image. Fundamentally, the objective of background subtraction algorithm is to identify interesting areas of a scene for subsequent analysis. “Interesting” usually has a straight forward definition: objects in the scene that move. Since background subtraction is often the first step in many computer vision applications, it is important that the extracted foreground pixels accurately correspond to the moving objects of interest. Even though many background subtraction algorithms have been proposed in the literature, the problem of identifying moving objects in complex environment is still far from being completely solved. The success of these techniques has led to the growth of the visual surveillance industry, forming the foundation for tracking, Object recognition, pose reconstruction, motion detection and action recognition.

In the context of the background subtraction, we have two distinct processes that work in a closed loop: they are background modeling and foreground detection. The simplest way to model the background is to acquire a background image which doesn’t include any moving object. Unfortunately, background modeling is hard and time consuming and is not well solved yet. In foreground detection, a decision is made as to whether a new intensity fits the background model; the resulting change label field is fed back into background modeling so that no foreground intensities contaminate the background model. Foreground objects are extracted by fusing the detection results from both stationary and motion points. The videos which are used for testing the algorithms are given in Fig. 1.

We make the fundamental assumption that the background will remain stationary. This necessitates that the camera be fixed and that lighting does not change suddenly. Our goal in this paper is to evaluate the performance of different background modeling and different background subtraction algorithms using criteria such as the quality of the background subtraction and speed of algorithm.

The rest of the paper is organized as follows: Section II summarizes the previous work done by various researchers. Background modeling techniques are given in section III. For the background model obtained in section III, background subtraction algorithms are given in section IV. Experimental results are given in section V. Finally we conclude our paper in section VI.

Fig.1 Videos to test the Algorithm Efficiency.

II. LITERATURE REVIEW

The research work on background subtraction and object detection is vast and we have limited the review to major themes. The earliest approach to background subtraction originated in the late 70s with the work of Jain and Nagel[1], who used frame differencing for the detection of moving objects. Current areas of research includes background subtraction using unstable camera where Jodoin et al. [2] present a method using the average background motion to dynamically filter out the motion from the current frame, whereas Sheikh et al. [3] proposed an approach to extract the background elements using the trajectories of salient features, leaving only the foreground elements. Although background subtraction methods provide fairly good results, they still have limitations for use in practical applications. The various critical situations of background subtraction are camera jitter, illumination changes, objects being introduced or removed from the scene. Changes in scene lighting can cause problems for many background methods. Ridder et al. [4] modeled each pixel with a kalman filter which made their system more robust to lighting changes in the scene. A robust background modeling is to represent each pixel of the background image over time by a mixture of Gaussians. This approach was first proposed by Stauffer and Grimson [5], [6], and became a standard background updating procedure for comparison. In their background model, the distribution of recently observed value of each pixel in the scene is characterized by a mixture of several Gaussians. In [7] a binary classification technique is used to detect foreground regions by a maximum likelihood method. Since in these techniques the probability density function of the background is estimated, the model accuracy is bounded to the accuracy of the estimated probability.

III. BACKGROUND MODELING TECHNIQUES

A. Change Detection Method

Let $I_i(x,y)$ represent a sequence including N images, i represents the frame index ranging from 1~N. Equation (1) is used to compute the value of CDM which is given as follows:

$$CDM_i(x, y) = d, \quad \text{if } d >= T$$
$$CDM_i(x, y) = 0, \quad \text{if } d < T$$
$$d = | I_{i+1}(x,y) - I_i(x, y) | \qquad (1)$$

Where T is the threshold value determined through experiments. CDM_i (x, y) represents the value of the brightness change in pixel position along the time axis. The estimated value in pixel(x,y) is estimated by the value of median frame of the longest labeled section in the same pixel(x,y). A basic approach to initialize background (BCK) is to use the mean or the median of a number of observed frames as given in equation (2).

$$BCK^n (x, y) = mean_t\, I^t (x, y)$$
$$BCK^n (x, y) = med_t\, I^t (x, y) \qquad (2)$$

The background model for three input videos is given in Fig. 2.

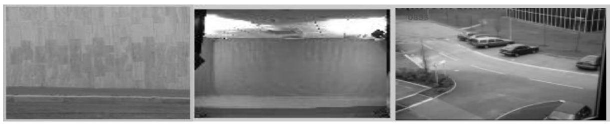


Fig.2 Background model for three videos using Change Detection Method.

B. Median Value Method

An estimate of the background image can be obtained by computing the median value for each pixel in the whole sequence. Let B(x, y) is the background value for a pixel location(x, y), med represents the median value, [a(x, y, t),….a(x, y, t+n)] represents a sequence of frames.

$$B(x, y) = med[a(x, y, t),\ldots.a(x, y, t+n)] \qquad (3)$$

For a static camera, the median brightness value of the pixel(x, y) should correspond to the background value in that pixel position, hence providing good background estimates. The background modeled using this approach is given in Fig. 3.

Fig.3 Background model for three videos using Median Value Method.

C. *Histogram based Background Modeling*

In the histogram based technique, illumination changes, camera jitter greatly affects the performance of the algorithm. This algorithm is based on the maximum intensity value of pixels for the video sequence of given size. The algorithm for the background modeling can be given as follows. The background images of the test videos are shown in Fig. 5.

```
for i=1→row
  for j=1→column
     for k=1→no. of frames
        x(k)=pixel(i,j,1,k);
     end
        bgmodel(i,j)=max(x);
  end
end
```

Fig.4 Algorithm for Histogram based Background Modeling.

Where bgmodel represents the background model, x represents the array to store the pixel values.

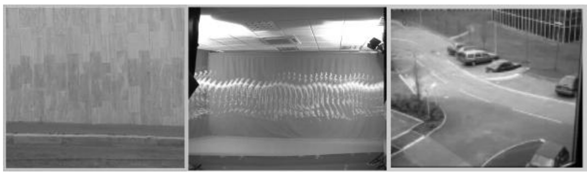


Fig.5 Background images for the test videos.

IV. FOREGROUND EXTRACTION TECHNIQUES

A. *Approximate Median Method*

Approximate median method uses a recursive technique for estimating a background model. Each pixel in the background model is compared to the corresponding pixel in the current frame, to be incremented by one if the new pixel is larger than the background pixel or decremented by one if smaller. A pixel in the background model effectively converges to a value where half of the incoming pixels are larger than and half are smaller than its value. This value is known as the median. The Approximate median method has been selected, for it handles slow movements, which are often the case in our environment, better than the Frame differencing Method. The Approximate median foreground detection compares the current video frame to the background model in equation (4), and identifies the foreground pixels. For this it checks if the current pixel bw(x, y) is significantly different from the modeled background pixel bg(x, y).

$$|bw(x, y) - bg(x, y)| > T \qquad (4)$$

A simplified pixel-wise implementation of the approximate median background subtraction method in pseudo-code is given in Fig. 6 where T represents the threshold value.

```
/* Adjust background model */
if (bw > bg) then bg = bg + 1 ;
else if (bw < bg) then bg = bg - 1 ;

/* Determenine foreground */
if (abs(bw - bg) > T) then fg = 1 ;
else fg = 0;
```

Fig.6 Approximate median background subtraction

B. *Frame Difference Method*

The second algorithm is the frame difference algorithm. The frame difference algorithm compares the frame with the frame before, therefore allowing for scene changes and updates.

$$|f_i - f_{i-1}| > T \qquad (5)$$

F is the frame, i is the frame number, T is the threshold value. This allows for slow movement update as the scene changes. A major flaw of this method is that for objects with uniformly distributed intensity values, the pixels are interpreted as part of the background. Another problem is that objects must be continuously moving. This method does have two major advantages. One obvious advantage is the modest computational load. Another is that the background model is highly adaptive. Since the background is based solely on the previous frame, it can adapt to changes in the background faster than any other method. A challenge with this method is determining the threshold value.

V. EXPERIMENTAL RESULTS

In order to evaluate the performance of our approach, some experiments have been carried out. In our experiments our approach is performed for three video sequences. The various parameters of the videos which are used to compare the techniques are listed in Table I. Two gait videos with different parameters and a car parking video are used in our experiments.

TABLE I
PARAMETERS OF TEST VIDEOS

PARAMETRES	**VIDEO**		
	Gait video1	**Gait video2**	**Car parking**
Length	2 sec	3 sec	20 sec
Frame width	180	720	352
Frame height	144	576	288
Data rate	15552 kbps	248868 kbps	200 kbps
Total bit rate	15552 kbps	248932 kbps	200 kbps
Frame rate	25 fps	25 fps	25 fps

We simulated the experiment on a pc with core 2 duo processor, 1 GB main memory, and with a speed of 1.73 GHz. As a number of different background algorithms were chosen to be compared; the algorithms would have to be in the same code format to ensure the speed was not being influenced by the code base of the platform it was running on. Here we have chosen MATLAB to run our experiments. The first step in our paper is to find the processing time for three videos in finding the background model using the change detection method, median value method and histogram based background modeling methods. The processing times are given in seconds are listed in Table II.

TABLE II
PROCESSING TIME FOR BACKGROUND MODELING TECHNIQUES

Background model	**VIDEO**		
	Gait video1	**Gait video2**	**Car parking**
Change Detection Method	13.9098	161.6029	367.4683
Median value	1.2720	26.1600	11.0673
Histogram based	0.4282	3.7902	4.7036

The threshold value for the change detection method is 156. In the next step of our approach we go for the foreground detection technique in which we can do the object segmentation and silhouette extraction. The techniques which we discussed in our method are frame difference method and approximate median method. The threshold value taken for the frame difference approach and approximate median method is 40. Threshold value is taken based on the empirical experiments. The processing time to extract the foreground images from the background model for the three videos using the combination of techniques is listed in Table III.

TABLE III
PROCESSING TIME FOR BACKGROUND SUBTRACTION TECHNIQUES

Background model	**Foreground extraction**	**VIDEO**		
		Gait video1	**Gait video2**	**Car parking**
Change Detection Method	Frame difference	13.8100	174.7984	431.7731
	Approximate median	14.0418	181.4313	438.1234
Median value	Frame difference	4.5001	41.3078	47.2300
	Approximate median	4.3612	41.7556	47.9200
Histogram based	Frame difference	3.8555	21.1704	43.1276
	Approximate median	3.9600	20.6270	41.7200

VI. CONCLUSIONS

In this paper, we have tested the background modeling techniques on various videos. The various pros and cons of each algorithm are discussed. We surveyed a number of background subtraction algorithms in the literature. We analyze them based on how they differ in preprocessing, background modeling, foreground detection. More research, however, is needed to improve robustness against environment noise, sudden change of illumination, and to provide a balance between fast adaptation and robust modeling. The experimental results show that median value based background modeling is robust and computational efficiency. Future work on this topic will follow one direction mainly i.e., it may suffer from errors of scene changes in part of the background. This problem can be coped with by upgrading the background model in the main process as Chein proposed[8].

REFERENCES

[1] R. Jain and H. Nagel. On the analysis of accumulative difference pictures from image sequences of real world scenes. IEEE TPAMI, 1979.

[2] P. Jodoin, J. Konrad, V. Saligrama, and V. Veilleux-Gaboury. Motion detection with an unstable camera. In Proc. IEEE Signal Intˇ Sl Conf. Image Process, pages 229–232, 2008.

[3] Y. Sheikh, O. Javed, and T. Kanade. Background subtraction for freely moving cameras. In IEEE International Conference on Computer Vision, 2009.

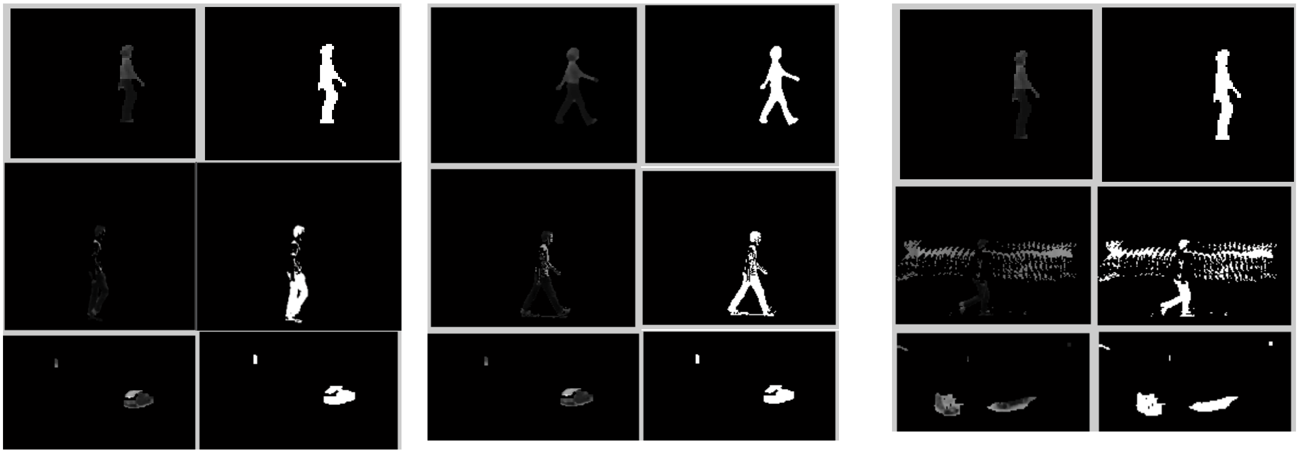
[4] Christof Ridder, Olaf Munkelt, and Harald Kirchner. “Adaptive Background Estimation and Foreground Detection using Kalman-Filtering,” Proceedings of International Conference on recent Advances in Mechatronics, ICRAM’95, UNESCO Chair on Mechatronics, 193-199,1995.

[5] C. Stauffer and W. E. L. Grimson, “Adaptive background mixture models for real-time tracking,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognit.*, 1999, vol. 2, pp. 246–252.

[6] C. Stauffer and W. E. L. Grimson, “Learning patterns of activity using real-time tracking,” *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[7] Mittal, A., Paragios, N.: Motion-based background subtraction using adaptive kernel density estimation. Proceedings of CVPR 2 (2004) 302-309.

[8] S. Chien, S. Ma, and L. Chen, “Efficient Moving Object Segmentation Algorithm using Background Registration Technique,” IEEE Tran. CSVT, vol. 12, no. 7, pp. 577-586, 2002.



(a) (b) (c)

Fig.7 (a) Foreground image and Silhouette image using Change Detection Method. (b) Foreground image and Silhouette image using Median value Background Modeling method. (c) Foreground image and Silhouette image using Histogram based Background Modeling Method.