# Motion Detection: A Survey

Ashish Kumar Sahu[#1], Abha Choubey[*2]

[#] *Department of Computer Science & Engineering,*
*Faculty of Engineering & Technology, SSTC, SSGI*
*Bhilai, India*

[1] aashish.sahu07@gmail.com

[*] *Department of Computer Science & Engineering,*
*Faculty of Engineering & Technology, SSTC, SSGI*
*Bhilai, India*

[2] abha.is.shukla@gmail.com

***Abstract*— Visual surveillance systems have gained a lot of interest in the last few years. Real-time segmentation of moving regions in image sequences is a fundamental step in many vision systems including human-machine interface, very low-bandwidth telecommunications and automated visual surveillance. A distinctive method is background subtraction, where each video frame is compared against a background model or reference. Pixels in the current frame that deviate significantly from the previous frame are considered to be moving objects. These "foreground" pixels are further processed for object localization and tracking. Background subtraction is often the first step in many computer vision applications. Several background models have been introduced to deal with different problems. At present methods used in moving object detection are mainly the frame subtraction method, the optical flow method and the background subtraction method. Even though many background subtraction techniques have been proposed, they are typically presented as parts of a larger computer vision application.**

***Keywords*— Surveillance, background subtraction, tracking**

## I. INTRODUCTION

Human body motion analysis has been an interesting research for its various applications, such as physical performance, evaluation, medical diagnostics, virtual reality, and human–machine interface. In general, three aspects of research directions are considered in the analysis of human body motion: tracking and estimating motion parameters, analyzing of the human body structure, and recognizing of motion activities. At present methods used in moving object detection are mainly the frame subtraction method, the background subtraction method and the optical flow method. The presence of moving objects determined by calculating the difference between two consecutive images, in the frame subtraction method. Its calculation is simple and easy to implement. The background subtraction method is to use the difference of the current image and background image to detect moving objects, with simple algorithm, but very sensitive to the changes in the external environment and has poor anti-interference ability. Optical flow method is to calculate the image optical flow field, and do clustering processing according to the optical flow distribution characteristics of image. This method can get the complete movement information and detect the moving object from the background better, however, a large quantity of calculation, sensitivity to noise, poor anti-noise performance, make it not suitable for real-time demanding occasions. Any motion detection system based on background subtraction needs to handle a number of critical situations such as:

1. Noise image, due to a poor quality image source;
2. Gradual variations of the lighting conditions in the scene;
3.Small movements of non-static objects such as tree branches and bushes blowing in the wind;
4. Undeviating variations of the objects in the scene, such as cars that park (or depart after a long period);
5. Sudden changes in the light conditions, (e.g. sudden raining), or the presence of a light switch (the change from daylight to non-natural lights in the evening);
6. Movements of objects in the background that leave parts of it different from the background model;

## II. RELATED WORK

. The importance and popularity of human motion analysis has led to several previous surveys. The major purpose of background subtraction is to generate a reliable background model and thus significantly improve the detection of moving objects. Some state-of the-art background subtraction methods include simple background subtraction (SBS), running average (RA), $\sum - \triangle$ estimation (SDE),
Multiple $\sum - \triangle$ estimation (MSDE), simple statistical difference (SSD), RA with discrete cosine transform (DCT) domain, and temporal median filter (TMF).

## III. BACKGROUND SUBTRACTION ALGORITHMS

### *A. Background Subtraction*

The basic scheme of background subtraction is to subtract the image from a reference image that models the background scene. Typically, the basic steps of the algorithm are as follows:

_ Background modeling constructs a reference image representing the background.

_ Threshold selection determines appropriate threshold values used in the subtraction operation to obtain a desired detection rate.

_ Subtraction operation or pixel classification classifies the type of a given pixel, i.e., the pixel is the part of background (including ordinary background and shaded background), or it is a moving object.

© 2013 IJAIR. ALL RIGHTS RESERVED

Even though there exist a myriad of background subtraction algorithms in the literature, most of them follow a simple flow diagram shown in Figure 1. The four major steps in a background subtraction algorithm are preprocessing, background modeling, foreground detection, and data validation. Preprocessing consists of a collection of simple image processing tasks that change the raw input video into a format that can be processed by subsequent steps. Background modeling uses the new video frame to calculate and update a background model. This background model provides a statistical description of the entire background scene. Foreground detection then identi_es pixels in the video frame that cannot be adequately explained by the background model, and outputs them as a binary candidate foreground mask. Finally, data validation examines the candidate mask, eliminates those pixels that do not correspond to actual moving objects, and outputs the final foreground mask.

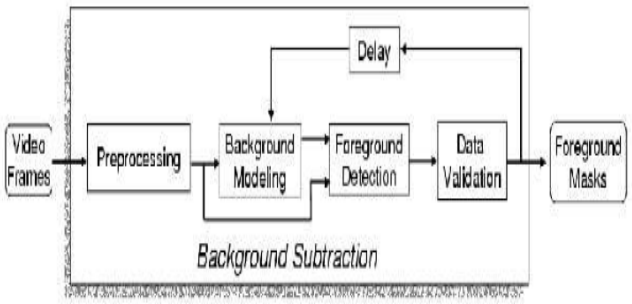


Figure 1. *Flow diagram of a generic background subtraction algorithm.*

### *B. Preprocessing*

Another key issue in preprocessing is the data format used by the particular background subtraction algorithm. Most of the algorithms handle luminance intensity, which is one scalar value per each pixel. However, color image, in either RGB or HSV color space, is becoming more popular in the background subtraction literature. In addition to color, pixel-based image features such as spatial and temporal derivatives are sometimes used to incorporate edges and motion information. The main drawback of adding color or derived features in background modeling is the extra complexity for model parameter estimation. The increase in complexity is often signi_cant as most background modeling techniques maintain an independent model for each pixel.

### *C. Background Modelling*

Background modeling is at the heart of any background subtraction algorithm. Much research has been devoted to develop a background model that is robust against environmental changes in the background, but sensitive enough to identify all moving objects of interest. We classify background modeling techniques into two broad categories - non-recursive and recursive. They are described in the following subsections. We focus only on highly-adaptive techniques, and exclude those that require significant resource for initialization.

1) *Non- Recursive Techniques*: A non-recursive technique uses a sliding-window approach for background estimation. It stores a buffer of the previous L video frames, and estimates the background image based on the temporal variation of each pixel within the buffer. Non-recursive techniques are highly adaptive as they do not depend on the history beyond those frames stored in the buffer. On the other hand, the storage requirement can be significant if a large buffer is needed to cope with slow-moving traffic. Given a fixed-size buffer, this problem can be partially alleviated by storing the video frames at a lower frame-rate r. Some of the commonly-used non-recursive techniques are frame differencing, median filter, linear predictive filter, non parametric model.

2) *Recursive Techniques*: Recursive techniques do not maintain a buffer for background estimation. Instead, they recursively update a single background model based on each input frame. As a result, input frames from distant past could have an effect on the current background model. Compared with non-recursive techniques, recursive techniques require less storage, but any error in the background model can linger for a much longer period of time. Most schemes include exponential weighting to discount the past, and incorporate positive decision feedback to use only background pixels for updating. Some of the representative recursive techniques are Approximated median filter, Kalman filter, Mixture of Gaussians(MoG).

3) *Foreground Detection:* Foreground detection compares the input video frame with the background model, and identifies candidate foreground pixels from the input frame. Except for the non-parametric model and the MoG model, all the techniques introduced in Section *C* use a single image as their background models. The most commonly- used approach for foreground detection is to check whether the input pixel is significantly different from the corresponding background estimate:

$$|I_t(x,y) - B_t(x,y)| > T$$

Another popular foreground detection scheme is to threshold based on the normalized statistics:

$$\frac{|I_t(x,y) - B_t(x,y) - \mu_d|}{\sigma_d} > T_s,$$

where µd and $\sigma_d$ are the mean and the standard deviation of It(x; y) - Bt(x; y) for all spatial locations (x; y).
Most schemes determine the foreground threshold T or Ts experimentally. Ideally, the threshold should be a function of the spatial location (x; y). For example, the threshold should be smaller for regions with low contrast. Stauffer and Grimson use the relative difference rather than absolute difference to emphasize the contrast in dark areas such as shadow:

© 2013 IJAIR. ALL RIGHTS RESERVED

$$\frac{|I_t(x,y) - B_t(x,y)|}{B_t(x,y)} > T_c$$

Nevertheless, this technique cannot be used to enhance contrast in bright images such as an outdoor scene under heavy fog. Another approach to introduce spatial variability is to use two thresholds with hysteresis. The basic idea is to first identify “strong" foreground pixels whose absolute differences with the background estimates exceeded a large threshold. Then, foreground regions are grown from strong foreground pixels by including neighboring pixels with absolute differences larger than a smaller threshold. The region growing can be performed by using a two-pass, connected-component grouping algorithm.

### *D. Data Validation*

We define data validation as the process of improving the candidate foreground mask based on information obtained from outside the background model. All the background models in Section 2.2 have three main limitations: first, they ignore any correlation between neighboring pixels; second, the rate of adaption may not match the moving speed of the foreground objects; and third, non-stationary pixels from moving leaves or shadow cast by moving objects are easily mistaken as true foreground objects. The first problem typically results in small false-positive or false-negative regions distributed randomly across the candidate mask. The most common approach is to combine morphological filtering and connected component grouping to eliminate these regions. Applying morphological filtering on foreground masks eliminates isolated foreground pixels and merges nearby disconnected foreground regions. Many applications assume that all moving objects of interest must be larger than a certain size. Connected-component grouping can then be used to identify all connected foreground regions, and eliminates those that are too small to correspond to real moving objects.
When the background model adapts at a slower rate than the foreground scene, large areas of false foreground, commonly known as \ghosts", often occur. If the background model adapts too fast, it will fail to identify the portion of a foreground object that has corrupted the background model. A simple approach to alleviate these problems is to use multiple background models running at different adaptation rates, and periodically cross-validate between different models to improve performance. Sophisticated vision techniques can also be used to validate foreground detection. Computing optical flow for candidate foreground regions can eliminate ghost objects as they have no motion. Color segmentation can be used to grow foreground regions by assuming similar color composition throughout the entire object. If multiple cameras are available to capture the same scene at different angles, disparity information between cameras can be used to estimate depth. Depth information is useful as foreground objects are closer to the camera than background. The moving-leaves problem can be addressed by using sophisticated background modeling techniques like MoG and applying morphological filtering for cleanup. On the other hand, suppressing moving shadow is much more problematic, especially for luminance-only video.

## IV. EXPERIMENTAL RESULTS

In this section, we compare the performance of a number of popular background modeling techniques. Table 1 lists all the techniques being tested, in the increasing order of complexity. We fix the buffer size for the median filter and the number of components for MoG so that they have comparable storage requirements and computational complexity. In the performance evaluation, we will vary the test parameters to show the performance of each algorithm at different operation points. In this paper, we apply the background models to luminance sequences only. For preprocessing, we first apply a three-frame temporal erosion to the test sequence, that is we replace It with the minimum of $I_{t-1}$, $I_t$, and $I_{t+1}$. This step can reduce temporal camera noise and mitigate the effect of snowfall present in one of our test sequences. Then, a 3_3 spatial Gaussian filter is used to reduce spatial camera noise. Simple thresholding with normalized statistics is used for foreground detection, except for MoG which has a separate foreground detection process as described in Section 2.2.2. No data validation is performed to postprocess the output foreground masks.

| Schemes | Fixed parameters | Test parameters |
|---|---|---|
| Frame differencing (FD) | None | Foreground threshold $T_s$ |
| Approximated median filter (AMF) | None | Foreground threshold $T_s$ |
| Kalman filter (KF) | None | Adaptation rates $\alpha_1, \alpha_2$ |
| | | Foreground threshold $T_s$ |
| Median filter (MF) | Buffer size $L = 9$ | Buffer sampling rate $r$ |
| | | Foreground threshold $T_s$ |
| Mixture of Gaussian (MoG) | Number of components $K = 3$ | Adaptation rate $\alpha$ |
| | Initial variance $\sigma_o^2 = 36$ | Weight threshold $\Gamma$ |
| | Initial weight $\omega_o = 0.1$ | Deviation threshold $D$ |

Table 1. *Background modeling schemes tested and their parameters.*

### A. *Test Sequences*

We have selected four publicly-available urban traffic video sequences from the website maintained by KOGS/- IAKS Universitaet Karlsruhey. A sample frame from each sequence is shown in the first row of Figure 2. The first sequence is called \Bright", which is 1500 frames long showing a traffic intersection in bright daylight. This sequence contains some “stop-and-go" traffic- vehicles come to a stop in front of a red-light and start moving once the light turns green. The second sequence is called “Fog", which is 300 frames long showing

© 2013 IJAIR. ALL RIGHTS RESERVED

the same traffic intersection in heavy fog. The third sequence "Snow" is also 300 frames long and shows the intersection while snowing. Fog and Snow were originally in color; we have first converted them into luminance and discarded the chroma channels. The first three sequences all have low to moderate traffic. They are selected to demonstrate the performance of background subtraction algorithms under different weather conditions. The last sequence
"Busy" is 300 frames long. It shows a busy intersection with the majority of the vehicle traffic owing from the top left corner to the right side. A quarter of the intersection is under a shadow of a building. A number of pedestrians are walking on the sidewalk on the left. The camera appears to be behind a window and the base of the window is partially reflected at the lower right corner of the video frames. This sequence is selected because of the large variation in the sizes of the moving objects and the presence of the shadow of a large building.

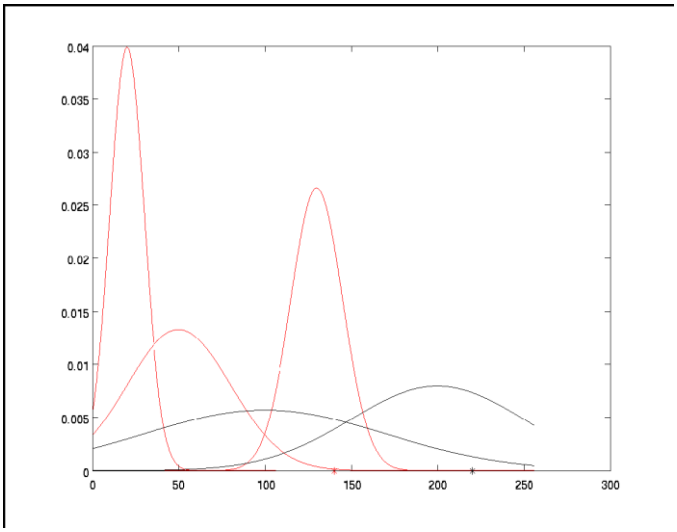


Figure: history of a pixel using Background Model Estimation

After background model estimation red distributions become the background model and black distributions are considered to be foreground

B. *Evaluation*

In order to have a quantitative evaluation of the performance, we have selected ten frames at regular intervals from each test sequence, and manually highlighted all the moving objects in them. These "ground-truth" frames are selected from the latter part of each of the test sequences to minimize the effect of the initial adaptation of the algorithms. In the manual annotation, we highlight only the pixels belonging to vehicles and pedestrians that are actually moving at that frame. Since we do not use any shadow suppression scheme in our comparison, we also include those shadow pixels cast by moving objects. The ground-truth frames showing only the moving objects are shown in the second row of Figure 2. We use two information retrieval measurements, recall and precision, to quantify how well each algorithm matches the ground-truth. They are defined in our context as follows:

$$\text{Recall} = \frac{\text{Number of foreground pixels correctly identified by the algorithm}}{\text{Number of foreground pixels in ground-truth}}$$

$$\text{Precision} = \frac{\text{Number of foreground pixels correctly identified by the algorithm}}{\text{Number of foreground pixels detected by the algorithm}}$$

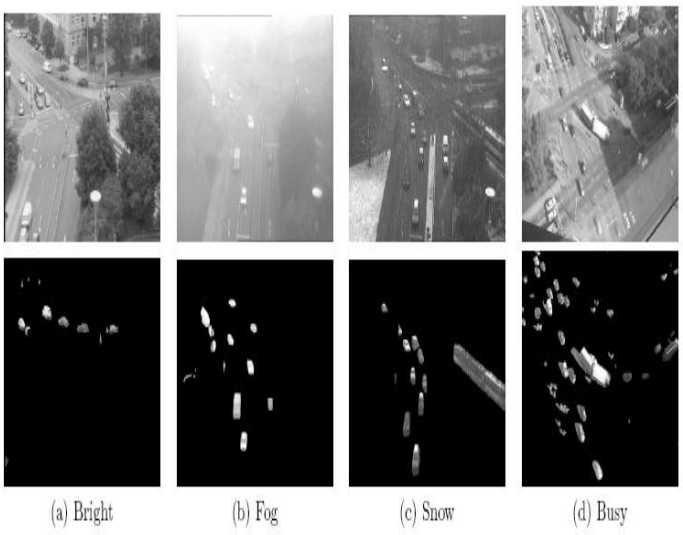


Figure 2. *Sample frames and the corresponding ground-truth frames from the four test sequences: Bright, Fog, Snow, and Busy.*

Recall and precision values are both within the range of 0 and 1. When applied to the entire sequence, the recall and precision reported are averages over all the measured frames. Typically, there is a trade-off between recall and precision - recall usually increases with the number of foreground pixels detected, which in turn may lead to a decrease in precision. A good background algorithm should attain as high a recall value as possible without sacrificing precision. In our experiments, we vary the parameters in each algorithm to obtain different recall-precision operating points. The resulting graphs for the four test sequences are shown in Figures 3(a) to (d). There are four plots for each sequence. The first plot corresponds to the two simplest algorithms, FD and AMF. The curves are generated by varying the foreground threshold Ts. The second plot corresponds to MF at buffer sampling rates of 1, 5, and 10 frames per second. The curves are also generated by varying Ts. The third plot corresponds to MoG at different combinations of α and $\Gamma$. The curves are generated by varying the deviation threshold D. Compared with the previous two schemes, there are far fewer actual data points on the MoG curves. The reason is that D directly affects the future states of MoG. To generate a single data point, one needs to run the algorithm through the entire video sequence at a particular value of D. On the other hand, Ts has no effect on the internal states of FD, AMF, or MF. To generate the results at different values of Ts, it is sufficient to

© 2013 IJAIR. ALL RIGHTS RESERVED

run the algorithm once, save the raw difference frames, and then threshold them with different values of Ts. The final plot contains results from KF at different values of $\alpha_1$ and $\alpha_2$. The curves are also generated by varying Ts. Note that in the cases when α1 and α2 are equal, the feedback information is not used. The update equation in (2) reduces to a leaky moving average with exponential decay on past values.

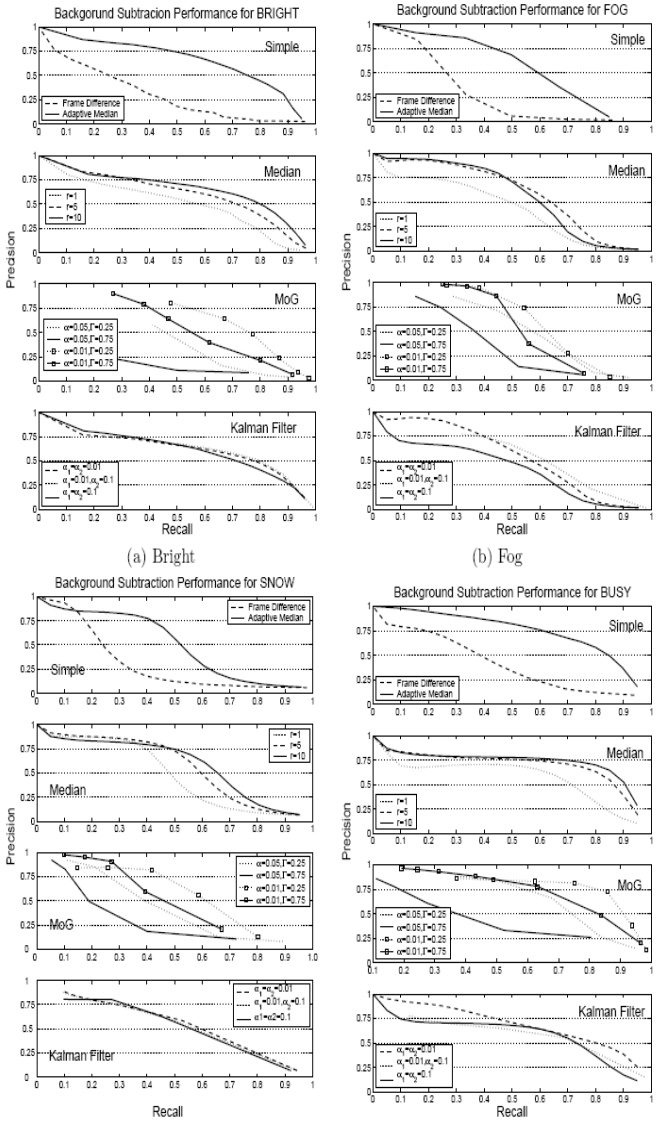


Figure 3. *Precision-recall plots for (a) Bright, (b) Fog, (c) Snow, and (d) Busy.*

## V. CONCLUSION

In this paper, we survey a number of background subtraction algorithms in the literature. We analyze them based on how they differ in preprocessing, background modeling, foreground detection, and data validation. Five specific algorithms are tested on urban traffic video sequences: frame differencing, adaptive median filtering, median filtering, mixture of Gaussians, and Kalman filtering. Mixture of Gaussians produces the best results, while adaptive median filtering offers a simple alternative with competitive performance. 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.

## ACKNOWLEDGMENT

Here, I would like to take this opportunity to express my heartfelt gratitude to the supervisor for this research project, Mrs. Abha Choubey for her patience with me and her down to earth personality which have given many pointers to guide me during my work in this report. We hope that this paper can be as informational as possible to you.

## REFERENCES

[1] Brajesh Patel, Neelam Patel *"Motion Detection based on multi frame video under surveillance systems"* Vol. 12, Mar. 2012.

[2] Cina Motamed *"Motion detection and tracking using belief indicators for an automatic visual-surveillance system"* June, 2005.

[3] David Moore *"A real world system for human motion detection and tracking"* June, 2003.

[4] G. Johansson, "Visual perception of biological motion and a model for its analysis", 1973.

[5] J. Renno, N. Lazarevic-McManus, D. Makris and G.A. Jones "Evaluating Motion Detection Algorithms: Issues and Results".

[6] Nan Lu, Jihong Wang, Q.H. Wu and Li Yang "An improved Motion Detection method for real time Surveillance" Feb, 2008.

[7] Sahu Manoj Kumar, Ms. S. Jaiswal, Patnaik Sadhana, Sharma Deepak *"Survey- Visual Analysis of Human Motion*" Apr, 2012.

[8] S.Birchfield, "Derivation of Kanade-Lucas-Tomasi tracking equation", http://robotics.stanford.edu/˜birch/ klt/derivation.ps , 1997.

[9] Shih Chia Huang "An Advanced Motion Detection Algorithm with Video Quality Analysis for Video Surveillance Systems" Vol. 21, Jan. 2011.

[10] Sumita Mishra, Prabhat Mishra, Naresh K Chaudhary, Pallavi Asthana *"A Novel comprehensive method for real time Video Motion Detection Surveillance"* Apr, 2011.

[11] Sen-Ching S. Cheung and Chandrika Kamath "Robust techniques for background subtraction in urban traffic video" Center for Applied Scientific Computing Lawrence Livermore National Laboratory.

[12] Thomas B. Moesland, Erik Granum, *"A Survey of Computer Vision- Based Human Motion Capture*", Computer Vision and Image Understanding, 81:231-268, 2001.

© 2013 IJAIR. ALL RIGHTS RESERVED