

# MOVING OBJECT DETECTION USING MATLAB

Gottipati. Srinivas Babu

NRI INSTITUTE OF TECHNOLOGY, ECE Department, Vijayawada, Andhra Pradesh, India

## ABSTRACT

Today Security is given very much importance and lot of electronic equipment is being used in security applications. Monitoring continuously the movements of persons or vehicles and reporting when predefined events take place is very common security applications. A human observation based system for implementing this has several disadvantages. In the olden days persons use to be employed for doing such observations. Decades before electronic cameras could solve the problem of man being physically present at such place. Instead man has to observe the camera's output on a TV and can detect when any expected events occur. The present day technology allows automatic detection based on predefined measures. In this foreground detection based moving object detection and vehicle tracking algorithm is implemented targeting a wide class of applications. An AVI file is read and it is decomposed into R, G and B components. Various operations are carried out and the moving objects are detected. Thresholds at various phases are decide the possibility of identifying the moving object of certain sizes. Moving objects also tracked in it. MATLAB is used for implementation of the algorithm. The algorithm is tested with input avi format video files consisting of 120 frames. Various applications of these algorithms are studied and implemented. Main focus is given on (unmanned aerial vehicle) UAV based and automatic traffic estimation applications.

**Keywords:** Security, Foreground, avi file, UAV.

## 1. INTRODUCTION

Human quest for an automatic detection system of everyday occurrence lead to the necessity of inventing an intelligent surveillance system which will make lives easier as well as enable us to compete with tomorrows technology and on the other hand it pushes us to analyze the challenge of the automated video surveillance scenarios harder in view of the advanced artificial intelligence. Now a days, it is seen that surveillance cameras are already prevalent in commercial establishments, with camera output being recorded to tapes that are either rewritten periodically or stored in video archives. To extract the maximum benefit from this recorded digital data, detect any moving object from the scene is needed without engaging any human eye to monitor things all the time. Real-time segmentation of moving regions in image sequences is a fundamental step

in many vision systems. A typical method is background subtraction. Image background and foreground are needed to be separated, processed and analyzed. The data found from it is then used further to detect motion. In this work robust routine for accurately detecting moving objects have been developed and analyzed. The traditional real time problems are taken under consideration including shadow while detecting motion.

The method chosen to obtain the goal, the problems faced during the implementation and the primary idea of the solution is discussed, along with the proposed algorithm with its describe implementation of algorithm, simulation results and conclusions and the future work.

### 1.1 Motion detection

Motion detection in consequent images is nothing but the detection of the moving object in the scene. In video surveillance, motion detection refers to the capability of the surveillance system to detect motion and capture the events. Motion detection is usually a software-based monitoring algorithm which will signal the surveillance camera to begin capturing the event when it detects motions. This is also called activity detection. An advanced motion detection surveillance system can analyze the type of motion to see if it warrants an alarm. In this, a camera fixed to its base has been placed and is set as an observer at the outdoor for surveillance. Any small movement with a level of tolerance it picks is detected as motion. Aside from the intrinsic usefulness of being able to segment video streams into moving and background components, detecting moving blobs provides a focus of attention for recognition, classification, and activity analysis, making these later processes more efficient since only "moving" pixels need be considered. There are three conventional approaches to moving object detection temporal differencing, background subtraction and optical flow. Temporal differencing is very adaptive to dynamic environments, but generally does a poor job of extracting all relevant feature pixels. Background subtraction provides the most complete feature data, but is extremely sensitive to dynamic scene changes due to lighting and extraneous events. Optical flow can be used to detect independently moving objects in the presence of camera motion; however, most optical flow computation methods are computationally complex, and cannot be applied to full-frame video streams in real-time without specialized hardware.

## 1.2 Motion in real time environment: Problems

Video motion detection is fundamental in many autonomous video surveillance strategies. However, in outdoor scenes where inconsistent lighting and unimportant, but distracting, background movement is present, it is a challenging problem. In real time environment where scene is not under control situation is much worse and noisy. Light may change anytime which cause system output less meaningful to deal with. Recent research has produced several background modeling techniques, based on image differencing, that exhibit real-time performance and high accuracy for certain classes of scene. Where the weather introduces unpredictable variations in both lighting and background movement.

## 2. Proposed Algorithm

The proposed algorithm can process the on-line and off-line video as shown in Figure 2.1.

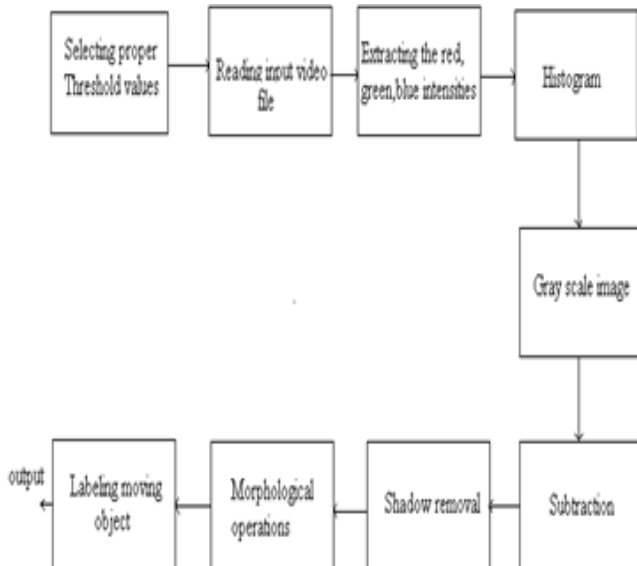


Figure 2.1. Block diagram of implemented algorithm

After selecting the optimum threshold values, input avi format video file is read and extracting the red, blue, green intensities from each and every frame of the input video, performing the histogram for background detection. Then frames are converted to grayscale images, now subtracting the background from the sequential frames for foreground detection. After detection of moving objects, shadow removing process has done for proper calculation of area of the moving object. Then morphological operations are applied and moving objects are shown with a rectangular box in the output.

## 2.1 Threshold Values

Proper threshold values have to be chosen for background, standard deviation and area of the moving objects. The statistical parameter standard deviation is used in the processing of removing the shadow of the moving object. In this algorithm threshold value of background chosen as 250 pixels, standard deviation is 0.25 and area of the moving object is 8 pixels. 8\*8 pixel is taken as one block in this algorithm.

## 2.2 Input Video

The input video format is avi. avi stands for audio video interleave. An AVI file actually stores audio and video data under the RIFF (Resource Interchange File Format) container format. In AVI files, audio data and video data are stored next to each other to allow synchronous audio-with-video playback. Audio data is usually stored in AVI files in uncompressed PCM (Pulse-Code Modulation) format with various parameters. Video Data is usually stored in AVI files in compressed format with various codecs and parameters. The aviread, aviinfo functions are mentioned to read the input video avi format. This Algorithm is tested with input video file having 120 frames.

## 2.3 Extraction

After reading the input video file, extracted the red, green and blue intensities separately to find out the histogram easily. `Image(:, :, 1)`, `image(:, :, 2)` and `image(:, :, 3)` functions are used to read the red, blue and green intensities of input video frames.

## 2.4 Histogram

An image histogram is a graphical representation of the number of pixels in an image as a function of their intensity. Histograms are made up of bins, each bin representing a certain intensity value range. The histogram is computed by examining all pixels in the image and assigning each to a bin depending on the pixel intensity. The final value of a bin is the number of pixels assigned to it. The number of bins in which the whole intensity range is divided is usually in the order of the square root of the number of pixels. Image histograms are an important tool for inspecting images. They allow you to spot background and gray value range at a glance. Histogram is used to extract the background. The histogram of a digital image with  $L$  total possible intensity levels in the range  $[0, G]$  is defined as the discrete function

$$h(r_k) = n_k$$

Where  $r_k$  is the  $k^{\text{th}}$  intensity level in the interval  $[0, G]$  and  $n_k$  is the number of pixels in the image whose intensity level is  $r_k$ . The value of  $G$  is 255 for images of class unit8, 65535 for images of class unit16, and 1.0 for images of class double. Indices in MATLAB cannot be start with 0, hence  $r_1$  corresponds to intensity level 0,  $r_2$

corresponds to intensity level 1, and so on, with  $r_L$  corresponding to level G.  $G=L-1$  for images of classes. Often, it is useful to work with normalized histograms, obtained simply by dividing all elements of  $h(r_k)$  by the total number of pixels in the image, which is denoted by  $n$

$$P(r_k) = \frac{h(r_k)}{n} \\ = \frac{n_k}{n}$$

For  $k=1, 2, \dots, L$ ,  $P(r_k)$  is an estimate of the probability of occurrence of intensity level  $r_k$ . Histogram, imhist, mat2gray functions are used in this part.

## 2.5 Grayscale image

Grayscale images are images without color, or achromatic images. The levels of a gray scale range from 0 (black) to 1 (white). After calculating the histogram, images are converted into gray scale images to reduce the complexity while applying the morphological operations.

## 2.6 Subtraction

This proposed algorithm dynamically extracts the background from incoming all video frames, it is subtracted from every subsequent frame and compared with the background threshold. If it is greater than the background threshold, it is assumed as foreground otherwise it is background. The Background is updated in each and every frame.

## 2.7 Shadow removal

Performing the operation using a function on each frame by  $8 \times 8$  block wise and result is compared with the variance threshold. If the result is less than the variance threshold, it is assumed as shadow and it takes logic 0 otherwise it takes logic 1.

## 2.8 Morphological operations

Morphology is a broad set of image processing operations that process images based on shapes. Morphological operations apply a structuring element to an input image, creating an output image of the same size. The most basic morphological operations are dilation and erosion. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, you can construct a morphological operation that is sensitive to specific shapes in the input image.

### 2.8.1 Dilation

Dilation is an operation that “grows” or “thickens” objects in a binary image. The specific manner and extent

of this thickening is controlled by a shape referred to as a structuring element. In other words, the dilation operation usually uses a structuring element for probing and expanding the shapes contained in the input image. Dilation is commutative; that is  $A+B = B+A$ . It is a convention in image processing to let the first operand of  $A+B$  be the image and the second operand be the structuring element, which usually is much smaller than the image. For example a simple binary image A containing one rectangular object, the example uses a  $3 \times 3$  square structuring element object.

```
[0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0]
```

After performing the dilation operation, binary image A is given by

```
[0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0]
```

### 2.8.2 Erosion

Erosion “shrinks” or “thins” objects in a binary image. As in dilation, the manner and extent of shrinking is controlled by a structuring element. Erosion operation is quite opposite to the dilation operation. After performing the erosion operation on binary image A, using  $3 \times 3$  square matrix as structuring element is given by

```
[0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 0 0 0 0 1 0 0
0 0 1 0 0 0 0 1 0 0
0 0 1 0 0 0 0 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0]
```

In practical image-processing applications, dilation and erosion are used most often in various combinations. An image will undergo a series of dilations and/or erosions using the same, or sometimes different, structuring elements. `imdilate` and `imerode` matlab functions are used in this part.

### 2.9 Labeling the moving object

After performing the morphological operations, the area of the moving object is calculated and labeling the moving objects with red color rectangle in the output.

## 3 SIMULATION RESULTS

The proposed algorithm is tested with input avi video file consisting of

```
file_id.NumFrames      = 120
file_id.Framespersecond = 8
file_id.width          = 160
file_id.Height         = 120
```

The proposed algorithm extracted the background from all 120 frames. The background is shown in figure 3.1



Figure 3.1. Reconstructed background

In figure 3.2, shows the moving vehicles on the road.



Figure 3.2. Moving object

To distinct between background and moving objects (foreground here), histogram operation is performed and background frame is subtracted from foreground, result frame is shown in figure 3.3



Figure 3.3. Detected moving object with shadow

This algorithm removes the shadow of the moving object in order to calculate the area of the object effectively. During this process, after comparing with standard deviation threshold, result frame is shown in figure 3.4. If the result is less than standard deviation threshold, it is shadow.

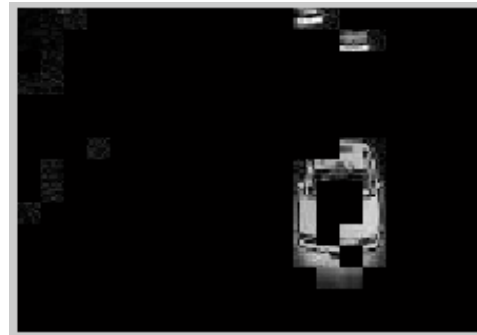


Figure 3.4 shadow removed object after comparing with standard deviation

Then compared with background threshold, if result is greater than the background threshold it is foreground otherwise it is foreground. The result frame is shown in figure 3.5.

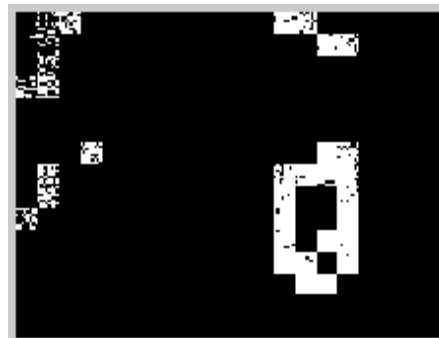


Figure 3.5 after comparing with background threshold

Then morphological operations i.e. dilation and erodes are performed and moving object is labeling with red color rectangular box and counting is incremented.

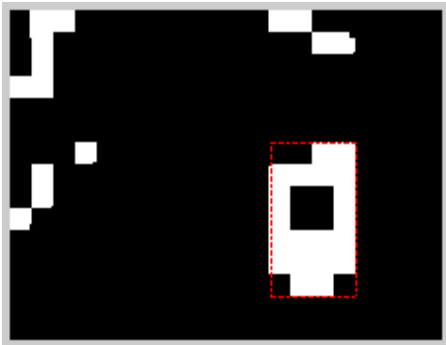


Figure 3.6 Labeling with moving object after morphological processing

The same process which is explained above for one moving object can be applied for remaining objects also and result frames are shown below.

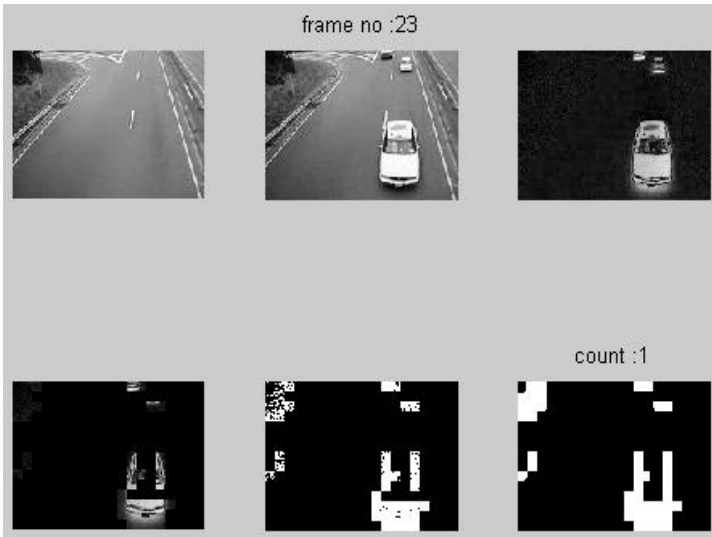


Figure 3.7 with counting 1

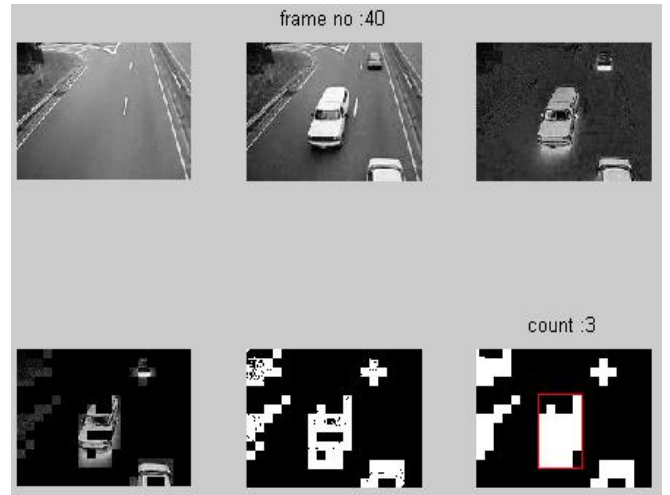


Figure 3.9 with counting 3

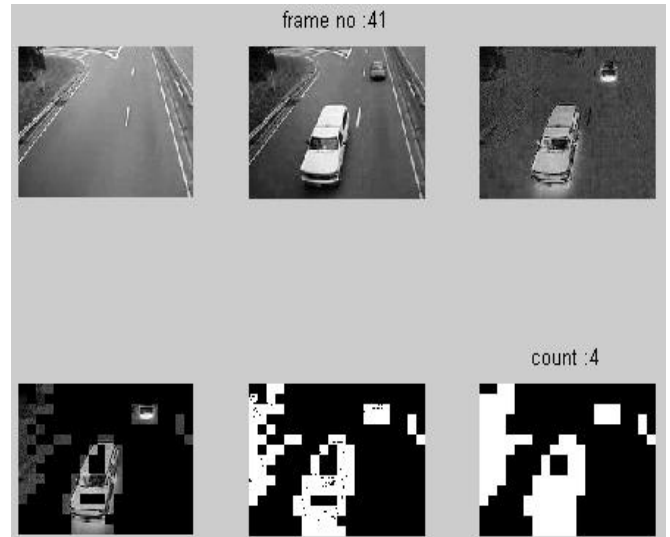


Figure 3.10 with counting 4

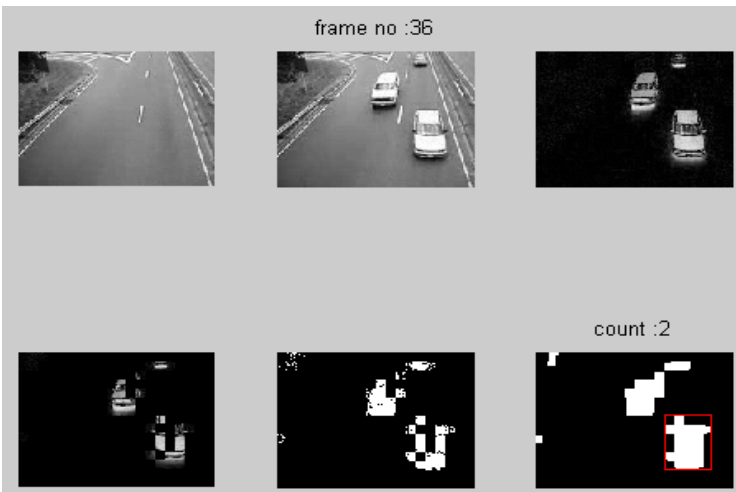


Figure 3.8 with counting 2

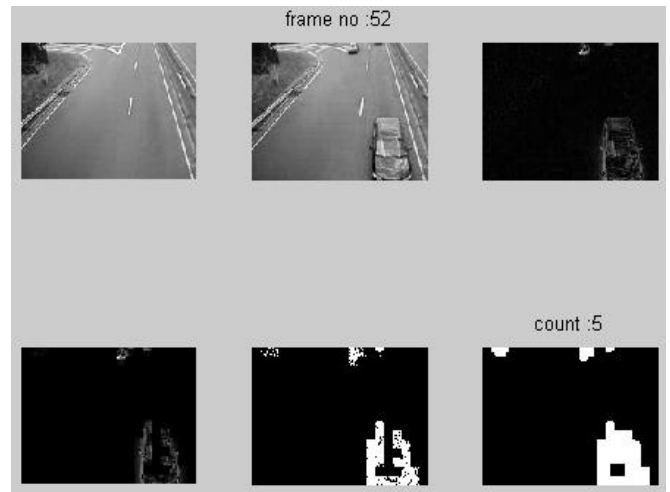


Figure 3.11 with counting 5



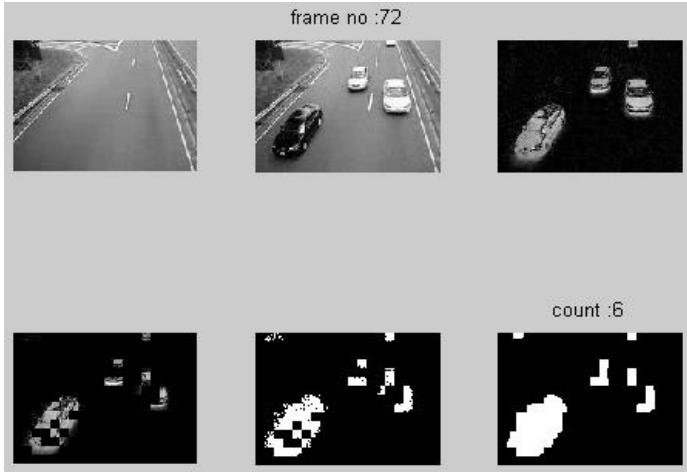


Figure 3.12 with counting 6

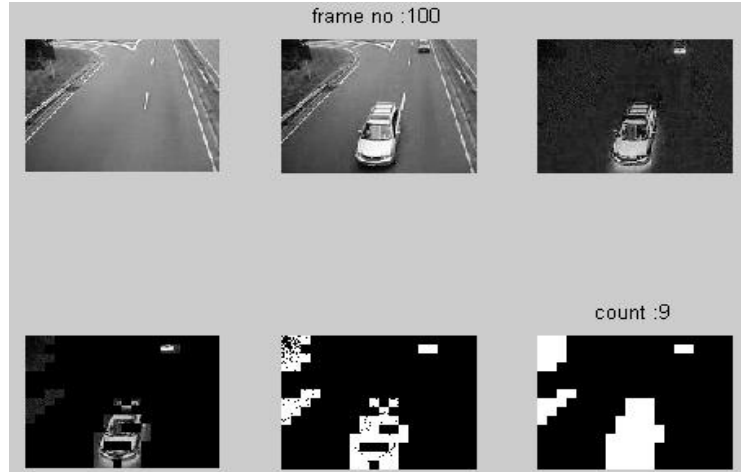


Figure 3.15 with counting 9

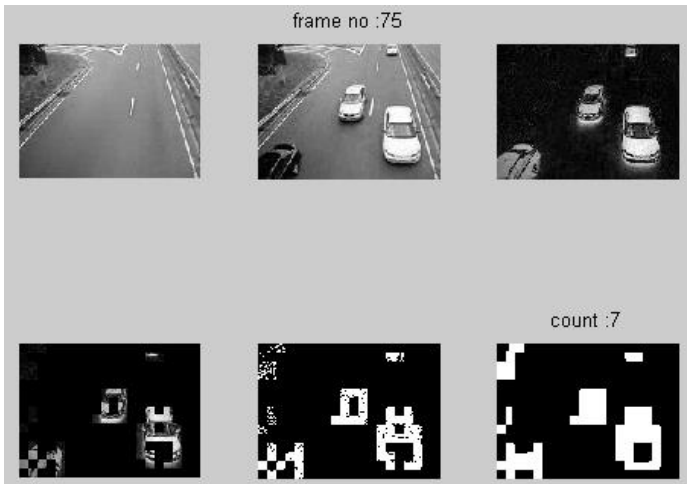


Figure 3.13 with counting 7

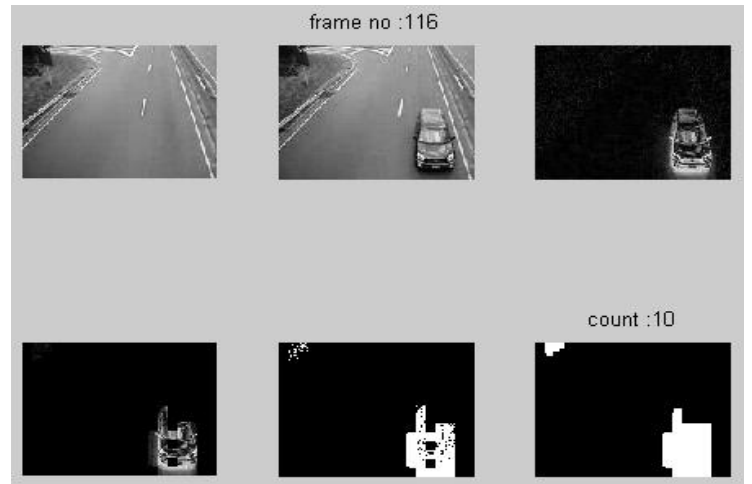


Figure 3.16 with counting 10

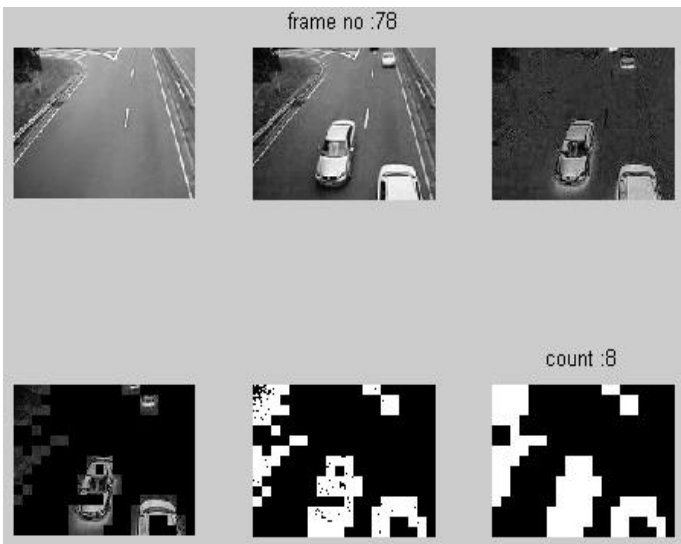


Figure 3.14 with counting 8

#### 4 CONCLUSIONS AND FUTURE WORK

The proposed algorithm extracted the background from the all frames of video and detected the foreground effectively. This algorithm also dynamically updating the background frame by frame .This algorithm also identify the shadow of the moving object and is removed to calculate the area of the object accurately. This algorithm also identifies even small object by adjusting the threshold values. Even the smallest, slowest, fastest, of a moving region is detected accurately by selecting the proper threshold value of the objects. Finally this algorithm works for On-line (Real time) and Off-line (Quasi real time) video processing and its computational complexity is low.

Future work will be directed towards achieving the following issues:

- Object classifications
- Better understanding of human motion not only vehicle, including segmentation and tracking of

- articulated body parts.
- Improved data logging and retrieval mechanisms to support 24/7 system operations
- Better camera control to enable smooth object tracking at high zoom, incase, video is vibrating Video stabilization algorithm is required.
- Acquisition and selection of “best views” with the eventual goal of recognizing individuals in the scene.

### REFERENCES

1. Ahmed Elgammal, David Harwood, Larry Davis, Non-parametric Model for Background Subtraction, Computer Vision Laboratory University of Maryland, College Park, MD 20742, USA.
2. Alan J. Lipton Hironobu Fujiyoshi Raju S. Patil. Moving Target Classification and Tracking from Real-time Video The Robotics Institute Carnegie Mellon University 5000 Forbes Ave Pittsburgh, PA, 5213.
3. C.R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, “Pfinder: Real-Time Tracking of the Human Body,” IEEE Trans. Pattern Analysis and Machine Intelligence.
4. Du Yuren Zhou Aijun Yuan Feng “Moving object detection for video monitoring system”,the Eighth International Conference on Electronic Measurement and Instruments, ICEMI'2007.
5. E. Grimson, C. Stauffer, R. Romano, and L. Lee, “Using Adaptive Tracking to Classify and Monitoring Activities in a Site, Proc. Computer Vision and Pattern Recognition Conf., pp. 22-29, 1998.
6. I. Haritaoglu, D. Harwood, and L. Davis, W4: Who, When, Where, What: A Real Time System for Detecting and Tracking People, Proc. Third Face and Gesture Recognition Conf., pp. 222-227, 1998.
7. J. Bergen, P. Anandan, K. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In Proceedings of the European Conference on Computer Vision, 1992.
8. Kedar.A.Patwardhan, Guillerimo Saprio “Robust Foreground Detection in Video using pixel layers”,IEEE Trans. Pattern Analysis and Machine Intelligence, Vol 30,no.4, April 2008.
9. Robert T. Collins, Alan J. Lipton, Takeo Kanade, Hironobu Fujiyoshi, David Duggins, Yanghai Tsin, David Tolliver, Nobuyoshi Enomoto, Osamu Hase gawa, Peter Burtl and Lambert Wixson, A System for Video Surveillance and Monitoring, The Robotics Institute, Carnegie Mellon University, Pittsburgh PA 1 The Sarnoff Corporation, Princeton, NJ.
10. R.C. Jain and H.H. Nagel, “On the Analysis of Accumulative Difference Pictures from Image Sequences of Real World Scenes,” IEEE Trans Analysis and Machine Intelligence.
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 7000 East Avenue, Livermore, CA 94550.