ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/274270897

High Density Traffic Management using Image background subtraction Algorithm

Article *in* International Journal of Computer Applications · February 2014

CITATION

1

READS

217

3 authors, including:

Ravindra S Hegadi

Solapur University

126 PUBLICATIONS **500** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Medical image analysis View project

Project International Conference on Recent Trends in Image Processing & Pattern Recognition View project

All content following this page was uploaded by Ravindra S Hegadi on 31 March 2015.

The user has requested enhancement of the downloaded file.

High Density Traffic Management using Image background subtraction Algorithm

Somashekhar. G.C
Department of ECE
Rajiv Gandhi institute of Tech,
Bangalore-560032, India

Sarala Shirabadagi
Department of ECE
Sahyadri Institute of Tech,
Mangalore-575007, India

Ravindra S. Hegadi
Department of Computer Science
Solapur University
Solapur-413255, India

ABSTRACT
Traffic congestion has been increasing because of increased population growth mainly in major cities due to urbanization. Traffic congestion causes increased air pollution, travel time and mostly traffic accidents; therefore we need an efficient traffic management system. Today most of the cities of the world have intelligent transport system which is equipped with electronics devices to communicate about the traffic condition with the moving vehicle and also monitor the traffic rules and regulation. Current traffic control techniques involving magnetic loop detectors buried in the road, infra-red and radar sensors on the side provide limited traffic information and require separate systems for traffic counting and for traffic surveillance. The disadvantages of the existing system are that it requires traffic personnel to monitor the traffic and magnetic loop detectors cause's high failure rate when installed on the road surfaces.

In contrast, video-based systems offer many advantages compared to traditional techniques. They provide more traffic information, combine both surveillance and traffic control technologies, are easily installed, and are scalable with progress in image processing techniques. Implementation of the project will eliminate the need of traffic personnel at various junctions for regulating traffic. Thus the use of this technology is valuable for the analysis and performance improvement of road traffic.

Traffic monitoring based on density of vehicles improve the traffic control system by calculating the density of vehicles on the road. This Project proposes to control the traffic using image processing algorithms and embedded systems to control the traffic signals. The videos taken by a camera is analyzed using Background Subtraction method to detect, track and count the number of vehicles moving in each lane to obtain the most efficient traffic management. The vehicle classification and speed detection is also done, such that vehicles are classified into heavy vehicle (trucks, bus) and low vehicle (bikes, cars) and the over speed vehicles are detected and indicated by red boundary.

Keywords: Back ground subtraction, Loop detectors and feature extraction.

1. INTRODUCTION
Current traffic control techniques involving magnetic loop detectors buried in the road as in Figure1.1, infra-red and radar sensors as in Figure 1.2 provide limited traffic information and require separate systems for traffic counting and for traffic surveillance. Inductive loop detectors do provide a cost-effective solution, however they are subject to a high failure rate when installed in poor road surfaces, decrease pavement life and obstruct traffic during maintenance and repair. Infrared sensors are affected to a greater degree by fog than video cameras and cannot be used for effective surveillance. In manual traffic control the individual traffic personnel is made to stand at each individual junction. Even though this concept being most reliable as the traffic flow is controlled based on priority but it created health concern. In the present traffic control technique which is timer based automatic timer system the traffic flow is analyzed for a

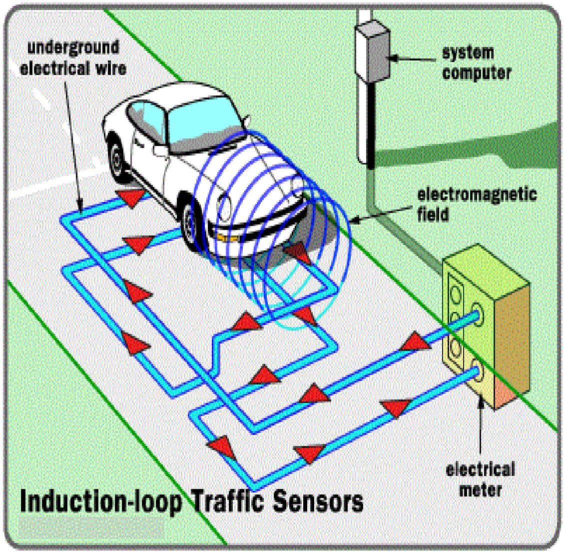

Figure 1.1: Inductive loop detectors based traffic management

certain period of time, the traffic lights are embedded, it is not reliable as the traffic flow can never be interpreted and it's proved to be a non reliable technology. It is based on 1 minute time interval, the vehicles move at a normal speed when the light is green for a certain interval of time(e.g. 1minute), but come to a stop when it turns red after the interval. Sometimes when there are no vehicles in lane but still it's given green because traffic signals change depending upon time interval.

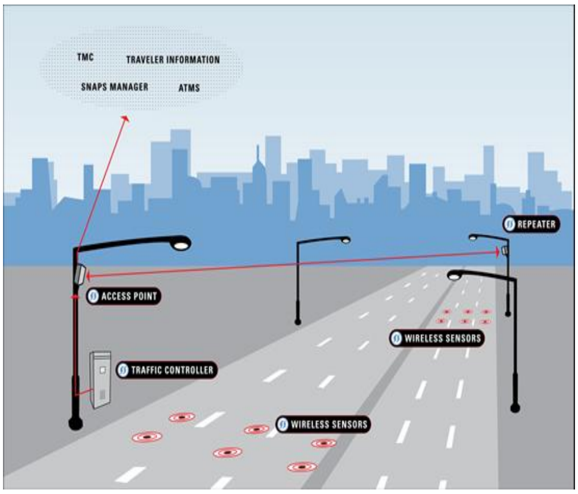

Figure 1.2: Sensor Based Traffic Control

Figure 1.3 shows the scenario of present traffic control technique which is timer based automatic timer system. Here the number of vehicles in lane1 is less i.e. 2 vehicles and in lane2 it's 3 vehicles and in lane3 it's 11 vehicles. Even when the number of vehicles in lane1 is less and in the lane3 is more but still lane1 will be given green signal first (as shown in Figure 1.3 a) until the time interval (e.g. 1 minute) and then green signal is given to lane2 as shown in Figure 1.3(b), but till then the traffic in lane3 will be growing which creates heavy traffic.

In contrast, video-based systems offer many advantages compared to traditional techniques. They provide more traffic information, combine both surveillance and traffic control technologies, are easily installed, and are scalable with progress in image processing techniques. Implementation of the project will eliminate the need of traffic personnel at various junctions for regulating traffic. Thus the use of this technology is valuable for the analysis and performance improvement of road traffic.

The main objective is to develop a methodology for automatic vehicle detection and its counting on lanes. Depending upon the count of vehicles the traffic signals are changed, where the traffic signals are controlled by PIC microcontroller. A system has been developed to detect and count dynamic objects efficiently. Intelligent visual surveillance for road vehicles is a key component for developing autonomous intelligent transportation systems. The algorithm does not require any prior knowledge of road feature extraction on static images; it includes detecting and tracking vehicles in surveillance video which uses segmentation with initial background subtraction and then using morphological operator to determine salient regions in a sequence of video frames.

1.1: Overview

Identifying moving objects from a video sequence is a fundamental and critical task in video surveillance, traffic monitoring and analysis. A common approach in identifying the moving objects is background subtraction, where each video frame is compared against a reference or background model. Pixels in the current frame that deviate significantly from the background are considered to be moving objects. These "foreground" pixels are further processed for object localization and tracking. 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. Moving object detection is the basic step for further analysis of video. It handles segmentation of moving objects from stationary background objects. This not only creates a focus of attention for higher level processing but also decreases computation time considerably.

The various steps of our proposed system are a camera (webcam) is fixed on polls or other tall structures to overlook the traffic scene [1].Each frame extracted from the video are then analyzed to track, detect and count vehicles. Then depending on the signal cycle (1 minute), time is allotted to each lane. For example, if the number of vehicles in a four-lane intersection is found to be 10, 30, 20 and 5, then signal are changed depending upon the count of vehicles. The lane which contains 30 vehicles is given green signal first and then the lane with 20 and then lane with 10 and lane with 5 vehicles.

The following steps are performed for the above mentioned objective:

Step1: Webcam initialization is done by properly selecting adaptor name, device name and format.
Step2: Delete the previous video input object and all the video source objects associated with the previous video input object.
Step3: Capture sequences of video frames by specifying the number of frames required.
Step4: Perform the background modeling in the video.
Step5: Perform the background subtraction.
Step6: Detect the moving vehicle.
Step7: Perform the tracking of moving object.
Step8: Perform the feature matching.
Step9: Perform the noise removal by properly selecting the filters.
Step10: Perform the counting of moving vehicle.
Step11: Perform classification of vehicle.
Step12: Detect the speed of the vehicle.
Step13: Highlight the over speed vehicle.
Step14: Perform the traffic signal control depending on count of vehicles using PIC microcontroller.
Step15: Perform the background updating.

1.2: Vehicle Tracking, detection, counting and classification

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 project, 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.

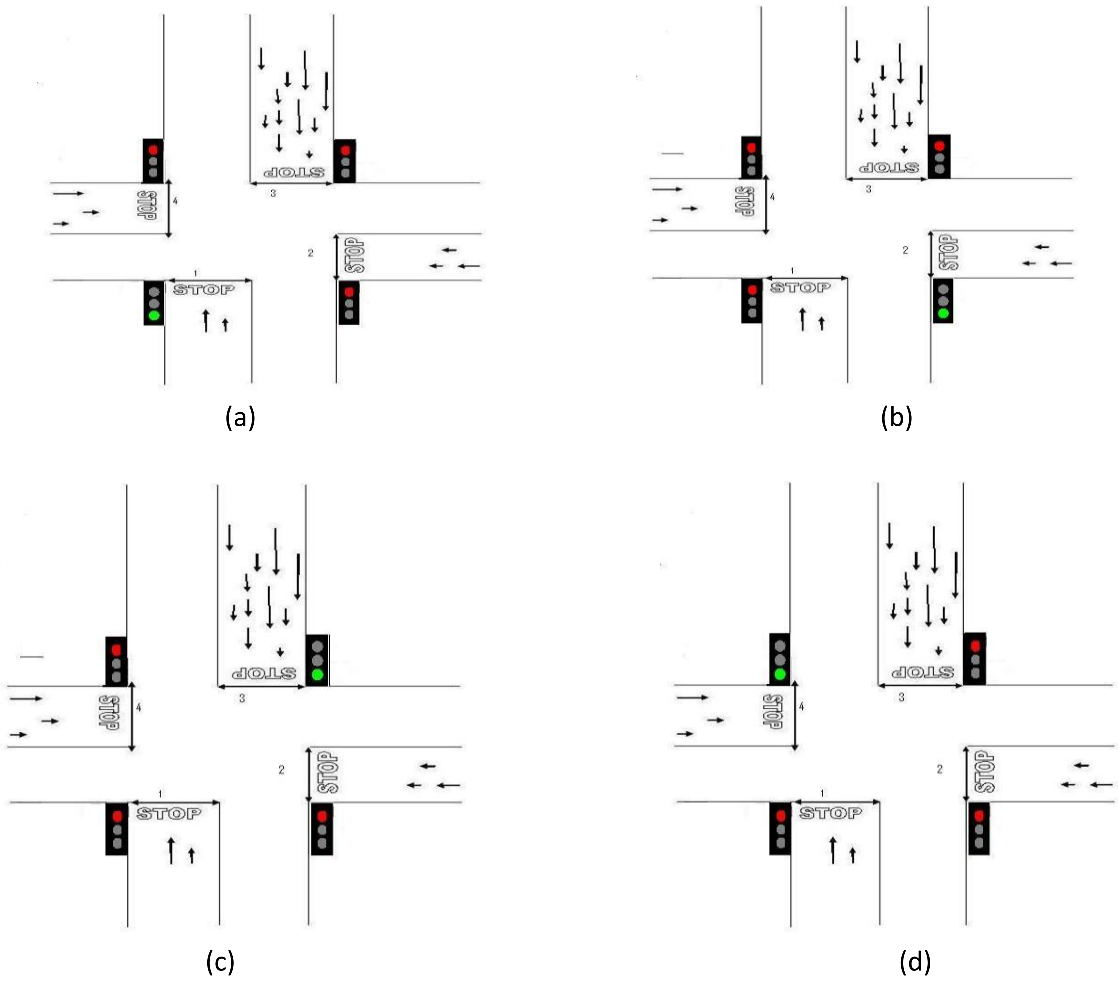

Figure 1.3: Current Traffic Control Based on Timer and Circularly Changing Signal

1.3: Problems in real time environment

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.

The video in avi format is processed by selecting RGB model smoothened by 5×5 averaging masks. After background modeling by median filtering, the vehicles are detected, tracked and classified. Speed of the vehicle is also estimated

2. PROPOSED ALGORITHM

In the proposed architecture flow diagram as shown in Figure 4.1, the video is acquired through a stationary camera and then the background modeling is done by median filtering The processing includes: 1) automatically determining the initial positions of moving objects, 2) extracting feature information from all moving objects within view, 3) tracking detected objects by feature and 4) classifying the moving objects into two categories: heavy vehicle and low vehicle. Low vehicle are cars and bikes and heavy vehicle are trucks and bus. Our tracking system integrates spatial position, motion, shape and color.

This integration makes the tracker insensitive to changes in background, interruption of motion and orientation of objects. We detect the motion and background variation to segment the moving object blobs, and then compare the similarity of the object blobs with different templates, thereby tracking the objects. After obtaining an accurate description of the observed object, we classify the objects and update the templates, taking into account any occlusion.

4. MOVING OBJECT DETECTION

The first step in tracking objects is separating the objects from the background. Two common methods are frame differencing and background subtraction. Frame differencing is basically a threshold of the difference between the current image and sequence images by assuming that the background does not change over successive frames. This method is easy and fast in many applications, but some problems appear when tracking multiple objects or when an object stops in which the moving object is not accurately detected. Hence, we turn to the other method, background subtraction, at the expense of updating the background. A pixel-wise median filter with *L* frame length is employed to build the background under the assumption that a moving object would not stay at the same position for more than half of L where L is previous frames stored in buffer. Otherwise, the object would be incorporated into the background [3]. The background model for pixel xt (m, n) at frame *t* using a length *L* median filter is given by equation 4.1.

$$\mathbf{xt}\,(\mathbf{m},\mathbf{n}) = \mathbf{medianL}\big(\,\mathbf{xt} - \mathbf{0.5L}\,(\mathbf{m},\mathbf{n}),\ldots,\mathbf{xt} + \mathbf{0.5L}\,(\mathbf{m},\mathbf{n})\big) \quad \textbf{..........................(4.1)}$$

This retains the stationary pixels in the background. A median filter can build the background even when moving objects exist in the scene, but usually requires a large amount of memory to save L frames at a time. This is only applied when we detect a large new blob, and this blob lasts for several frames. Background subtraction is performed in color and in texture. Background subtraction is performed on RGB color model.

4.1.1 The RGB Color Model

The RGB (*Red, Green, Blue*) color model uses a Cartesian coordinate system and forms a unit cube shown in Figure 1.

The Figure 4. shows the RGB model in the Cartesian coordinate system. The dotted main diagonal of the cube, with equal amounts of *Red*, *Green* and *Blue*, represents the gray levels. This diagonal is further referred to as the *gray diagonal*. The RGB color model is hardware oriented and is used in many image capturing, processing and rendering devices.

Subtract the foreground from the background in each RGB color channel and then take the maximum absolute values of the three differences as the difference value Diff_c in color space.

$$\mathbf{Diff\ c} = \mathbf{max}\{\,|\mathbf{R\,f} - \mathbf{Rb}\,|\,,|\mathbf{G\,f} - \mathbf{Gb}\,|\,,|\mathbf{B\,f} - \mathbf{Bb}\,|\,\} \quad \textbf{...............(4.2)}$$

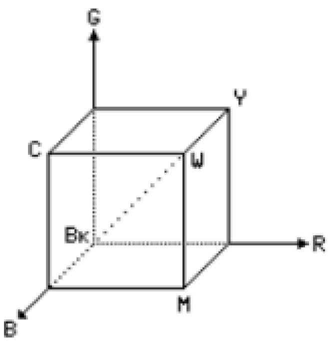

Figure 4.2: RGB Model in Cartesian

The equation 4.2 indicate the subtraction of foreground from background in RGB color channel and then taking the maximum absolute values of the three differences as the difference value Diff_c in color space. Where $R_{f,}$, $G_{f,}$, B_f are red green blue values in foreground frame and $R_{b,}$, $G_{b,}$, B_b are red green blue values in background frame.

The binary foreground pixels F (x, y) are produced by equation 4.3.

$$F(x,y) = \begin{cases} 1 & if\ \text{Diffc} > Th \\ 0 & otherwise \end{cases} \quad \textbf{................ (4.3)}$$

The resulting foreground contains noise due to the clutter in the background. The 'close' binary morphological operator is applied to remove noise. The foreground F is segmented into several isolated blobs by an eight connected algorithm. Assuming that each initial blob contains a moving object, which can be a person, vehicle or people groups, after finding individual blobs that correspond to objects, the bounding boxes of these regions are drawn.

4.1.2 Noise Removal

The background and the discrepancy image contains the motion region as well as large number of noise, which is due to environmental factors, illumination changes, and during transmission of video from the camera to the further processing. These noises are removed by the morphological operation.

5. FEATURE EXTRACTION

Two types of features are extracted in each moving blob. They are centroid of object and color as a feature. These features are generally used in object tracking. The centroid of a blob tells us the spatial position of the blob. It is the average of positions of all pixels in the blob. An object will not move far from its last position, therefore, the centroid presents us with a strong and useful feature for tracking object.

5.1: Object tracking

The tracking process compares the feature vector $R_{i,t}$ with all templates $T_{i,t-1}$ (i=1,2…M). If a matching is found, then the template is updated for the next matching through an adaptive filter as given in equation 4.7. Where β (∈[0.0, 1.0]) is a learning constant which specify how much information from the incoming image is put to the background.

$$\boldsymbol{Ti,t} = \boldsymbol{\beta Ti,t-1} + (\mathbf{1} - \boldsymbol{\beta})\boldsymbol{Ri,t} \quad \textbf{........(4.7)}$$

If no matching is found for several successive frames, then a new candidate template

$T_{M+1,t}$ will be created. The system remembers only the templates that are matched during the last 50 frames. A hierarchical matching process is performed in the order of centroid, shape and color.

The vehicle classification done based on shape of the object , by calculating the length and area of the object. Low vehicle such as bike and cars have less area compare to heavy vehicle such as trucks.

5.2: Vehicle Counting

The tracked binary image mask1 forms the input image for counting. This image is scanned from top to bottom for detecting the presence of an object. Two variables are maintained i.e., count that keeps track of the number of vehicles and count register counter, which contains the information of the registered object. When a new object is encountered it is first checked to see whether it is already registered in the buffer, if the object is not registered then it is assumed to be a new object and count is incremented, else it is treated as a part of an already existing object and the presence of the object is neglected.

This concept is applied for the entire image and the final count of objects is present in variable count. A fairly good accuracy of count is achieved. Sometimes due to occlusions two objects are merged together and treated as a single entity.

5.3: Steps to count vehicle

1. Traverse the mask in image to detect an object.
2. If object encountered then check for registration in counter.
3. If the object is not registered then increment count and

register the object in count labeled with the new count.
4. Repeat steps 2-4 until traversing not completed.

5.4: Speed Calculation

Now after tracking each object in video save the frame

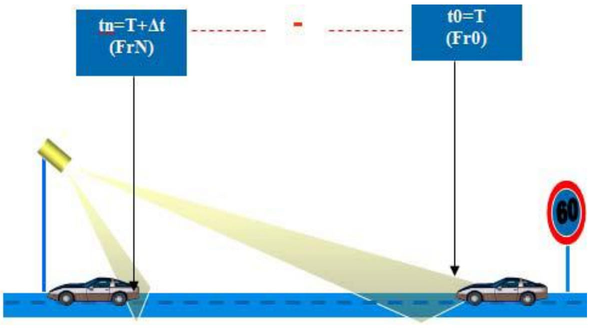

Figure 5.1 Speed detection model

number that the object entered the scene at (Fr_0), and the frame number that the object left the scene at (Fr_N), then speed calculation can be held out by calculating the number of frames consumed by the object to pass-by the scene (enter and leave it) and since the duration of each frame (extracted from the video Frame Rate) is know therefore we can calculate the total time taken by the object to pass-by the whole scene.

$$\mathbf{T = N * TF,\ N = FrN - Fr0} \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots..(\mathbf{5.1})$$

T: Total time taken by the object to pass-by the scene.
TF: duration of one frame.
N: Total number of frames.
Then by using the simple equation for speed calculation which states that:

$$\mathbf{T = \Delta t = (N * TF)}$$

$$\mathbf{Speed = d / (tn - t0) = d / \Delta t} \ . \ \ \ldots\ldots\ldots\ldots\ldots(\mathbf{5.2})$$

We can calculate the speed of the moving object easily. In equation 5.2 speed of vehicle is calculated by Distance Travelled by the vehicle/Frame Rate where d is the distance travelled by vehicle from its initial position at Fr_0 to the position after Fr_N, t0 and tn is the time between Fr_0 and Fr_N. During the tracking phase, the system has stored several information about each object in order to help the system to find the speed:

1- Label → The label that the object takes after labeling operation and correct labels operation.

2- Fr_0 → Frame 0, → Fr_N Frame N

3- Image → Captured Image to the object when it was at the center of scene.

The distance travelled by the object is determined by using the centroid. It is calculated by using the Euclidean distance formula. The variables for this are the pixel positions of the moving object at initial stage to the final stage

$$\textbf{Distance} = \sqrt{(X2 - X1)^2 + (Y2 - Y1)^2} \ldots\ldots\ldots\ldots (\mathbf{5.3})$$

In equation 5.3 X1 = previous pixel position and X2 = present pixel position in width Y1 = previous pixel position and Y2 = present pixel position in height The velocity of moving object is calculated by the distance it travelled with respect to the time. Euclidean distance formula is used to calculate the distance between the sequences of frames. By using the values of distance with respect to frame rate, the velocity of the object is defined.

$$\textbf{Velocity} = \frac{\textbf{Distance travelled}}{\textbf{Frame rate}}$$

After calculating speed, the system checks to see if it is violating the pre-defined speed limit determined by the user during the system configuration.
The Figure 5.1 shows the speed detection model Fr_0 indicates the initial position of vehicle at time t0 and Fr_N indicates the position of vehicle after moving N frames at time tn. Hence by knowing the amount of pixels moved from initial position to the position of vehicle after N position and by knowing the time between t0 and tn the speed of the vehicle is calculated by equation 5.2.

6. RESULT

Background modeling by median filtering is initially done for standard traffic videos in which the vehicle tracking, vehicle detection, speed calculation, vehicle classification, and vehicle counting is done. Video1 of 530 frames, width 640and height 360.

Input Video	Format	Frames	Actual No of Vehicles	Detected No of Vehicles	Accuracy
Video 1	RGB	530	4	4	100%
Video 2	RGB	100	4	4	100%
Video 3	RGB	120	6	6	100%
Video 4	RGB	600	11	10	91%

Table 1. Accuracy of counting in Videos

7. CONCLUSION AND FUTURE WORK

The background subtraction technique using a median filtering and frame differencing is studied. The problem of selecting threshold for frame differencing is seen hence median filtering is chosen. Noise removal using morphological operator have been studied. The project worked is considered in ideal conditions.

A system has been developed to detect, count and classify vehicles on a road efficiently. The system effectively combines simple domain knowledge about object classes with time domain statistical measures to identify target objects in the presence of partial occlusions and ambiguous poses, and the background clutter is effectively rejected. The experimental results show that the accuracy of counting vehicles was 91%, although the vehicle detection was 100% which is attributed towards partial occlusions. The system has also been implemented in real time which gives an accuracy of 91.4%, although there was a 100% vehicle detection due to occlusion of two objects which are treated as a single entity and less frame rate of webcam which cannot count very fast moving vehicle.
The computational complexity of our algorithm is linear in the size of a video frame, frame rate and the number of vehicles detected.

7.1. Future work:

- Better understanding of human motion not only vehicle.

- Improved data lagging and retrieval mechanism to support 24/7 system operations
- Better camera to be used to track smooth objects tracking at high zoom, incase video is vibrating video stabilization algorithm is required.
- The background subtraction algorithm has to be improved to work in different light conditions, snow, and rain.
- The focus shall be to implement the controller using DSP as it can avoid heavy investment in industrial control computer while obtaining improved computational power and optimized system structure.

8. REFERENCES

[1] J. Heikkila and O. Silven "A real-time system for monitoring of cyclists and pedestrians". In Proc of Second IEEE Workshop on Visual Surveillance, pages 74–81, Fort Collins, Colorado, June 1999.

[2] A. J. Lipton, H. Fujiyoshi, and R.S. Patil "Moving target classification and tracking from real-time video". In Proc. of Workshop Applications of Computer Vision, pages 129– 136, 1998.

[3] H. Wang and S.F. Chang "Automatic face region detection in mpeg video sequences". In Electronic Imaging and Multimedia Systems, pages 160–168, SPIE Photonics China, November 1996.

[4] R. T. Collins et al "A system for video surveillance and monitoring" VSAM final report, Technical report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, May 2000.

[5] M. Saptharishi, J.B. Hampshire II, and P. Khosla "Agent-based moving object correspondence using differential discriminative diagnosis". In Proc. Of Computer Vision and Pattern Recognition, pages 652–658, 2000.

[6] T. Brodsky et al, "Visual Surveillance in Retail Stores and in the Home", chapter 4, pages 51–61 of Video-Based Surveillance Systems, Kluwer Academic Publishers, Boston, 2002.

[7] R. Cutler and L.S. Davis, " Robust real-time periodic motion detection, analysis and applications". In IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 8, pages 781–796, 2000.

[8] A. Amer, "Voting-based simultaneous tracking of multiple video objects". In Proc. SPIE Int. Symposium on Electronic Imaging, pages 500–511, Santa Clara, USA, January 2003.

[9] Z. Jinglei, L. Zhengguang, and T. Univ, "A vision-based road surveillance system using improved background subtraction and region growing approach," Eighth ACIS Int. Conf. on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Qingdao, August 2007, pp. 819-822.

[10] S.Gupte, O. Masoud, et al, "Vision-based vehicle classification," in Proc. IEEE *Conference* on Intelligent Transportation Systems, pp.46-51, 2000, Dearborn, USA.

View publication stats