

APPLYING EDGE DENSITY BASED REGION GROWING WITH FRAME DIFFERENCE FOR DETECTING MOVING OBJECTS IN VIDEO SURVEILLANCE SYSTEMS

Abhilash K. Sonara¹, Pinky J. Brahmhatt², Manoj D. Chaudhary³

¹M.E Student, ECE Department, L.D. College of Engineering, Gujarat, India

²Associate Professor, ECE Department, L.D. College of Engineering, Gujarat, India

³M.E Student, ECE Department, L.D. College of Engineering, Gujarat, India

Abstract

Detecting moving object is the key step in Video based Surveillance Systems. This paper presents a unique approach for detecting moving object from a video sequence based on edge detection and Edge Density Based Region Growing. To start with the video sequence is sub-divided into its constituent frames. The method proceeds by considering two consecutive frames at time intervals (t-1) and t. Then after we use Sobel's operator to detect the edges present in respective frames. Detecting edges helps in estimating the object boundaries. After this we perform frame difference which eliminates most of the edges belonging to background. To highlight the regions containing the moving object we apply Edge Density Based Region Growing algorithm on the edge difference image. At last using Connected Component Analysis and Morphological Opening we extract the exact moving object. Experimental results show that the method is robust enough to detect the moving objects with complex backgrounds.

Keywords: Contrast Stretching, Edge Density Based Region Growing, Edge Detection, Frame Difference, Morphological Opening, Moving Object Detection, Video Surveillance.

-----***-----

1. INTRODUCTION

Moving object detection is the basic step for further processing and analysis of video captured using surveillance cameras. The advancement in computing power, availability of large-capacity storage devices and high speed network infrastructure has paved the way for cheaper, multi sensor video Surveillance Systems. Video surveillance demands design and implementation of intelligent systems that can automatically detect, track and recognize objects in the captured video sequence. Moving object detection has a wide number of applications including Visual surveillance, Event detection; Activity based Human recognition, Path detection, Object tracking etc [1]-[3]. Traditionally, the video outputs are processed online by human operators and are usually saved to tapes for later use only after a forensic event. After object detection comes object classification step which is used to categorize detected objects into predefined classes such as human, vehicle, animal, clutter, etc. It is necessary to distinguish objects from each other in order to track and analyze their actions reliably. Based on the movements of objects and background, video sequences can be categorized into two types: moving objects with moving background and moving objects with fixed background. Conventional approaches to moving object detection include Frame Differencing; Background Subtraction, and Optical Flow.

Background subtraction attempts to detect moving object by subtracting the current image pixel-by-pixel from a reference image (background image) that is created by averaging images over time in an initialization period. The pixels where the difference is above a threshold are classified as foreground and the remaining serve as those belonging to background. Though background subtraction techniques perform well at extracting most of the relevant pixels belonging to moving regions, they are usually sensitive to sudden dynamic changes [4]. Frame differencing attempts to detect moving regions by taking pixel-by-pixel difference over two or three consecutive frames from the video sequence. This method is highly adaptive to dynamic changes [5]. Optical flow methods employ flow vectors of moving object over time to detect moving regions in an image. Using these methods one can detect objects in motion even from a moving camera [6]. The only difficulty with optical flow methods is that they are computationally complex and their real time implementation requires specialized hardware.

The rest of the paper is organized as follows. Section 2 gives brief overview of few existing methods for object detection. Section 3 describes the proposed methodology in detail. Experimental results are discussed in section 4 while section 5 concludes the paper by presenting scope for future work.

2. LITERATURE SURVEY

Reference [7] utilizes textural edges in a frame for detecting the moving object. The edges extracted edges from each frame are represented as segments. Then after the authors have used Local Directional Pattern (LDP) to represent the texture information by considering the edge orientations in eight different directions. In reference [8] the authors M.Sankari and C. Meena have proposed a moving object detection methodology that makes use of background subtraction based on some statistical assumptions from the previous frame. The authors H. S Mohana, Aswatha Kumar, and Shiva Kumar have implemented a robust model-based tracker and classifier that utilizes Kalman Filtering in [9].

3. METHODOLOGY

To detect the moving object in the surveillance video captured from stationary camera the simplest method is frame difference. Frame differencing has an advantage of low computational complexity and ease of hardware implementation. In this method the unchanged part is eliminated during subtraction. In this paper we propose an improved and effective method for object detection utilizing framed difference and edge density map. The method is further divided in two stages: 1) Pre-Processing and 2) Exact Methodology.

3.1 Pre-Processing

This stage involves three steps: 1) Separating captured video into its constituent frames 2) Transforming RGB frames to grayscale images and 3) Contrast Stretching. The flow chart for Pre-Processing stage is shown in Fig-1.

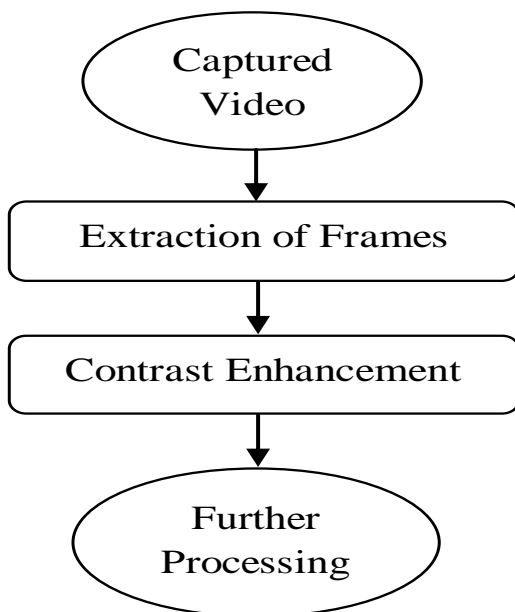


Fig-1: Steps in Pre-Processing Stage

The first step in pre-processing stage is extraction of frames contained in the captured video. The resultant frames will be RGB images. Working directly on a RGB image is computationally inefficient because it involves large number of working levels. To reduce the number of working levels we convert these RGB images into grayscale images. This task is accomplished using (1).

$$Gray(i, j) = 0.229 * R + 0.587 * G + 0.114 * B \quad (1)$$

Here Gray denotes the resultant output grayscale image while, R, G, and B indicate the red, green and blue components of the original RGB frame respectively [10]. To enable detection of moving objects from complex background as well as dim light environment we employ gray level stretching to highlight the foreground pixels [10]. The resultant images from first stage are shown in Fig-2, Fig-3 and Fig-4 respectively.



Fig-2: RGB Frame



Fig-3: Grayscale Frame



Fig-4: Result of Contrast Stretching

3.2 Exact Methodology

The flowchart for the proposed method is shown in Fig-5. To eliminate the stationary background we consider two consecutive frames. They are denoted as F_K and F_{K-1} respectively.

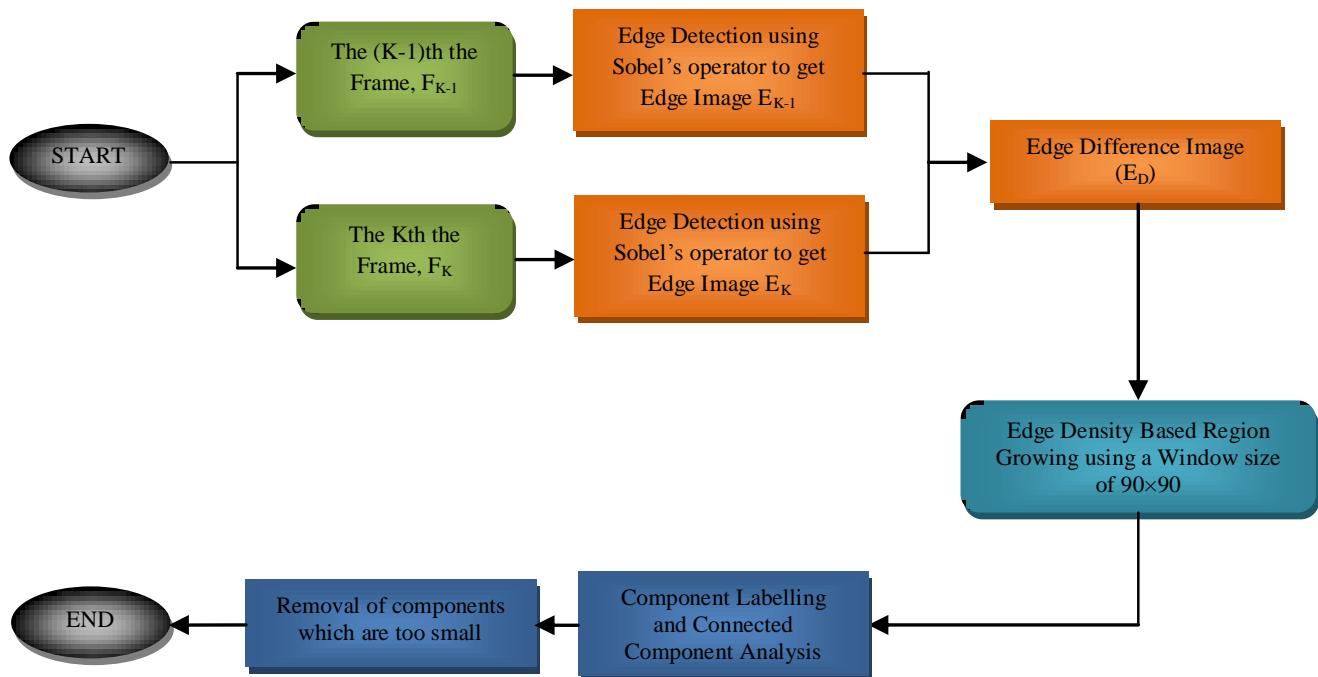


Fig-5: Flow Chart for Proposed Method

3.2.1 Edge Detection

In the next step we detect the edges present in each of these frames. Three different operators have been analyzed for detecting the edges present in frames. These are Prewitt's operator, Sobel's operator and Canny's operator [10]. The result of edge detection using each of these three operators is

shown in Fig-6. As visible using Prewitt's operator we may miss the edges that represent the moving object, while using Canny's operator results in detecting extra edges apart from the main object and will make our further processing task complex. Hence in this approach we have considered Sobel's operator for detecting edges.



(a) Output of Prewitt's operator

(b) Output of Sobel's operator

(c) Output of Canny's Operator

Fig-6: Result of Edge Detection using different operators

The result of applying edge detection on frames F_K and F_{K-1} is shown in Fig-7 and Fig-8 respectively. After Edge detection, the next step is to obtain the difference image to eliminate the edges of stationary objects. This is accomplished using (2).

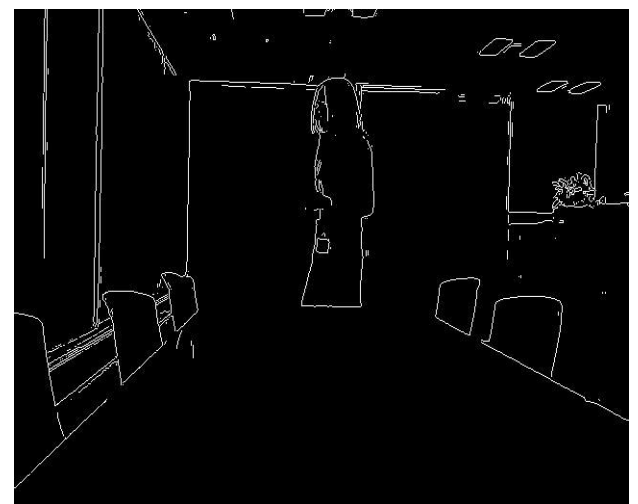
$$ED = E_{(K-1)Sobel} - E_{(K)Sobel} \quad (2)$$

3.2.2 Edge Density Based Region Growing

After this the next step is to apply Edge Density Based Region Growing Algorithm [11]. This is done to connect the regions containing the moving object which has been highlighted by computing edge difference. To implement this we count number of edge pixels over a pre-defined window. This number is then normalized by dividing it with total no of pixels in the window. The resultant value is compared with a threshold (τ). If the value is greater than the pre-defined threshold, all the pixels in the window are assigned a value of 1 else they are set to 0.

3.2.3 Connected Component Labeling

After connecting the regions containing the moving object the next step is to assign labels to these regions so that we can extract them one by one. The labeling is done based on 8-connectivity. Next we search for smallest rectangle which accommodates each of these regions as shown in Fig-11.

**Fig-7:** Edge image (E_K)**Fig-8:** Edge Image (E_{K-1})

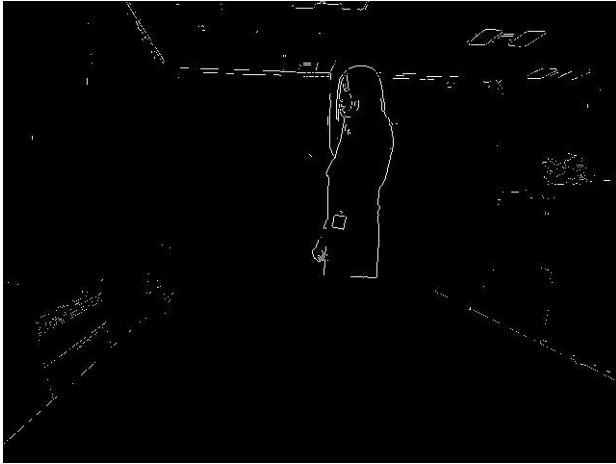


Fig-9: Edge Difference Image

Fig-10 shows the result of applying Edge Density Based Region Growing Algorithm on Edge Difference Image.

As we can observe, Fig-11 contains extra regions with dimensions much smaller than that of the detected moving object. To eliminate these regions we apply a series of morphological opening operations. The resultant modified image is shown in Fig-12.



Fig-10: Result of applying EDBRG

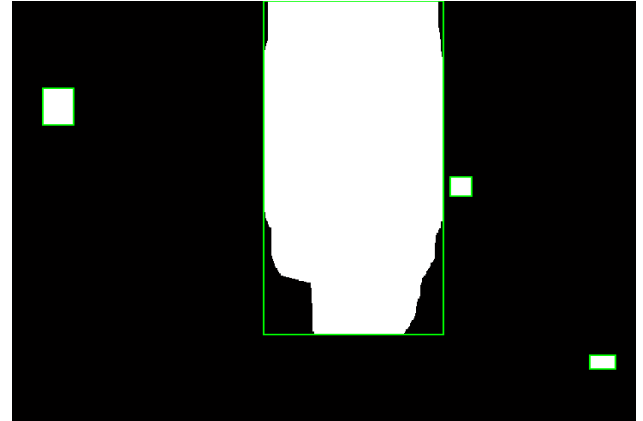


Fig-11: Connected Component Labeling



Fig-12: Modified Image after Morphological Opening

The region visible with white mark in Fig-12 is the region which contains the moving object. The exact moving object is extracted by retrieving corresponding pixels values from original image as shown in Fig-13.



Fig-13: Detected Moving Object

4. RESULTS AND DISCUSSIONS

The proposed method is implemented on Matlab 7.8.0, R2009A, on a PC having Intel Core-i3 Third generation CPU with 4-GB RAM capacity running on Windows-8 Operating System. The size of the frames considered for processing is 480×640 . Fig-14 shows additional simulation results. Two different videos have been considered: 1) A woman moving in her office 2) A cylinder rolling on a flat platform. The total no of frames analyzed in first video is 179 while that in second video is 133. The frame pairs considered for simulation are (15, 17), (35, 37), (119,121) respectively from first video and (50, 52), (75, 77) and (90, 92) respectively from second video.

5. CONCLUSIONS

In this paper a unique method for detecting moving object based on edge detection, frame difference and edge density based region growing is presented. Use of contrast stretching in pre-processing before edge detection makes the method robust enough for detecting the moving objects in complex backgrounds. The main advantage of the method is that it takes into account the edges present in the frames and hence operates directly on binary images. This reduces the computation time as in a binary image we have to work with only two levels. Further improvement can be made by carrying out experiments with different window sizes and thresholds while applying the edge density based region growing algorithm. As a part of future work we look forward to integrate object tracking with the proposed algorithm.

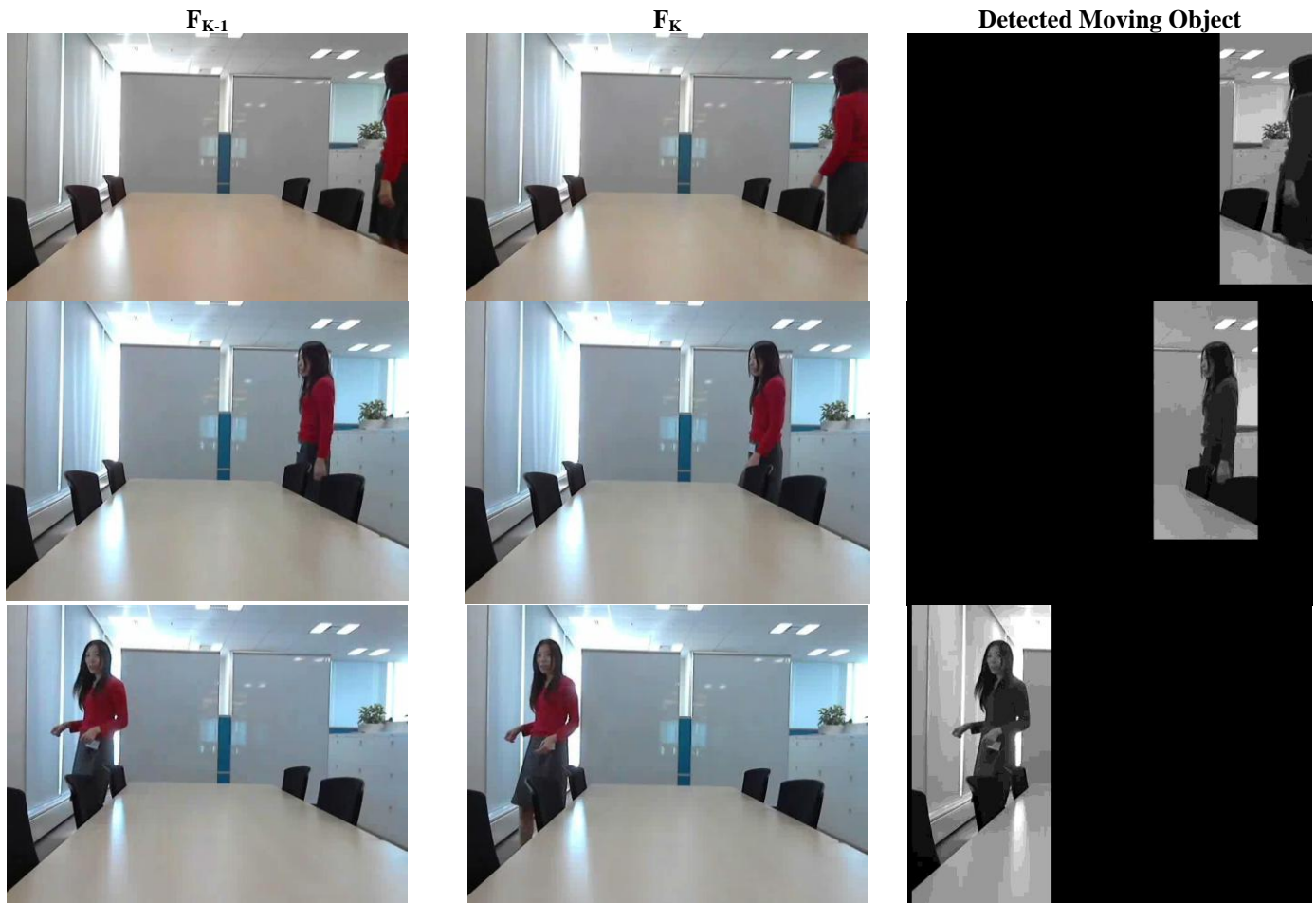


Fig 14 (a): Additional Simulation Results at different positions; Human Being

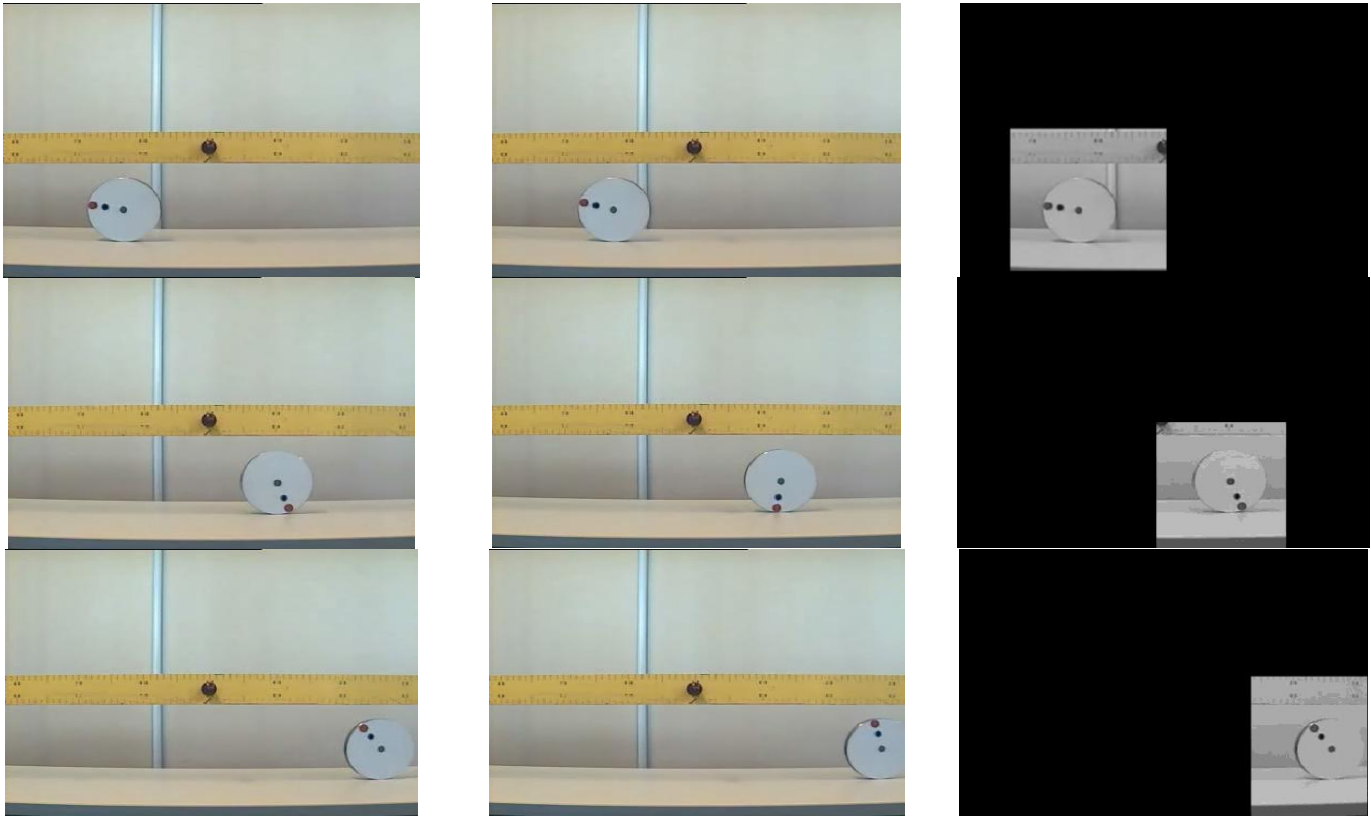


Fig-14 (b): Additional Simulation Results at different positions; Moving Cylinder

REFERENCES

- [1]. Kim Changick, "Fast and Automatic Video Object Segmentation and Tracking for Content-Based Applications", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 9, pp. 1945-1062, July-1999.
- [2]. K. Wu, T. Xu, H.Zhang, J. Song, "Overview of Video-Based Vehicle Detection Technologies", Sixth International Conference on Computer Science and Education, pp. 821-825, August-2011.
- [3]. R. Collins, A. Lipton, T. Kanade, "Introduction to Special Section on Video Surveillance", In the Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 745-746, 2000.
- [4]. L. Maddalena, A. Petrosino, "A Self-Organizing Approach to Background Subtraction for Visual Surveillance Applications", IEEE Transactions on Image Processing, Vol. 17, No. 7, pp. 1168-1177, 2008.
- [5]. Zhu Minghan, and Luo Dayong, "Moving Objects Detection and Tracking Based on Two Consecutive Frames Subtraction Background Model". Computer Measurement and Control, 2006, pp. 1004-1006.
- [6]. B. Horn, and B. Schunck, "Determining optical flow". Artificial Intelligence, 1981, vol. 17, pp.185-203.
- [7]. Taskeed Jabid, Tahseen Mohammad, Tanveer Ahsan, M. Abdullah-Al-Wadud, Oksam Chae, "An Edge-Texture based

Moving Object Detection for Video Content Based Application", In the Proceedings of 14th International Conference on Computer and Information Technology (ICCIT 2011), December-2011.

[8]. M.Sankari and C. Meena, "Adaptive Background Estimation and object detection applying in Automated visual surveillance" (IJCSIS) International Journal of Computer Science and Information Security, Vol. 8, No. 4, July 2010.

[9]. H.S. Mohana, Aswatha Kumar and Shiva kumar Malnad College of Engineering, Hassan, Karnataka, "Vehicle Counting And Classification Using Kalman Filter And Pixel Scanner Technique And Its Verification With Optical Flow Estimation", Global Journal of Computer Science and Technology 46 Vol. 10 Issue 8 Ver. 1.0 September 2010.

[10]. R.C Gonzalez, R.E Woods, Digital Image Processing, 3rd Edition, Prentice Hall, 2007.

[11]. Jianbin Jiao, QixiangYe, Qingming Huang, "A configurable method for multi-style license plate recognition", Pattern Recognition 42, 2009, pp: 358-369.