Moving object detection and velocity estimation using MATLAB

T. Malavika[1], M. Poornima[2]

Assistant Professor,MVJ College of Engineering,Bangalore,Karnataka,India

Abstract — **Moving object detection is an important step in the video processing. We propose an implementation of image subtraction algorithm to detect a moving object. Detection of object motion has been implemented and the velocity is been calculated. The algorithm is processed with matlab software and we calculated the distance, frame per time, velocity. This can be used in speed measurements in traffic signal monitoring systems.**

Keywords- **moving object detection, velocity estimation, subtraction algorithm**

1. INTRODUCTION

Moving object detection in real time is a challenging task in visual surveillance systems. It often acts as an initial step for further processing such as classification of the detected moving object. In order to perform more sophisticated operations such as classification, we need to first develop an efficient and accurate method for detecting moving objects. A typical moving object detection algorithm has the following features: (a) estimation of the stationary part (background) (b) obtaining difference images of frames (c) Applying the threshold value (d) tracking (e) estimate velocity.

Moving object detection is always an important task in this world of technology. Moving object detection plays an important role in automatic visual surveillance, tracking system and also to avoid collision. The use of smart object detection, tracking and classification algorithms are not limited to video surveillance only. Other application domains also benefit from the advances in the research on these algorithms. Some examples are virtual reality, video compression, human machine interface, augmented reality, video editing and multimedia databases etc.

Optical flow method has been used for detecting moving objects in image sequences [5]-[8]. For example, adaptive optical flow for person tracking is dependent on being able to locate a person accurately across a series of frames. Optical flow can be used to segment a moving object from a scene, provided the expected velocity of the moving object is known; but successful detection also relies on being able to segment the background. Other works such as moving objects segmentation using optical flow estimation presented a method for the segmentation of moving objects, where a powerful variation method using active contours for computing the optical flow is used. However, the high computational time to extract the optical flow and the lack of discrimination of the foreground from the background, make this method unsuitable for real time processing.

2. MOVNG OBJECT DETECTION

Moving object detection is useful in various fields like border security forces, traffic signal analysis etc. There are various algorithms used for detecting moving objects. The background subtraction detects moving objects by subtracting estimated background models from images. A moving object detection and tracking system with a static camera has been developed to estimate velocity, distance parameter. For that application we presented subtraction algorithm after performing, noise removal filter, edge detection and calculate centroid, velocity and distance of moving object in scene.

In order to detect a movement in a video, the video is taken using a static camera. The video is split into frames. The frame size is 256x256 pixels. The first frame, which is called "reference", represents the reference pixel values for comparison purpose, and the second frame, which is called the "input", contains the moving object. The two frames are compared and the differences in pixel values are determined. A threshold is applied and the pixel is identified as background or foreground. If the "difference" average pixel value is smaller than a certain threshold value, then the output image background will be white (pixel value is 255); otherwise, the background will be black (pixel value is 0). Now the pixels are classified and the motion is tracked in the video.

3. BACKGROUND INFORMATION

Background modeling is used to adapt the change in the dynamic environment. It is very challenging for every background subtraction based method. Traditional method utilizes temporal differencing or optic flow based method. The temporal differencing method utilizes two or more consecutive frames to extract moving regions. Optical flow uses the characteristics of flow vectors of the moving objects over time. If the background contents are not visible for long time, all these methods will fail to generate accurate background model. These methods are vulnerable and prone to false detection, if the temporal

changes are generated by noise or illumination change due to weather condition. Considering the robustness, suitability and reduced data access rate, current research has focused on the benefit of edges structure in processing sequence images. But edges show shape and size variation within frames due to the changes in illumination and noise. Moreover, the variations for different edges are not the same. Without considering this variation from the environment, detectors output cannot be reliable. To solve this problem, the edge is represented as segment that allows incorporating knowledge to every edge segment about its motion, shape, position, and size variation. And thus, it helps to model the environment using the statistics of all edge segments. Thus, in this proposed method, edges are treated differently, i.e., segments having small motion variation statistic will be matched with a high threshold, while a low threshold may be used for those having large motion variation.

Background subtraction is particularly a commonly used technique for motion segmentation in static scenes. It attempts to detect moving regions by subtracting the current image pixel-by-pixel from a reference background image that is created by averaging images over time in an initialized period. The pixels for which the difference is above the threshold are classified as foreground. After creating foreground pixel map, some morphological post processing operations such as erosion, dilation and closing are performed to reduce the effects of noise and enhance the detected regions. The reference background is updated with new images over time to adapt to dynamic scene changes.

Background subtraction technique often relies on deriving the probabilistic model of the background. When a new image is captured, the difference between the image and background model is computed for moving object detection. Unfortunately, the derivation of the model is complex and computationally expensive. Although alternative approaches not based on probabilistic modeling of the background image exist, they are very specific to applications. Therefore, most of existing approaches for moving object detection are computationally heavy and subject to large delays, adversely affecting the performance of real-time surveillance.

Background subtraction is a computational vision process of extracting foreground objects in a particular scene. A foreground object can be described as an object of attention, which helps in reducing the amount of data to be processed as well as provide important information to the task under consideration. Often, the foreground object can be thought of as a coherently moving object in a scene. We must emphasize the word coherent here because if a person is walking in front of moving leaves, the person forms the foreground object while leaves though having motion associated with them are considered background due to its repetitive behavior. In some cases, distance of the moving object also forms a basis for it to be considered a background, e.g if in a scene one person is close to the camera while there is a person far away in background, in this case the nearby person is considered as foreground while the person far away is ignored due to its small size and the lack of information that it provides. Identifying moving objects from a video sequence is a fundamental and critical task in many computer-vision applications. A common approach is to perform background subtraction, which identifies moving objects from the portion of video frame that differs from the background.

Background subtraction is a class of techniques for segmenting out objects of interest in a scene for applications such as surveillance. There are many challenges in developing a good background subtraction algorithm. First, it must be robust against changes in illumination. Second, it should avoid detecting non-stationary background objects and shadows cast by moving objects. A good background model should also react quickly to changes in background and adapt itself to accommodate changes occurring in the background such as moving of a stationary chair from one place to another. It should also have a good foreground detection rate and the processing time for background subtraction should be real-time.

3.1. Edge Detection

Edge detection is the first step to recover information from images. Edges are the significant local changes of intensity in an image. Edges typically occur on the boundary between two different regions in an image. Edge also can be defined as discontinuities in image intensity from one pixel to another. A typical edge detector has the following steps: (a) it suppresses noise as much as possible, without destroying the true edges; (b) it applies a filter to enhance the quality of the edges in the image, (c) it determines which edge pixels should be discarded as noise and which should be retained, (d) it determines the exact location of an edge. An optimal edge detector should satisfy the following criteria: (a) the optimal detector must minimize the probability of false positives (detecting spurious edges caused by noise) as well as that of false negatives (missing real edges), (b) the edges detected must be as close as possible to the true edges, (c) the detector must return one point only for each true edge point; that is, it minimizes the number of local maxima around the true edge created by noise.

3.2. Canny Edge Detector

The Canny edge detector is one of the most commonly used image processing tool to detect edges from image. It has the following steps:

3.2.1 Gray Scale Conversion

In photography and computing, a grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest. To convert any color to a grayscale representation of its luminance, first one must obtain the values of its red, green, and blue (RGB) primaries in linear intensity encoding, by gamma expansion. Then, add together 30% of the red value, 59% of the green value and 11% of the blue value.

3.2.2 Noise Reduction

The Canny edge detector uses a filter based on the first derivative of a Gaussian, because it is susceptible to noise exists in raw unprocessed image data. Thus, at first the raw image is convolved with a Gaussian filter. The result is a slightly blurred version of the original which is not affected by a single noisy pixel to any significant degree.

3.2.3 Gradient Computation

The edge may point to different directions .The edge detection first derivative is obtained and the point of maxima is calculated.

The Canny algorithm uses an optimal edge detector based on a set of criteria which include finding the most edges by minimizing the error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only once when a single edge exists for minimal response .

4. OVERVIEW OF THE SYSTEM

The aim is to build a robust and novel moving object detection algorithm that can detect object in a variety of challenging real world scenarios. The video is organized into frames. Image processing tool box is used for obtaining the object by subtracting the frames. The movement of the object is described using edge detection techniques. Experiment datasets will be collected for these scenarios for testing the performance of the algorithm.

1. The input video is converted into frames. We capture two consecutive frames, i.e. frame N and frame N+1.
2. We convert each of these frames to gray scale.
3. We subtract frame N from frame N+1, to generate the difference image.
4. We run canny edge detection technique which is preferred over the other techniques like Prewitt and Sobel image in order to remove noise and to detect the edges.
5. We identify moving edge segments and also draw a bounding box over it.

6. The centroid is then calculated.
7. We can then calculate the distance moved by the object by calculating distance moved to the total frame rate.

From figure 1, we have applied the subtraction algorithm and thereby able to track moving objects effectively.
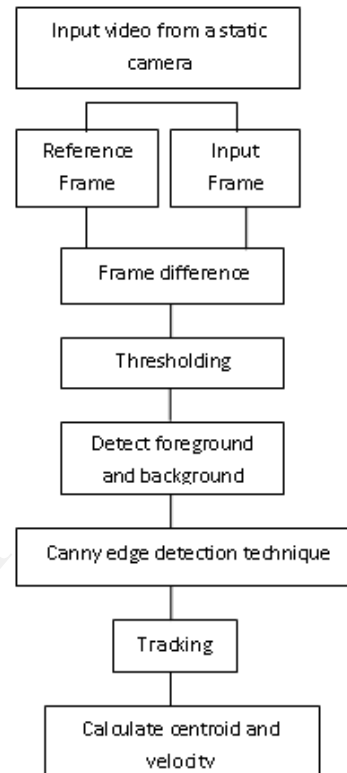

Figure 1.System overview

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.

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

where 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.

Velocity = Distance Travelled/Frame Rate

5. RESULTS AND ANALYIS

This method was tested on several video sequences. All images were of size 320×240 with background motion, illumination change. The proposed system was able to detect almost all of the moving objects in the sequences. MATLAB was used as an image processing environment tool. The video is split into frames and then filtering is done in order to remove noise. Then canny edge detection technique is applied. Then subtraction method is applied. The moving object is identified and is separated from the background. Every edge segment is matched using a common threshold. Also every background segment has its own motion statistic. An edge segment that has high movement statistic will be matched with a wider search area and low threshold. Non-moving edge segments will benefit from high threshold and narrow search region.

Figure 2 shows a sample input frame obtained from the given video and the algorithm is applied and finally the output is obtained .The output clearly draws a bounding box on the moving object present in the scene and thus the velocity can be calculated. After tracking moving object we calculated time per frame of video and then with the help of centroid of moving object which is like a reference point we calculate distance of moving object .We also calculate velocity of moving object.

In matlab we have implemented this image subtraction algorithm .Whenever in the example, the man is moving, it will detect and track movement. Then we calculated centroid using matlab command and then found out the distance according to reference point. The detection output is more accurate and thus can significantly improve the performance of video surveillance based applications.
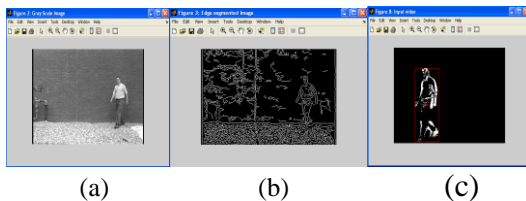

(a) (b) (c)

Figure 2. (a) A sample input image frame (b) canny edge detection is performed (c) Moving edge segments using our proposed method.

6. CONCLUSION AND FUTURE WORK

This paper illustrates the suitability of using this model to detect moving objects for the video surveillance based applications. The strength of our approach lies in the ability to separate background edge segments from moving edge segments. The velocity and speed of the moving object can also be calculated. In our future work, we will incorporate multi image detection and calculating speed for the multiple moving objects present in the scene for more sophisticated vision based applications like airport security, activity recognition, etc. This work can also be used along with other detection techniques to detect the kind of the object.

REFERENCES

[1]. J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[2]. M. A. A. Dewan, M. J. Hossain, and O. Chae. Background independent moving object segmentation for video surveillance. *IEICE Transactions*, 92-B(2):585–598, 2009.

[3]. D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. K. Jain. A background model initialization algorithm for video surveillance. In *in proc. IEEE ICCV 2001, Pt.1*, pages 733–740, 2001.

[4]. Mahbub Murshed, Adin Ramirez, Oksam Chae. Statistical Background Modeling: An Edge Segment based Moving Object Detection Approach.in *Seventh IEEE International Conference on Advanced Video and Signal Based Surveillance, 2010.* pages 300-306, May. 2010.

[5]. J. H. Duncan, and T. C. Chou, "Temporal edges: The detection of motion and the computation of optical flow," in *Proc. IEEE 2nd Int. Conf. Computer Vision*, Florida, USA, Dec. 1988, pp. 374-382.

[6]. S. Denman, V. Chandran, and S. Sridharan, "Adaptive Optical Flow for Person Track ng," in *Proc. Digital Image Computing: Techniques and Applications*, Cairns, Australia, Dec. 2005, pp. 44-50.

[7]. F. Ranchin, and F. Dibos, "Moving objects segmentation using optical flow estimation", in *Proc. Workshop on Mathematics and Image Analysis*, Paris, France, Sep. 2004.

[8]. L. Li, W. Huang, I.Y.H. Gu, and Q. Tian, "Foreground object detection in changing background based on color co-occurrence statistics," in *Proc. IEEE 6th Workshop Applications of Computer Vision*, Florida, USA, Dec. 2002, pp. 269-274.

[9]. W. k. Pratt, "*Digital image processing,"* fourth Edition, Pixel Soft, Inc.Los Altos, California, Willy.

[10].R. C. Gonzalez University of Tennessee, Richard E. Woods MedData Interactive, "*Digital Image Processing,"* Second Edition, Prentice Hall.