ISSN: 2250–3676
[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-3, 731 – 736

APPLICATION OF SAD ALGORITHM IN IMAGE PROCESSIG FOR MOTION DETECTION AND SIMULINK BLOCKSETS FOR OBJECT TRACKING

Menakshi Bhat[1], Pragati Kapoor[2], B.L.Raina[3]

[1]*Assistant Professor, School of Electronics & Communication Engg. ,Lingaya's University, Faridabad India*
bhatmenakshi@gmail.com
[2]*Associate Professor, School of Electronics & Communication Engg. ,Lingaya's University, Faridabad India*
pragati_ahuja@yahoo.com
[3]*Assistant Professor, School of Electronics & Communication Engg. ,Lingaya's University, Faridabad India*
rbushan@rediffmail.com

Abstract

The process of locating a moving object in time that is visualized by camera and used in surveillance, animation and robotics is usually associated to the video tracking. Tracking is defined as the set of constraints that describes the better action performed. The key difficulty in video tracking is to associate target locations in consecutive video frames, especially when the objects are moving fast relative to the frame rate. Here, a video tracking system is been being employed in which motion model describes how the image of the target might change for different possible motions of the object to track. Tracking algorithm adopted for this system is to analyze the video frames to estimate the motion parameters. These parameters characterize the location of the target. In this paper Simulink is integrated with MATLAB to build a model for object tracking and data transfer is easily handled between the programs. Here the Simulink based customizable framework is designed for rapid simulation, implementation, and verification of video and image processing algorithms and systems. The aim of this paper work is to implement an efficient methodology to track the moving object present inside the moving videos. Using blockset, the framework is carried out using the image processing steps such as; Video processing, frame display, background subtraction, edge Detection, segmentation and tracking. Videos and images from open source can be accomplished and so it can be easily implemented. Implementation of this methodology using Simulink blockset is more useful for security and video surveillance. This system also provides additional services such as information about the location and identity of objects at different points in time which forms the basis for detecting unusual object movement s. Tracking the objects, true position is done by tracking its state using region filtering and this uses information from the current blob and the previous object state to create an estimate of the objects in new state.

Index Terms: *Image Processing, Motion detection, Object Tracking*

--- *** ---

1. INTRODUCTION

Continuous-scene monitoring applications, such as ATM booths, parking lots or traffic monitoring systems, generate large volumes of data. Recording and archiving such volumes of data is a real problem, and one way to solve this is to reduce the size of the data stream right at the source. In addition to traditional methods for compressing individual video images, we could identify and record only "interesting" video images, such as those images with significant amounts of motion in the field of view. That could significantly help reduce the data rates for surveillance-specific applications. In this section, we'll illustrate the design flow of a motion detection system based video - surveillance system. The system will be able to identify and record only "interesting" video frames containing motion. The development process will follow the following three steps:

1. Functional Requirements
2. System modelling and simulation
3. Code generation and implementation

Video Surveillance

ISSN: 2250–3676
[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-3, 731 – 736

In this section we will show how to create the line detection model, and how it can be integrated in Simulation and Real-Time Implementations.

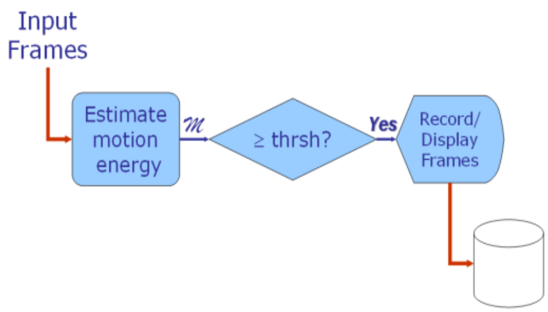

Figure1. Motion Detection Process

2. MOTION ENERGYESTIMATION

The Motion Energy estimation is based on the calculation of the **S**um of **A**bsolute **D**ifferences (SAD)" according to the following equation:

$$SAD = \sum_i \sum_j \left| I_k(i,j) - I_{k-1}(i,j) \right|$$

where:

$I_k(i,j)$ - Current kth Frame

$I_{k-1}(i,j)$ -Previous (k-1)th Frame

Functional Description

The Video Surveillance system block diagram is shown in **Error! Reference source not found.**. The source video is displayed and played back through the system. This incoming video stream is processed by a subsystem that estimates the motion within the scene and captures the interesting video frames. The system can display either the recorded video frame or the Absolute Differences (AD) image.

The user can configure the motion energy threshold value and select which image will be displayed (Display Control). The system outputs include, besides the displayed image, the following features:

- A graph of the motion energy as a function of time. This graph displays the threshold value as well.
- The Frame count of the recorded image
- An output signal triggered by motion detection (Trigger)

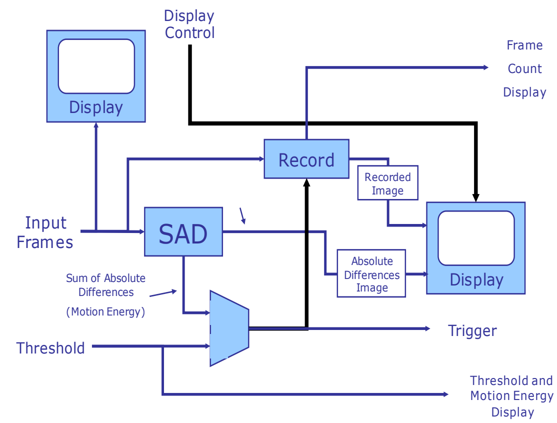

Figure 2: Video Surveillance Block Diagram

To detect changes between two images, the images are compared pixel by pixel. For this purpose we form a difference image. Suppose we have a reference image containing only stationary components. Consider a sequence of image frames $f(x,y,t_1)$, $f(x,y,t_2)$ ……….. $f(x,y,t_n)$ and let $f(x,y,t_1)$ be the reference image. An accumulative difference image ADI is formed by comparing this reference image with every subsequent image in the sequence. A counter for each pixel location in the accumulative image is incremented, every time a difference occurs at that pixel location between the reference and an image in the sequence.

ADI corresponds to the two types of accumulative difference images: positive and negative.

Sum of absolute differences (**SAD**) is a widely used as simple algorithm for measuring the similarity between image blocks. It works by taking the absolute difference between each pixel in the original block and the corresponding pixel in the block being used for comparison. These differences are summed to create a simple metric of block similarity.

The sum of absolute differences may be used for a variety of purposes, such as object recognition, the generation of disparity maps for stereo images, and motion estimation for video compression.

3. OBJECT TRACKING SUBSYSTEM

Object tracking is used to describe the process of recording movement and translating that movement onto a digital model. Simulink with Video and Image processing blockset enable to run fast simulations for real-time embedded video, vision, and imaging systems. It can create executable specifications for communicating the system to downstream design teams and to provide a golden reference for verification throughout the design process. The amount of work does not vary with the complexity or length of the performance to the same degree as when using traditional techniques.

The aim of this paper work is to implement an efficient methodology to track the moving object present inside the moving videos. Using blockset. Our paper is in the same vein as given by [1].Here the framework is carried out using the image processing steps such as; Video processing, frame display, background subtraction, edge Detection, segmentation and tracking such as :First, the videos are separated as frames and pre-processing method is used for the color conversion to subtract the foreground objects from the background, and background subtraction is used to find the total or sudden change in intensity in the video. Edge detection is performed as the middle wok and to extract the boundary of the object from the background segmentation module is executed. Finally tracking will be carried out. The main objective of the object tracking is to detect and track the moving xobject through video sequence. Most of the image processing techniques covered here are tracking related experimental results.

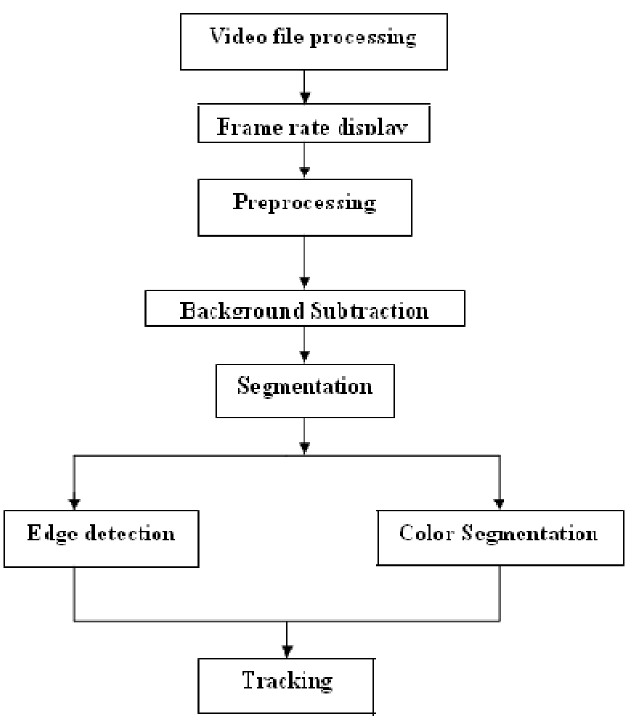

Figure 3: Flow Chart

This proposed framework combines the existing and recent techniques of video processing techniques for tracking system. The output of the system delivers a Mat lab-based application development platform intended for tracking system. It allows the user to investigate, design, and evaluate algorithms and applications using object based videos. It offers standardized, which do not require detailed knowledge of the target hardware and is based on the following:

Pre-processing

Pre-processing is mainly used to enhance the contrast of the image, removal of noise and isolating objects of interest in the image. Pre-processing is any form of signal processing for which the output is an image or video, the output can be either an image or a set of characteristics or parameters related to image or videos to improve or change some quality of the input. Pre-processing helps to improve the video or image such that it increases the chance for success of other processes.

Frame Rate Display

Frame rate or frame frequency is the frequency at which an imaging device produces unique consecutive images, frames that applies equally well to compute graphics, video cameras, film cameras and motion capture system. Frame rate is most often expressed in frames per second and is also uttered in progressive scan monitors as Hertz (Hz).The frame rate display block calculates and displays the average update rate of the input signals.

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 model.

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[10]. 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.

4. SEGMENTATION

The processing of subdividing the image into its constituent parts or object is called image segmentation.

Segmentation based on Edge Detection

Edge detecting is a collection of very important local image pre-processing method used to locate changes in the intensity function. An edge is a property attached to an individual pixel and is calculated from the image function behavior in a neighborhood of that pixel. Edge takes a grayscale image as its input, and returns a binary image of the same size as the input image, with 1's where the function finds edges and 0's elsewhere.

- Roberts operators
- Prewitt Operator
- Sobel operator
- Canny operator

The Sobel method finds edges using the Sobel approximation to the derivative. It returns edges at those points where the gradient of image is maximum. The Canny method finds edges by looking for local maxima of the gradient of image. The gradient is calculated using the derivative of a Gaussian filter. The Roberts method finds edges using the Roberts approximation to the derivative. It returns edges at those points where the gradient of the image is maximum. The Edge detection block outputs a binary image with the edges shown in white. This output is displayed in the Edges window. The compositing block accepts the original video frames shown in the original window and the output of the edge detection block as inputs at its Image1 and Mask ports, respectively. The input to the Mask port tells the compositing block which pixels to highlight. As a result, the model displays a composite image in the overlay window, where the original pixel values are overwritten by the white edge values.

Edge Detection Methods

Three most frequently used edge detection methods are used for comparison. These are

- Roberts Edge Detection
- Sobel Edge Detection
- Prewitt edge detection
- Canny Edge Detection.

The details of these methods follow as:

1. The Roberts Detection: The Roberts Cross operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. It thus highlights regions of high spatial frequency which often correspond to edges. In its most common usage, the input to the operator is a grayscale image, as is the output. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point.

+1	0
0	-1

Gx

0	+1
-1	0

Gy

Fig. 4. Roberts Mask

2. The Prewitt Detection: The Prewitt edge detector is an appropriate way to estimate the magnitude and orientation of an edge. Although differential gradient edge detection needs a rather time consuming calculation to estimate the orientation from the magnitudes in the x and y -directions, the compass edge detection obtains the orientation directly from the kernel with the maximum response. The Prewitt operator is limited to 8 possible orientations, however experience shows that most direct orientation estimates are not much more accurate. This gradient based edge detector is estimated in the 3x3 neighborhood for eight directions. All the eight convolution masks are calculated. One convolution mask is then selected, namely that with the largest module

-1	+1	+1
-1	-2	+1
-1	+1	+1

0°

+1	+1	+1
-1	-2	+1
-1	-1	+1

45°

Fig. 5. Prewitt Mask

3. The Sobel Detection: The Sobel operator performs a 2-D spatial gradient measurement on an image and so emphasizes regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. In theory at least, the operator consists of a pair of 3x3 convolution kernels as shown in Figure 4. One kernel is simply the other rotated by 90o.This is very similar to the Roberts Cross operator. The convolution masks of the Sobel detector are given below;

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Fig. 6. Sobel Mask

4. Canny Edge Detection Algorithm

The Canny edge detection algorithm is known to many as the optimal edge detector. Canny's intentions were to enhance the many edge detectors already out at the time he started his work. He was very successful in achieving his goal and his ideas and methods can be found in his paper, "A Computational Approach to Edge Detection". In his paper, he followed a list of criteria to improve current methods of edge detection. The first and most obvious is low error rate. It is important that edges occurring in images should not be missed and that there be no responses to non-edges. The second criterion is that the edge points be well localized. In other words, the distance between the edge pixels as found by the detector and the actual edge is to be at a minimum. A third criterion is to have only one response to a single edge. This was implemented because the first two were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the canny edge detector first smoothes the image to eliminate and noise. It then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (non -maximum suppression). The gradient array is now further reduced by hysteresis. Hysteresis is used to track along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds and if the magnitude is below the first threshold, it is set to zero (made a non edge). If the magnitude is above the high threshold, it is made an edge. And if the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2.

CONCLUSION & RESULTS

The objective of tracking is to associate target objects in consecutive video frames. It is more useful for military entertainment, sports, and medical applications and for validation of computer vision and robotics. Videos and images from open source can be accomplished and so it can be easily implemented and its usage is wider. Implementation of this methodology using Simulink blockset is more useful for security and video surveillance. This system also provides additional services such as information about the location and identity of objects at different points in time is the basis for detecting

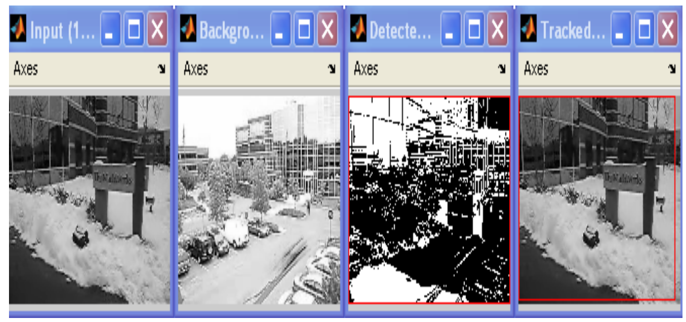

Figure7: Object Tracking

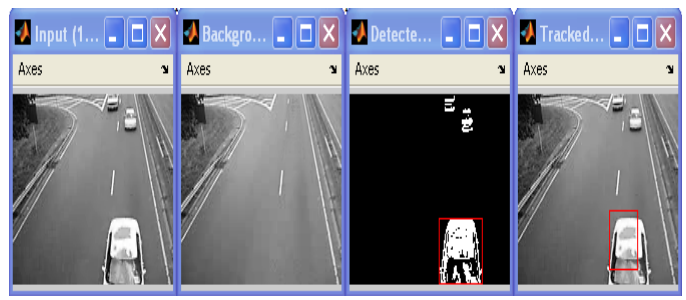

Figure 8: Object Tracking

Figure 9: Motion Estimation

REFERENCES

[1] Zhou, Q.; Aggarwal, J. K.; "Tracking and classifying moving jects from video", Proc of 2nd IEEE Intl Workshop on Performance Evaluation of Tracking and Surveillance (PETS'2001), Kauai, Hawaii, USA (December 2001).

[2] Stauffer C.; "Estimating tracking sources and sinks", Proc of 2nd IEEE Workshop on Event Mining (in conjunction with CVPR'2003),4, Madison, Wisconsin (June 2003).

[3] Zhenggao Xu. Applications Engineer Implementing "Video Tracking. Systems on DM6437 EVM Using. MATLAB and Simulink". A Model-Based Design Approaches.

[4] M.C. Shin, D. Goldgof, and K.W. Bowyer ."Comparison of Edge Detector Performance through Use in an Object Recognition Task". Computer Vision and Image Understanding, vol. 84, no. 1, pp. 160-178, Oct. 2001.

[5] Iasonas Kokkinos, and Petros Maragos (2009),"Synergy between Object Recognition and image segmentation using Expectation and Maximization Algorithm"., IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), Vol. 31(8), pp. 1486-1501, 2009.

[6] Y.Ramadevi, B.Kalyani, T.Sridevi(2010), " Synergy between Object Recognition and Image Segmentation", International Journal on Computer Science and Engineering, Vol. 02, No. 08, 2010, 2767-2772

[7] N.Senthilkumarn, R.Rajesh(2009), "Edge Detection Techniques for Image Segmentation- A Survey of Soft Computing Approaches", IJRTE, vol1,No2, 2009 250-254.

[8] S. Wachter and H.-H. Nagel, "Tracking persons in monocular image sequences," Computer Vision and Image Understanding, vol. 74, pp. 174–192, 1999.

[9] Wen-Xiong Kang, Qing-Qiang Yang, Run-Peng Liang (2009), "The Comparative Research on Image Segmentation Algorithms," First International Workshop on Education Technology and Computer Science.

[10] X. Liu, J. Samarabandu. Multiscale Edge-based Text Extraction from Complex Images. In Proceedings of IEEE International Conference on Multimedia and Expo, IEEE Press, pp. 1721–1724, 2006.

[11] Implementation of Object Tracking System using region filtering algorithm based on Simulink Blocksets by Dr.P.Subashini / International Journal of Engineering Science and Technology (IJEST) August 2011.

[12] R. C. Gonzalez and R. E. Woods. "Digital Image Processing". 2nd ed.Prentice Hall, 2002.

[13]Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain by Mamta Juneja , Parvinder Singh Sandhu, International Journal of Computer Theory and Engineering, Vol. 1, No. 5, December, 2009.