Real Time Movement Detection for Human Recognition

[1]Sreedevi M, [2]Yaswanth Kumar Avulapati, [3]Anjan Babu G, [4]Sendhil Kumar R

Abstract–Traditional video surveillance takes a huge amount of storage space. Recording everything captured by a surveillance camera consumes excessively the storage space and hence limits the duration of video that can be stored. In addition, recording everything makes it time-consuming for a human to review the stored video. All these disadvantages limit the effectiveness of traditional video surveillance.

To solve these problems recording only video that contains import this project uses a robust motion detection algorithm for real time motion detection by considering and information, i.e., video that contains motion in the scene. This can be done with a web camera and a DSP *algorithm that detects motion*. When the lighting that frame difference signal. Once the motion robustly distinguishes motion from lighting distinguish real motion from lighting Changes.

So an algorithm is condition Changes, it is difficult to is detected, it is required to classify the object in motion is human or nonhuman developed changes by removing the mean from the illumination changes. The correlation method used to classify the object as human or nonhuman gives the same performance level as the existing methods.

Keywords-- Surveillance, Motion Detection, Motion Detection

I. INTRODUCTION

Traditional video surveillance takes a huge amount of storage space. Recording everything captured by a surveillance camera consumes excessively the disadvantages limit the effectiveness of traditional video surveillance. To solve these problems recording only video that contains important information, i.e., video that contains motion in the scene. This can be done with a web camera and a motion detection algorithm that detects motion. When the lighting condition changes, it is difficult to distinguish real motion from lighting changes. So an algorithm is developed that robustly distinguishes motion from lighting changes by removing the mean from the frame difference signal. Once the motion is detected, the object in motion is classified as human or nonhuman. This project uses a robust motion detection algorithm for real time motion detection by considering illumination changes also. Will activate a warning system and capture the live streaming video. Continuous scene monitoring applications, such as ATM booths, parking lots or traffic monitoring systems, generate large volumes of data. Recording Real-time detection of moving objects is very important for video surveillance 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.

[1]Assistant Professor, Dept of Computer Science, S.V.University, Tirupati
[2]Academic Consultant, Dept of Computer Science, S.V.University, Tirupati
[3]Associate Professor, Dept of Computer Science, S.V.University, Tirupati
[4]Assistant Professor Dept of CS, Tirumalai Engg College, Kanchipuram

II. SYSTEM ARCHITECTURE

Capturing the live video feed through a web cam: To detect motion first monitored and kept under surveillance have to be captured. This is done by using a web cam which continuously provides a sequence of video frames in a particular speed of FPS (frames per second).

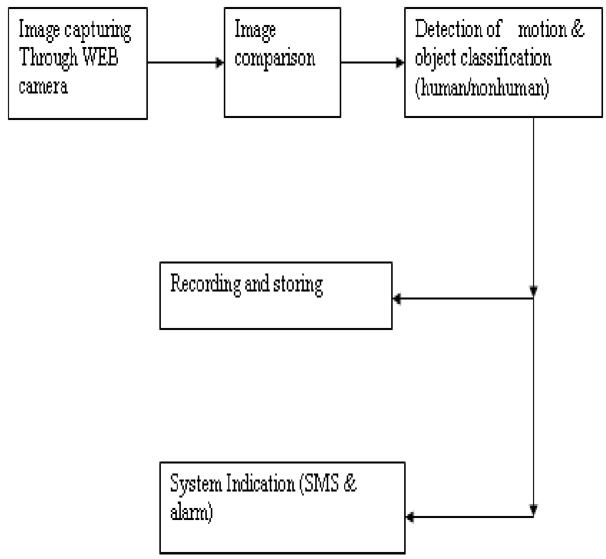

Fig 1. System Architecture

- Comparing the current frames captured with previous frames to detect motion: For checking whether any motion is present in the live video feed, the live video frames being provided by the web cam must be compared with each other by using motion detection algorithms block based SAD and 2Dcross correlation algorithms.
- Storing the frames on the memory if motion is detected: If motion is being detected, it is required to store such motion so that the user can view it in the near future. This also helps the user in providing a legal proof of some inappropriate activity since a video coverage can be used as a proof in the court of law.
- Human or Nonhuman recognition: Image processing is done on captured frames and object is identified as human or non human.
- Indicating through SMS and alarm when the motion is detected: Once motion has been detected in the live stream, the software will activate a warning system and capture the live streaming video and creates an active alert by sending a message to the cell phone.

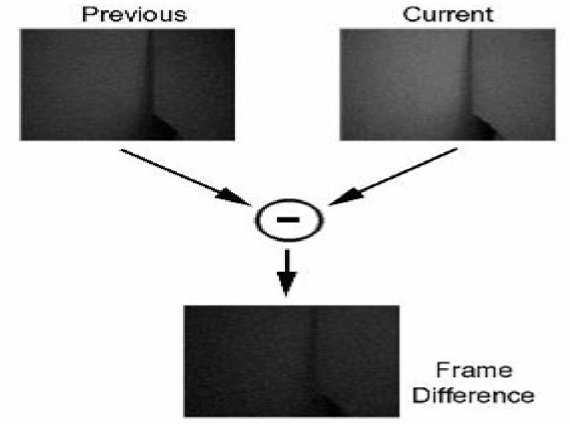

Fig 2: Motion Detection using simple SAD

Motion Estimation using SAD

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

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

Functional Description

The Video Surveillance system block diagram is shown. 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)

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,t1), f(x,y,t2) ……….. f(x,y,tn) and let f(x,y,t1) 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.

III. 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 block set 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.

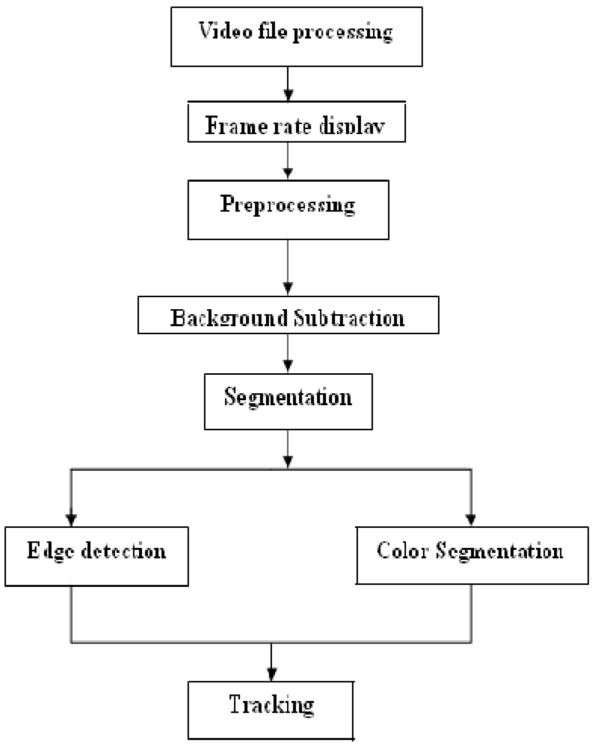

Fig 3: Object Tracking Subsystem

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.

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:

A. 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.

B. 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.

C. Background Subtraction

It 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.

IV. RESULTS

Fig 4: Test Result (Human/Object)

Fig 5: Test Result (Human/Object)

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Fig 6: Test Result (Non Human)

V. CONCLUSION & FUTURE WORK

The algorithm is implemented in Mat lab program. The size of the input video image is 352×288 pixels and the sample rate is 30 frames per second. In this project a methodology is developed to detect the motion in a live streaming video and once motion has been detected in the live stream, the software will activate a warning system and capture the live streaming video frames only in which motion is detected. So this reduces the total number of video frames to record. This saves the storage space required in the hard disk.

The following features can be added to the existing work. In this proposed work human or nonhuman recognition of the object in motion is done only for considering still images. In future this can be done for real time systems. System can be developed to recognize human activities. Also once the object is classified as human or nonhuman, exact identification of nonhuman (whether it is a cat, dog, bike etc) can be done and face recognition of human can be done.

REFERENCES

[1] Y.L. Tian and A. Hampapur, "Robust Salient Motion Detection with Complex Background for Real-time Video Surveillance," IEEE Computer Society Workshop on motion and Video Computing, Breckenridge, Colorado,January 5 and 6, 2005.

[2] C. Bahlmann, Y. Zhu, Y. Ramesh, M. Pellkofer, T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," IEEE Intelligent Vehicles Symposium, Proceedings,2005,pp. 255-260.

[3] A. Manzanera and J.C. Richefeu, "A new motion detection algorithm based on Σ -background estimation," Pattern Recognition Letters, vol. 28, n 3, Feb 1, 2007, pp. 320-328.

[4] J. Guo, D. Rajan and E.S. Chng, "Motion detection with adaptive background and dynamic thresholds," 2005 Fifth International Conference on Information, Communications and Signal Processing, 06-09 Dec. 2005, pp. 41-45.

[5] Y. Shan and R.S. Wang, "Improved algorithms for motion detection and tracking," Optical Engineering, vol. 45, n 6, June 2006.

[6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human. Detection", Proceedings of the Conference on Computer Vision and Pattern Recognition, San Diego, California, USA, pp. 886–893, 2005.

[7] Q. Zhu, S. Avidan, M-C Yeh, , K-W Cheng, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients", Proceedings of the IEEE Computer Society Conference on Computer vision and Pattern Recognition, ISSN: 1063-6919, Volume 2, pp. 1491-1498, June 2006.

[8] Nehme, M.A.; Khoury, W.; Yameen, B.; Al-Alaoui, M.A., "Real time color based motion detection and tracking", *Proc.* ISSPIT 2003, 3rd IEEE International Symposium on Signal Processing and Information Technology, *2003*, 14-17 Dec. 2003, pp. 696 – 700, 14-17 Dec. 2003.

[9] Nan Lu, Jihong Wang, Q.H. Wu and Li Yang, "An Improved Motion Detection Method for Real-Time Surveillance" IAENG International Journal of Computer Science, vol. 35 issue 1, pp119-128

[10] Asif Ansari1, Dr.T.C.Manjunath (Ph.D., IIT Bombay) 2, Dr. C.Ardil3 "Implementation of a Motion Detection System" International Journal of Computer Science Volume3.

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)