SSRG International Journal of Electronics and Communication Engineering
ISSN: 2348-8549/ https://doi.org/10.14445/23488549/IJECE-V9I5P103
Volume 9 Issue 5, 17-20, May 2022

Original Article

Smart Maneuvering of Security Camera

Dhatreyee Eluri[1], A. Raghu Ram[2]

[1] *Department of ECE, Jawaharlal Nehru Technological University, Hyderabad, Telangana.*
[2] *Department of EEE, Jawaharlal Nehru Technological University, Hyderabad, Telangana.*

Received: 18 March 2022 Revised: 15 May 2022 Accepted: 27 May 2022 Published: 30 May 2022

Abstract **-** *One of the ongoing demanding research problems in computer vision is visual surveillance in dynamic situations, particularly for humans and automobiles. It is a critical technology in the battle against terrorist attacks, crime, public health and safety, and effective traffic management. The endeavour entails the creation of an effective visual surveillance system for use in complex contexts. Detecting directional movement from a video is critical for target tracking, object categorization, activity recognition, and behaviour understanding in video surveillance. The first relevant phase of data is detecting object tracking in streaming video, and background subtraction is a typical method for foreground segmentation. Different backdrop subtraction approaches are simulated in this work to tackle the issues of lighting variance, background clutter, shadows, and concealment.*

Keywords **-** *Computer vision, Motion detection, Background subtraction.*

1. Introduction

Surveillance cameras are a common method of averting and investigating criminal activities. The power to recognize offenders by identifying them while examining crime footage in real-time, or the ability to stop the crime entirely by causing potential lawbreakers to give up trying or think twice before performing the crime. As a result, surveillance footage has become an important aspect of the investigation[1].

Intelligent surveillance systems could use independent image processing technology on a computer to interact with a camera linked via wired or wireless media. The camera's signals are transformed into digital signals before being forwarded to the host device[2]. Image processing software can communicate with other hardware, such as a microcontroller unit that controls the camera's movement[3]. In intelligent surveillance systems, image processing has several applications. It can notify users of situations such as video deletion, alarm systems, and break-ins [4]. A sequence of prerecorded IDs is used to identify these incidents on the show. This method integrates various camera inputs into a single large, distortion-free image[6]. The image subtraction technique is the most frequently used image processing technique, which evaluates two successive video frames to monitor an object's movement.

2. Objective

The project's major goal is to build a surveillance system that only records when motion is detected. It tries to limit the amount of data stored to preserve hard drive space while shortening the duration of video clips for easy inspection. The project's three main components are a camera, a computer, and a turnable camera. The input component of the software will be the camera. The computer acts as a storage media and a central algorithm processing unit. The computer delivers the necessary data to the turntable after processing the data. A microcontroller receives basic instructions, and a motor controls the camera image on this turntable.

2.1. Algorithm

The USB camera and motion sensor detect the moving object, and Image Processing processes it. The situation determines the commands of the software. It will save recorded video to a computer and modify the motor's angle. In this project, the video sequence frames are transformed to grayscale and filtered to eliminate frame noise. It takes the previous and current frames and subtracts them[7]. If the frame changes, it will simply continue processing and extracting information from the video frames before saving it as a video file on the computer.

The flow chart in the above diagram depicts the program's specific procedure and operations. The frames of a video series are encoded to grayscale and further filtered to eliminate the frame noise produced. It takes the current and previous frames and subtracts them, and then based on the threshold value obtained, movement will be decided. If there is any movement, then the frame will be recorded.

2.2. Implementation of Image processing methods:

2.2.1 Image Acquisition

The camera captured the image directly and fed it into the Digital Signal Processor during image capture. The camera completely determines the color bits. It is also where the recorded video is taken. For ease of processing, the image is transformed from color (16 or 32 bits) to grayscale (8 bits).

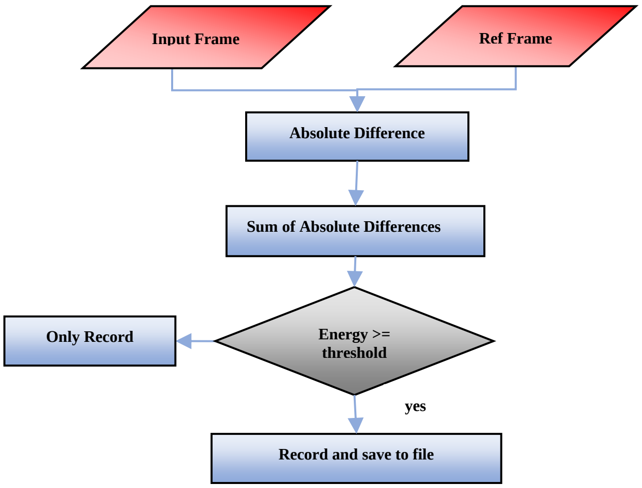

Fig. 1 Flowchart of Surveillance Camera designed

2.2.2. Pre-Processing

Once we have the frames from the video, we must use preprocessing techniques[8] to make them easier to analyze. The most common techniques include color enhancement, restoration, processing, compression, and noise removal. To avoid data loss, preprocessing must be done with caution.

2.2.3. Motion Detection

First, the objects are detected[9] in the video frames before beginning the object-tracking process. Because motion recognition algorithms are often the primary source of information and the cheapest in computing memory, most of the methods covered in this area are motion recognition algorithms[10]. As a result, we'll go through the frame difference algorithm, background subtraction strategies[11,12,13], and optical flow, as shown in fig 2 below.

2.2.4. Filters

The background features in the removed image may produce a false positive, resulting in issues such as recording video at the incorrect time and the camera facing the wrong direction. Filtering removes these unwanted noises[21]. Convoluting the image[15] with a low-pass filter, the kernel blurs it. It is used to lower noise levels. When this filter is used, it eliminates high-frequency information from the image, resulting in blurred edges.

2.2.5. Absolute Difference

The Absolute Difference block of the SAD processor calculates the absolute differences in the larger search region between the reference pixels of 4X4 block size and the matching current input pixel data of 4X4 block size. A 4X4 block requires 16 absolute difference units.

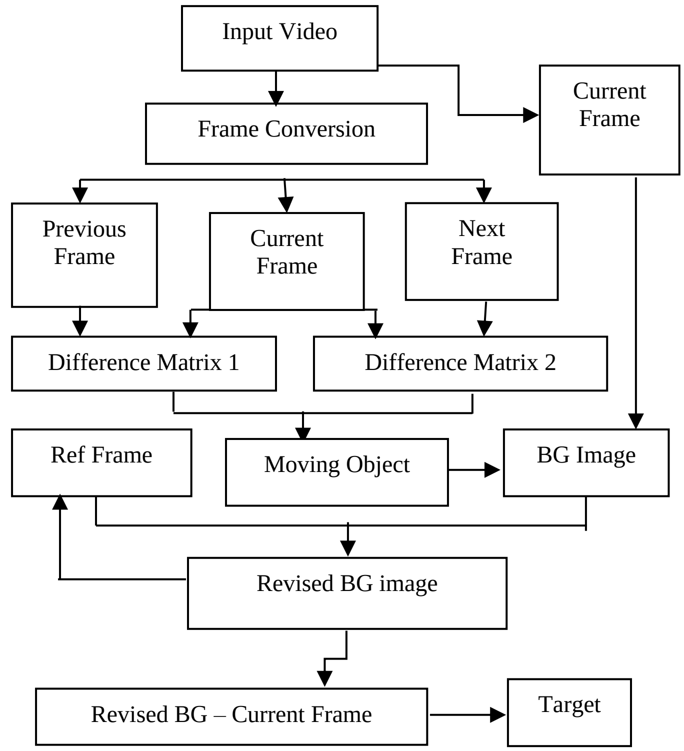

Fig. 2 Background subtraction method flowchart

2.2.6. Sum of Absolute Difference(SAD)

This block is used to generate a single SAD value. And the outcomes of each 4X4 block's absolute difference units are summed, and this method is iterated with the next incoming 4X4 block using the 4X4 block size's reference pixel.

2.2.7. Comparator

The minimal SAD value is determined by comparing the single SAD values of each 4X4 block size in the search region using a comparator. Compared to the other SAD blocks, the matching 4X4 block-sized current input pixel data of the smallest SAD value is deemed identical to the reference block. The reference block size determines the block size. The Sum of Absolute Difference algorithm employs a pixel-based approach[17] for finding the disparity.

Finding the optimum motion vector for the current frame, which is the displacement of the coordinate of the best identical block from the previous frame, is defined as motion estimation. The Sum of Absolute Differences metric is the most commonly used for calculating distortion because it adds up the absolute discrepancies between successive elements in the reference and candidate blocks. The higher processing complexity of block matching algorithms (BMA) could be a top priority in real-time coding applications. Different VLSI technologies are intended to minimize computational complexity by accelerating huge arithmetic calculations. On the other hand, the requirement for specialized hardware appears to contradict the adaptability demanded by current coding video systems. Using a programmable CPU core and programmable gate array devices (FPGAs) to conduct important tasks is a feasible solution to this problem.

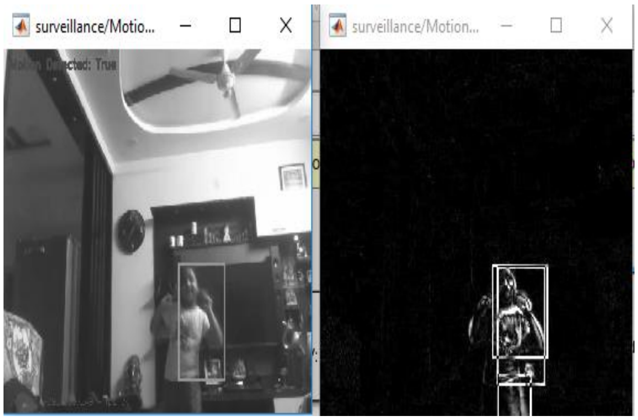

Fig. 3 Input Video frames & Processed Video Frames

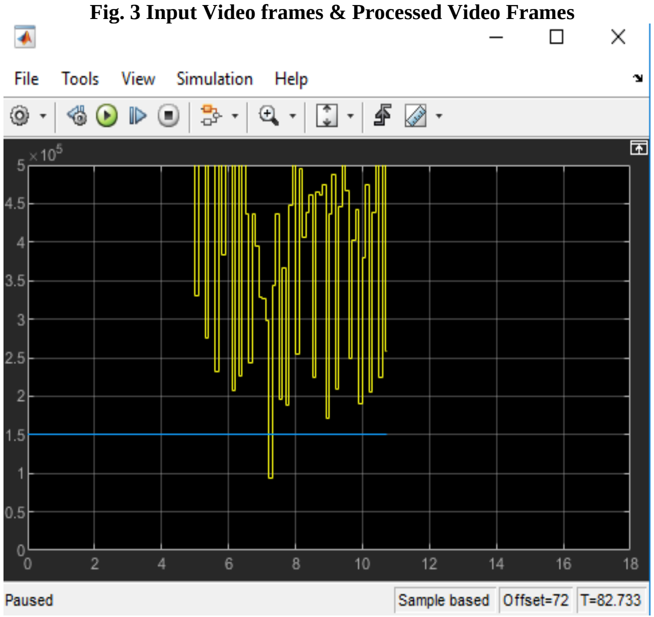

Fig. 4 Threshold Graph for the input video frames

The image frame will be captured from the camera, and then the image is processed using the Image Processing techniques to remove all the external noises. The above fig 3 shows the captured image frame and its processed image frame.

Above Fig 4 shows the Threshold Graph, which describes the movement of the image frame. The movement of the image will be recorded only if the threshold value exceeds the given minimum limit[20].

Calculations

The following equation is used to calculate the SAD (Sum of Absolute Differences) for Motion Energy estimation:

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

where, $I_{(k-1)} (i,j)$ = previous frame,

$I_k (i,j)$ = current frame.

3. Results and Discussion

The theoretical and real values are similarly based on the data acquired during testing. The person's height unalters the accuracy, and it's worth mentioning that detected pixels have bigger values than predicted ones since movement leads to adding pixels to the screen.

A minor delay exists between motion detection and recording, but it is nearly unnoticeable since the delay can only be identified using timers to monitor it through the block diagram. The delay values are extremely close to 40 milliseconds, which would be the time it takes a 29 fps camera to capture each frame. The delay is primarily determined by the camera being utilized rather than the application itself. Because all of the specific and general objectives have been met, the proponents can now conclude that the completed project is a surveillance system that efficiently uses computer memory for recording, can correctly determine the height of moving objects and has a short response time.

References

[1] M. Cross, Guardian, Explainer: CCTV, 2009. [Online]. Available: http://www.theguardian.com/commentisfree/libertycentral/2009/nov/06/explainer-cctv-surveillance-cameras.

[2] E. C. Hannah, "*Method and Apparatus for Processing Digital Video Camera Signals*," US Patent, pp. 192, 1996.

[3] J. A. Kalomiros, and J. Lygouras, "Design and Evaluation of a Hardware / Software FPGA-Based System for Fast Image Processing," *Microprocessors and Microsystems*, vol. 32, no. 2, pp. 95-106, 2007. *Crossref*, https://doi.org/10.1016/j.micpro.2007.09.001

[4] D. Tian, "*Utilizing a Smart Camera System for Immersive Telepresence*," US Patent Application 2014/0253667 A1, 2014.

[5] Berkeley Design Technology, Inc. Choosing a Processor: Benchmark and Beyond, World Wide Web, 2006. [Online]. Available: http://www.bdti.com/articles/20060301_TIDC_Choosing.pdf

[6] D. Chun, "*CCTV System Having Improved Detection Function and Detecting Method Suited for the System,*" US Patent 5 671 009, 1997.

[7] Alan J. Lipton, Hironobu Fujiyoshi, and Raju S, Patil, "Moving Target Classification and Tracking from Real-Time Video," *In Proceedings of the 4th IEEE Workshop on Applications of Computer Vision, WACV '98, IEEE Computer Society*, vol. 8, 1998. *Crossref*, https://doi.org/10.1109/ACV.1998.732851

[8] Forsyth D.A, and Ponce J, "*Computer Vision: A Modern Approach,*" Pearson Education, Upper Saddle River, NJ, 2003.

[9] Ying-Li Tian, and Arun Hampapur, "Robust Salient Motion Detection with Complex Background for Real-Time Video Surveillance*," 2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05)*, vol. 1, 2005. *Crossref,* https://doi.org/10.1109/ACVMOT.2005.106

[10] Shih-Chia Huang, "An Advanced Motion Detection Algorithm with Video Quality Analysis for Video Surveillance Systems," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 21, no. 1, pp. 1-14, 2011. *Crossref,* https://doi.org/10.1109/TCSVT.2010.2087812

[11] Ahmed Elgammal, David Harwood, and Larry Davis, "Non-Parametric Model for Background Subtraction," *European Conference on Computer Vision,* vol. 1843, pp. 751-767, 2000. *Crossref,* https://doi.org/10.1007/3-540-45053-X_48

[12] K Toyama et al., "Wallflower: Principles and Practice of Background Maintenance," *Proceedings of the Seventh IEEE International Conference on Computer Vision,* vol. 1, no. 3, pp. 255-261, 1999. *Crossref,* https://doi.org/10.1109/ICCV.1999.791228

[13] L. Maddalena, and 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. *Crossref,* https://doi.org/10.1109/TIP.2008.924285

[14] Geetha D et al., "Human and Animal Movement Detection in Agricultural Fields," *SSRG International Journal of Computer Science and Engineering,* vol. 6, no. 1, pp. 15-18, 2019. *Crossref,* https://doi.org/10.14445/23488387/IJCSE-V6I1P103

[15] Lina J. Karam, and David Rice, Image Convolution Concepts and Applications Online Tutorial, Arizona State University. [Online]. Available: http://www.eas.asu.edu/~karam/2dconvolution/

[16] Akshay Choudhari et al., "Camera Surveillance System Using Motion Detection and Tracking," *International Journal of Innovative Research in Advanced Engineering (IJIRAE),* KBTCOE, vol. 1, no. 4, p. 38, 2014.

[17] Nishu Singla, "Motion Detection Based on Frame Difference Method," *International Journal of Information and Computation Technology,* vol. 4, no. 15, pp. 1559-1565, 2014.

[18] Ponce J, and Forsyth D.A, "*Vision of the Computer: A Modern Approach*," Pearson Education, Upper Saddle River, NJ, 2003.

[19] Steven W. Smith, "*The Scientist and Engineer's Guide to Digital Signal Processing*," Second Edition, California Technical Publishing, 1999.

[20] Nobuyuki Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 9, no. 1, pp. 62-66, 1979. *Crossref,* https://doi.org/10.1109/TSMC.1979.4310076

[21] Jain R, Kasturi R, and Schunck G, "Machine Vision," *McGraw Hill,* 1995.