Moving Object Detection using Tracking, Background Subtraction and Identifying Outliers in Low Rank Video

Mr. Pankaj Ughade, M.Tech Scholar, Department of Computer Science and Engg, Rashtrasant Tukadoji Maharaj Nagpur University, Maharashtra

Mr. Vikash Kumar, M.Tech Scholar, Department of Computer Science and Engg, Rashtrasant Tukadoji Maharaj Nagpur University, Maharashtra

Abstract- Detection of moving objects in a video sequence is a difficult task and robust moving object detection in video frames for video surveillance applications is a challenging problem. Object detection is a fundamental step for automated video analysis in many vision applications. Object detection in a video is usually performed by object detectors or background subtraction techniques. Frequently, an object detector requires manual labeling, while background subtraction needs a training sequence. To automate the analysis, object detection without a separate training phase becomes a critical task. We done a survey of various techniques related to moving object detection and propose the optimization methods that can lead to improved object detection and the speed of formulating the low rank model for detected object. In this project proposes, the three modules for detecting moving object with fixed camera and detecting moving object with moving camera and detecting and removing outlier present in sequence of frames, so we consider the outlier may be any variation, distortion or noise in the sequence of frames. The project proposes the modules, work on first process video then segment video and robustly recognized the moving object in video sequence and removing the outlier with low rank model.

Index terms- Soft Impute method, Temporal Differencing, Moving object extraction, background subtraction, Object Detection, Markov Random Field.

1. INTRODUCTION

This project involved implementing moving object detection for static and dynamic background software. This chapter provides a discussion of the project objective to achieve. It also gives a high-level overview of the system, leaving design and implementation details for discussion in the respective chapters. It also provides a roadmap for the reader about the overall presentation and structure of the report. The primary goal of this project is to critically discuss the various techniques for detecting moving objects methods in static and dynamic background in video. A second goal is to present a technique for formulating low rank model for detected object.

Nowadays, it is seen that surveillance cameras are already prevalent in commercial establishments, with camera output being recorded to tapes that are either rewritten periodically or stored in video archives. To extract the maximum benefit from this recorded digital data, detect any moving object from the scene is needed without engaging any human eye to monitor things all the time. Real-time segmentation of moving regions in image sequences is a fundamental step in many vision systems [10]. A typical method is background subtraction. Many background models have been introduced to deal with different problems. One of the successful solutions to these problems is to use a multi-color background model per pixel proposed by Grimson et al [12]. However, the method suffers fromslow learning at the beginning, especially in busy environments. In addition, it cannot distinguish between moving shadows and moving objects. Image background and foreground are needed to be separated, processed and analyzed. The data found from it is then used further to detect object. In this project work robustly for accurately detecting and tracking moving objects have been developed and analyzed. The new method currently operates on video taken from a stationary camera. The traditional real time problems are taken under consideration including outlier while detecting moving object. An improved outlier detection method is coordinated to handle the issue.

1.1 Objective

Automated video analysis is important for many vision applications, such as surveillance, traffic monitoring, augmented reality, vehicle navigation, etc.There are three key steps for automated video analysis: object detection, object tracking, and behavior recognition[9]. As the first step, object detection aims tolocate and segment interesting objects in a video. Then, suchobjects can be tracked from frame to frame, and the trackscan be analyzed to recognize object behavior. Thus, objectdetection plays a critical role in practical applications [15]. The main research objectives to design the project are

- To improve the accuracy detecting the object in video and cut down the cost of computations using the process of optimization.
- To handle static background and dynamic background while process the video.
- To detecting and removing the outliers present in sequence of frames.

1.2 Object detection

Detecting regions that correspond to moving objects invideo sequence plays a very important role in many computer vision applications. In simplest form Object detection from video sequence is the process of detecting the moving objects in frame sequence using digital image processing techniques. Moving object detection is the basis of moving object identification and tracking. Moving Object detection in consequent images is nothing but the detection of the moving object in the scene. In video surveillance, motion detection refers to the capability of the surveillance system to detect motion and captures the events. Moving object detection is usually a software-based monitor in algorithm which will signal the surveillance camera to begin capturing the event when it detects object. A side from the intrinsic usefulness of being able to segment video streams into moving and background components, detecting moving blobs provides a focus of attention for recognition, classification, and activity analysis, making these later processes more efficient since only "moving" pixels need be considered. There are three conventional approaches to moving object detection [9] temporal differencing, background subtraction and optical flow. Temporal differencing is veryadaptive to dynamic environments, but generally does a poor job of extracting all relevant featurepixels. Background subtraction provides the most complete feature data, but is extremely sensitiveto dynamic scene changes due to lighting and extraneous events. Optical flow can be used to detectindependently moving objects in the presence of camera motion; however, most optical flowcomputation methods are computationally complex, and cannot be applied to full-frame video streams in real-time without specialized hardware [9].

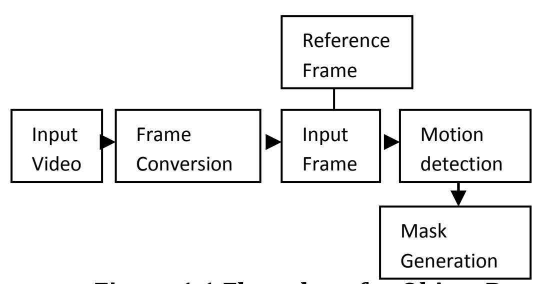

Figure 1.1 Flow chart for Object Detection

1.3 Moving Object in real time environment: Problems

Moving object detection is fundamental in many autonomous video surveillance strategies. In real time environment where scene is not undercontrol situation is much worse and noisy [1]. Recent study has produced several background modelingtechniques, based on image differencing, that exhibit real-time performance and high accuracy forcertain classes of scene. The aim of this project is to assess the performance of some of these background modeling techniques (namely, Soft Impute Method, Markov Random Field, Graph cut segmentation and detection and removal of outlier Algorithm) using video sequences of outdoor scenes where the weather introduces unpredictable variations in both lighting and backgroundmovement which is occurred because of outliers (any variation in illumination, distortion or noise in sequence of frames in video.). The results are analyzed and reported, with the aim of identifying suitable directions forenhancing the robustness of motion detection techniques for outdoor video surveillance systems [2]. Motion in indoor and other situations are considered and analyzed as well.

[3] Offers a look at current state of research in the field of moving object detection in videos. The chapter discusses various object detection techniques in sequences of frames, such as optical flow method, consecutive frame subtraction and background subtraction. The architecture, block diagram, and methodology from our project modules are explained in chapter. In block 4 and 5 explain which software tools and platforms used in developing project and design (Data flow diagrams of modules of project) and implementation techniques. [6] Presents the result of every module separately and also discusses the conclusion of problem statement.

2. THE PROPOSED METHODS

Following on from the previous chapter's introduction to the project methodologies and project tools and platform, the goal of this chapter is to mention details of the actual implementation of the code. It details the steps taken for each project module and working of each module that make up the system. Brief analysis of the project's code is made and discussed, along with the technologies and algorithm used in implementing the project. Moving Object detection is the basic step for further analysis of video. Every tracking method requires an object detection mechanism either in every frame or when the object first appears from stationary background object. When working with video data, it can be helpful to select a representative frame from video and the methods can be applied to the processing of all the frames in the video. The method computes the estimated foreground and background model of frame specified by rank.

In this section, we focus on the problem of detecting moving object in low rank representation. We first consider the case without camera motion, with camera motion and outliers' present in video. So three key steps we will consider design the project. Modules are as follows:

1. Moving Object Detection in Static Camera
2. Moving Object Detection in Moving Camera
3. Detect and Remove the Outliers

All the three modules in the project should go through the common module [8] i.e. preprocessing module that module contains the entire basic step required to compute data matrix and transformation matrix. Once we will get both input matrix then will work on the algorithm. Here Figure 2.1 shows the basic data flow for moving object detection system. Here the flow chart that shows first call

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 08 | Aug-2016 www.irjet.net p-ISSN: 2395-0072

preprocessing then call soft-impute method then segment the object.

The input to the algorithm is a sequence of video frames which convert RGB to gray-level format. The algorithm produces a binary mask for each video frame. The pixels in the binary mask that belong to the background are assigned 0 values while the other pixels are assigned to be 1.

The preprocessing module performs basic steps to process the video frames for detecting object from video. Figure 2.2 shows the preprocessing steps of video sequence.

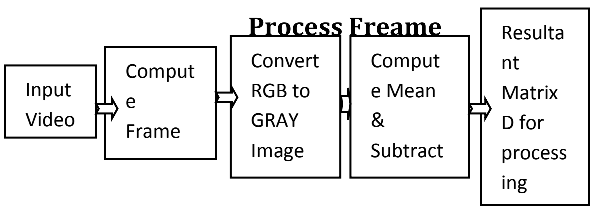

Figure 2.1 Preprocessing steps of video sequence

2.1 Case 1: Moving Object Detection in static camera

This section describes moving object detection in static camera where the objects are in moving state and the background are static. Background refers to a static scene and foreground refers to the moving objects. Objective is to estimate the foreground support as well as underlying background images. Steps are as follows:

- Preprocessing [Moving Object And Static Background]
- Transformation Matrix
- Segment the interesting object from video sequence.

The following figure 2.2 shows the detecting moving object in static background.

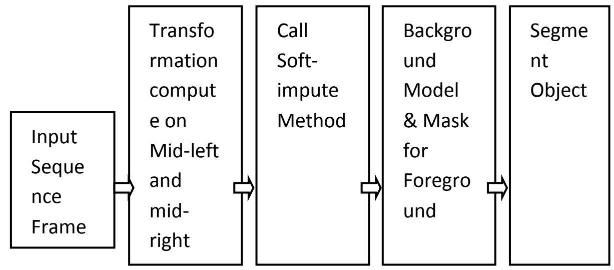

Figure 2.2 Detecting Moving Object in Static Background

As the steps given for detecting object in static background ,first it preprocess all the frames from frame first to the last frame, as each frame is process it compute the transform matrix for each frame i.e. what variation are arise in the object position. Here the figure shows the preprocessing for each frame.

When work on static then the compute the transform matrix by compute differences of two frames, start from mid-1 to left and then mid+1 to right, because the most of the variation arise at middle part of video. Figure 2.3 shows the transform matrix computation for the static camera module.

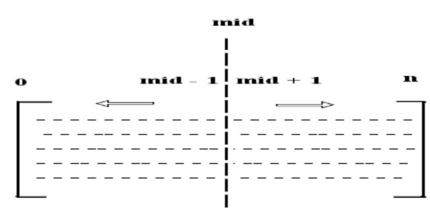

Figure 2.3 Transformation for static camera

Once compute the transform and data matrix then call Soft-Impute method. The outcome of Soft-Impute is the smoothen background and mask for foreground. To formulate the background model and foreground mask, the SOFT-IMPUTE [24] method is used which produces a sequence of solutions for which the criterion decreases to the optimal solution with every iteration and the successive iterates get closer to the optimal set of solutions of the problem. SOFT-IMPUTE decreases the value of the objective function towards its minimum, and at the same time gets closer to the set of optimal solutions of the problem In many applications measured data can be represented in a matrix *Xm×n*, for which only a relatively small number of entries are observed. The problem is to "complete" the matrix based on the observed entries, and has been dubbed the matrix completion problem.

SOFT-IMPUTE iteratively replace the missing elements with those obtained from a soft-threshold SVD. SOFT-IMPUTE algorithm, which makes use of the following lemma [11]:

Lemma 1. Given a matrix Z, the solution to the optimization problem

$$\underset{Z}{\text{minimize}} \quad \frac{1}{2}\|W-Z\|_F^2+\lambda\|Z\|_*$$

is given by $\hat{Z} = \mathbf{S}_\lambda(W)$ *where*

$$\mathbf{S}_\lambda(W) \equiv UD_\lambda V' \quad \text{with} \quad D_\lambda = \operatorname{diag}\left[(d_1-\lambda)_+,\ldots,(d_r-\lambda)_+\right],$$

UDV' *is the SVD of* W, $D = \operatorname{diag}[d_1,\ldots,d_r]$, *and* $t_+ = \max(t,0)$.

Using Lemma 1, the optimal solution to can be obtained by iteratively using:

$\hat{B} \leftarrow \Theta_\alpha(\mathcal{P}_{\hat{S}^\perp}(D)+\mathcal{P}_{\hat{S}}(\hat{B}))$ with arbitrarily initialized $\hat{B}$.

The foreground is defined as any object that moves differently from the background. Foreground motion gives intensity changes that cannot be fitted into the low-rank model of background. Thus, they can be detected as outliers in the low-rank representation. Generally, we have a prior that foreground objects should be contiguous pieces with relatively small size.

2.1.2 Algorithm: Background model and foreground mask estimation using soft impute method

Soft Impute: iterative soft threshold SVD to impute the missing values

Input: D=[$I_1, I_2, \ldots\ldots I_n$] € IR^{m*n}

Initialization:

'X': is incomplete matrix

'maxRank': is the desired rank in the constraint

'Omega': is the mask with value 1 for data and 0 for missing part

Steps:

```
ifisEmpty(Z)
        z=x;
end
ifisEmpty(Omega)
    Omega=true(size(x))
end
ifisEmpty(maxRank)
        maxRank=-1;
end
Repeat
while(1)
    -c=x*omega+z*(1-omega)
    -apply the SVD(single value Decomposition)
    -d=diag(D)
    -index=find(d>alpha)
    -'z' recompute based on index
    -k=length(index)
Termination condition
    Repeat
    -if (k<maxRank&& omega >0.0001)
        alpha=alpha+eta;
    else
```

```
                break;
        end
```

Output: smooth Background Model and masks for foreground model.

After executing the Soft-Impute, will get the Background model and foreground mask then segment the interesting object from video sequence.

As the steps given for detecting object in static background, first it preprocess all the frames from frame first to the last frame, as each frame is process it compute the transform matrix for each frame i.e. what variation are arise in the object position. Here the figure shows the preprocessing for each frame [12].

Figure 2.4 Preprocessing for each frame in video sequence

As shown in figure 2.4 preprocessing for each frame in video then will get matrix transformation i.e. tau matrix. Then Soft-Impute method is called for getting the low rank module and mask for foreground. Figure 2.5 show the smoothen background model and foreground mask for the input video sequence.

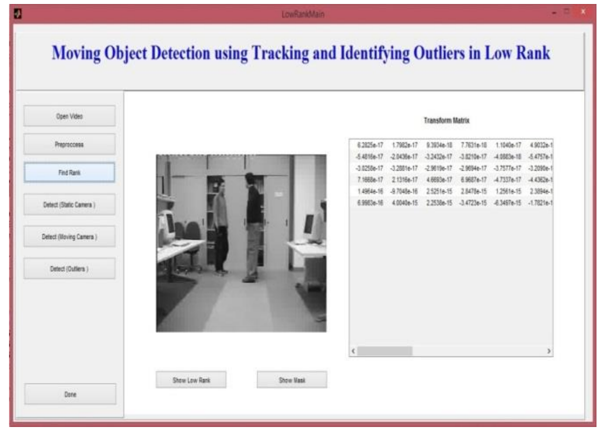

Figure 2.5 Foreground Mask for the moving Object in Video

As figure 2.5 shows the foreground mask for the object in video sequence then we will segment the interesting object for the video. So figure 2.6 shows the four panels in which, the first panel show the original video sequence and the second panel shows the smoothen background model for video and third panel shows the foreground i.e. interesting object and last panel shows the segmented object form background.

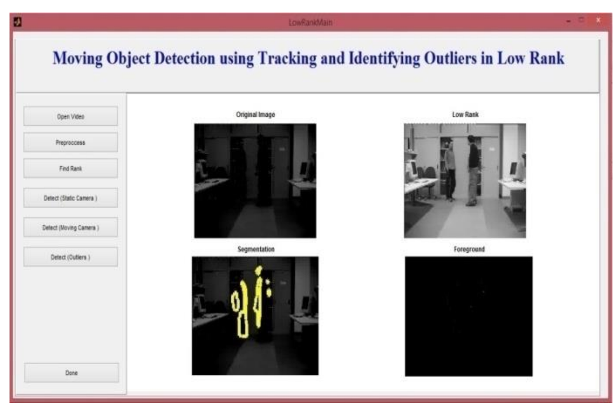

Figure 2.6 Four panels for segment the interesting object form static background

2.1.2 Case 2: Moving Object Detection in Moving Camera

The above section is based on the assumption that the videos are captured by static cameras and background is static and the foreground is moving. In this section, we propose the method which handles the both background and foreground are moving which is caused by moving cameras [13].

The proposed method uses image registration for detection moving object in motion camera. The registration is a process which makes the pixel in two images precisely coincide to the same points in the video. Once registered the image can be combined or fused in a way that improve detection of foreground in motion camera [14].

In this method, we use dataset having object is moving in the video with motion background. So it follows the step for detecting moving object in moving background.

Steps:

- Preprocessing [Moving Object And Moving Background]
- Transformation Matrix [Indentify different transformation]
- Segment the interesting object from video sequence.

The following figure 5.5 shows the detecting moving object in moving background [15].

Figure 2.7 Detecting Moving Object in Moving Background

As discussed in static module, the transformation matrix computation is different for the moving camera. The transformation matrix would determine which type of transformation present in sequence of frame i.e. similarity transform , affine transform, rigid transform or the projective transform , depends on which transformation used based on that the number of parameter is required [16].

Transformations are used:

• Position objects in a scene (modeling)

• Change the shape of objects

• Create multiple copies of objects

• Projection for virtual cameras

As project module work on the position objects in a scene (modeling), so to find position of object in video frame sequence.The image registration process which perform registration of the object present in video sequence. Determine the transformation between corresponding points. First, assume that all pairs of corresponding points are related by the same transformation and then compute parameters of transformation given corresponding points. Maps points (x, y) in one coordinate system to points (x', y') in another coordinate system [16].

As discussed in chapter above that, the transformation matrix computation is different for the moving camera. The transformation matrix would determine which type of transformation present in sequence of frame i.e. similarity transform , affine transform , rigid transform or the projective transform , depends on which transformation used based on that the number of parameter is required. Figure 2.8 shows the four panel for detecting object form video sequence , the first panel show the original video sequence and the second panel shows the smoothen background model for video and third panel shows the foreground i.e. interesting object based on different transformation and last panel shows the segmented object form background [17].

Figure 2.8 Four panels for segment the interesting object form moving background

2.1.3 Case 3: Detecting and Remove outliers

This section discussed the case where the video sequence may contains undesirable artifacts due to occlusion (e.g. a image noise, any variation from underlying data generation method, distortion present in video sequence, blurring present in video sequence). The previous method did not address the problem based on outliers. So we develop the module detecting and removing outliers present in sequence of frame. Here figure 2.8 shows the steps detection and removal of outlier.

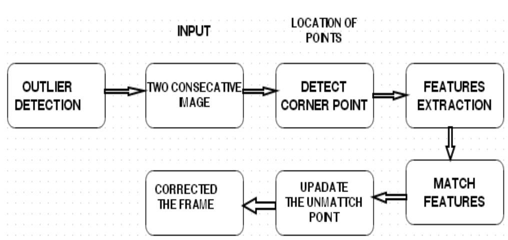

Figure 2.9 Steps for outlier's detection and remove

Function Used in outlier module:

i. Corner metric matrix and corner detector

The Corner Detector object finds corners in a grayscale image. It returns corner locations as a matrix of [x y] coordinates. The object finds corners in an image using the Harris corner detection (by Harris & Stephens), minimum eigen value (by Shi &Tomasi), or local intensity comparison (Features from Accelerated Segment Test, FAST by Rosten& Drummond) method. Finds corners in an image.

ii. extractFeatures

The extractFeatures function extracts feature vectors, also known as descriptors, from a binary or intensity image. The function derives the descriptors from pixels surrounding an interest point. These pixels describe and match features specified by a single-point location. The function extracts feature vectors from an input intensity or binary image

iii. matchFeatures

matchFeatures(FEATURES1,FEATURES2) also returns the metric values that correspond to the associated features indexed by INDEX_PAIRS in a *P*-by-1 matrix MATCH_METRIC.

As the video may contain any distortion then the proposed module should detect a remove the undesirable artifacts, for that it call the Detect outlier method, this method work on the corner detection technique. Once will detect the corner points then the method should indentifythe which image points are inliers and which are the outliers. Figure 2.8 shows the first two frames of video sequence where the hand movement are different in both frames , this works on the corner point detection approach and find outliers point .those outlier remove and then segment the interesting object form video [18].

This section provides results obtained in case where the video sequence may contains undesirable artifacts due to occlusion (e.g. a image noise, any variation from underlying data generation method, distortion present in video sequence, blurring present in video sequence). The previous method did not address the problem based on outliers. So we develop the module detecting and removing outliers present in sequence of frame [19]. Here figure 2.10 shows the video frame having the distortion present.

Figure 2.10 Distortion present in video Sequence

As the video may contain any distortion then the proposed module will detect an remove the undesirable artifacts, for that it call the Detect outlier method, this method work on the corner detection technique. Once will detect the corner points then the method should indentify the image points are inliers and which are the outliers. Figure 2.11 shows the first two frames of video sequence where the hand movement are different in both frames , this works on the corner point detection approach and find outliers point .those outlier remove and then segment the interesting object form video.

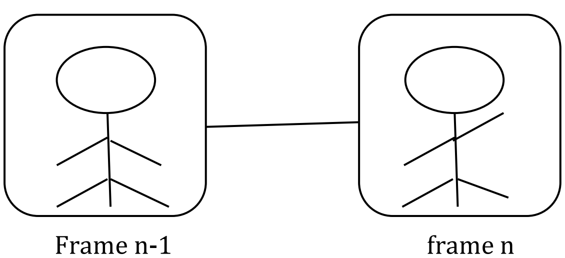

Figure 2.11 First two frames from video sequence

The outlier's detection module work on the corner point detection where it uses vision package, based on it first find the location of corner point (based on local intensity comparison). Figure 2.11 shows the corner points in both of the images input to the outlier module. So figure 2.12 first show the corner in 'A' image and corners in 'B' image.

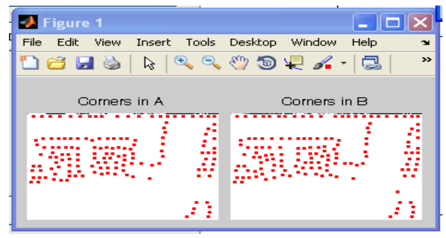

Figure 2.12 Corner Detection Detect the corner point

Once we get the corner points then, used the extract feature method. This method returns the valid point between two images. Then call math features it will match the points those point are not match shows by yellow line between two corner points. Figure 2.12 shows the unmatched point between two images.

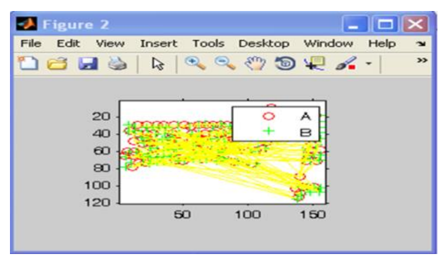

Figure 2.13 Corner Detection draw unmatched (yellow color) the corner point

Once detected and remove the outlier from video sequence then segment the interesting object from the video. Figure 2.13 shows the processing of two frames in first show the corner point, second shows the unmatched points and last show the corrected frame [17].

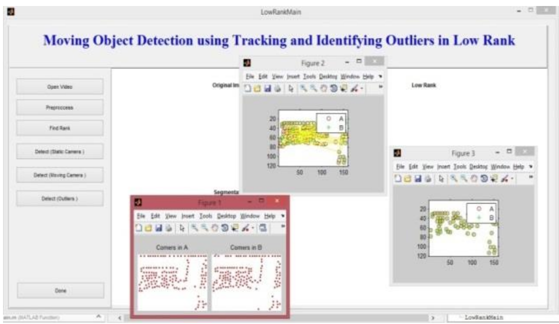

Figure 2.14 processing of two frames based on corner points

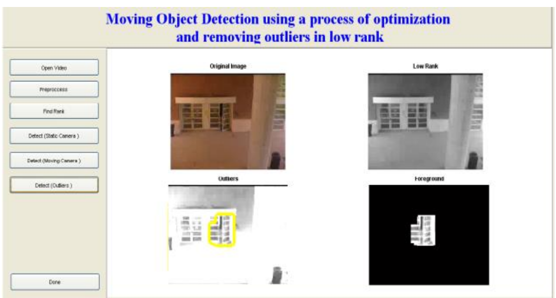

Figure 2.15 Four panels for segment the interesting object in outlier Detection

3. Conclusion

Here in this thesis, we have presented the methods for moving object detection and have carried out the work on object detection accurately. We had implemented the project to detect the moving object with static and moving camera and detect and remove the outlier present in the sequence of frames. The outlier may be any distortion, noise or blurred present in video.

The method define in the project to detect object may misclassify unmoved objects or large texture less regions as background since they are prone to entering the low-rank model. To address these problems, incorporating additional models such as object appearance or shape

Figure 2.14 Figure 2.15 and shows the four panel for detecting object form video sequence, the first panel show the original video sequence and the second panel shows the smoothen background model for video and third panel shows the foreground i.e. interesting object based on different transformation and last panel shows the segmented object form background [18].

prior to improve the power of project module can be further explored in future. Currently, project module works in a batch mode. Thus, it is not suitable for real-time object detection. In the future, we plan to develop the online version of module that can work incrementally, e.g., the low-rank model extracted from beginning frames may be updated online when new frames arrive. This project module has hopefully presented to the user with a best understanding possible of the moving object detection system developed.

4. References

[1] N. Paragios, and R. Deriche.. Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. Patt. Analy. Mach. Intell. 22, 3, 266–280, 2000.

[2] A Survey on Moving Object Detection and Tracking in Video Surveillance System Kinjal A Joshi, Darshak G. Thakore International Journal of Soft Computing and

Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012.

[3] C. Stauffer and W. Grimson. Adaptive background mixture models for realtime tracking. In Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, page 246252, 1999.

[4] Sumer Jabri, ZoranDuric, Harry Wechsler, Azriel Rosenfeld, "Detection and Location of People in Video Images Using Adaptive Fusion of Color and Edge Information," In Proc. 15th Int'l Conf. on Pattern Reg, 2000,vol. 4,pp. 627 – 630.

[5] James W. Davis, Stephanie R. Taylor, "Analysis and Recognition of Walking Movements," In Proc.16th Int'l Conf. on pattern Recognition, 2002, vol.1, pp. 315 – 318.

[6] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Who? When? Where? What? A real time system for detecting and tracking people".In Proc. Int. Conf. Auto. Face and Gesture Recog., 1998, pages 222–227.

[7] A Unified Approach to Moving Object Detection in 2D and 3D Scenes Michal Irani and P. Anandan IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 20, NO. 6, JUNE 1998 577.

[8] Shireen Y. Elhabian, Khaled M. El-Sayed and Sumaya H. Ahmed," Moving Object Detection in Spatial Domain using Background Removal Techniques - State-of-Art", Recent Patents on Computer Science, 2008.

[9]XiaoweiZhou,Can Yang, and Weichuan Yu," Moving Object Detection by Detecting Contiguous Outliers in the Low-Rank Representation", IEEE TRANSACTIONS ON PATTERN ANLYSIS AND MACHINE INTELLIGENCE, IEEE, March 2013.

[10] Imamura.K, Kubo.N, Hashimoto.H,"Automatic moving object extraction using x-means clustering Picture Coding Symposium (PCS),pp246 - 249 , Dec 2010.

[11] A. G. Bors and I. Pitas, "Optical Flow Estimation and Moving Object Segmentation Based on Median Radial Basis Function Network," *IEEE Trans. Image Process.*, vol. 7, no. 5, pp. 693-702, May 1998.

[12] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving Target Classification and Tracking from Real-time Video," Oct. 1998, *in Proc. IEEE Workshop Applications of Computer Vision*, pp. 8-14.

[13] S. S. Ghidary, Y. Nakata, T. Takamori, and M. Hattori, "Human Detectionand Localization at Indoor Environment by Home Robot," *in Proc. Int.Conf.on Syst. Man Cybern.*, Oct. 2000, vol. 2, pp. 1360-1365.

[14] A. M. McIvor. "Background subtraction techniques", In Proc. of Image and Vision Computing, Auckland, New Zealand, 2000.

[15] Sumer Jabri, ZoranDuric, Harry Wechsler, Azriel Rosenfeld, "Detection and Location of People in Video Images Using Adaptive Fusion of Color and Edge Information," In Proc. 15th Int'l Conf. on Pattern Reg, 2000,vol. 4,pp. 627 – 630.

[16] James W. Davis, Stephanie R. Taylor, "Analysis and Recognition of Walking Movements," In Proc.16th Int'l Conf. on pattern Recognition, 2002, vol.1, pp. 315 – 318.

[17] I. Haritaoglu, D. Harwood, and L. Davis, "W4: Who? When? Where? What? A real time system for detecting and tracking people".In Proc. Int. Conf. Auto.Face and Gesture Recog., 1998, pages 222– 227.

[18]Yoginee B. Bramhe(Pethe), P.S. Kulkarni, "An Implementation of Moving Object Detection,Tracking and Counting Objects for Traffic Surveillance System," Int'l Conf. on Computational Intelligence and Comm. Networks (CICN), 2011, pp. 143 – 148.

[19] C. Stauffer and W. Grimson. Adaptive background mixture models for realtime tracking. In Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, page 246252, 1999.

[20] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower:Principles and Practice of Background Maintenance," Proc. IEEE Int'l Conf. Computer Vision, 1999.

[21] T. Moeslund, A. Hilton, and V. Kruger, "A Survey of Advances in Vision-Based Human Motion Capture and Analysis," Computer Vision and Image Understanding, vol. 104, nos. 2/3,pp. 90-126, 2006.

[22] T. Sato. S. Kaneko. Y. Niwa. Y. Kazuhiko. Robust Object Detection by Radial Reach Filter. In IECE Trans, J86-DII, 616-624, 2003.

[23] T. Tanaka. A. Shimada. R. Taniguchi. T. Yamashita. D. Arita. Adaptive Background Modeling Considering Spatio-temporal Features for Object Segmentation. In Proc. MIRU, 127-134, 2009

[24] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral Regularization Algorithms for Learning Large Incomplete atrices," J. Machine Learning Research, vol. 11, pp. 2287-2322, 2010.