Use of Multiple Thresholding Techniques for Moving Object Detection and Tracking

S.Vijayalakshmi
Research Scholar
Department of Computer Science
VHNSN College
Virudhunagar, Tamilnadu, India

D.Christopher Durairaj, Ph.D
Associate Professor
Department of Computer science
VHNSN College
Virudhunagar, Tamilnadu, India

ABSTRACT

The present work proposes many threshold techniques for moving object detection and tracking system. It applies more than one threshold techniques during segmentation phase of the work. Object detection is done by background subtraction with Alpha method and object tracking is carried out by feature point tracking approach. It is observed that Otsu threshold method seems to have produced a perfect extraction and yielded good result in moving object tracking. The results of applying multiple thresholds are reported in this paper.

General term:

Image processing, object recognition

Keywords:

Object detection, tracking, threshold methods

1. INTRODUCTION

Now a day moving object detection and tracking has become the most important task for real time applications such as identifying an object of interest in a traffic flow, measuring speed of the vehicle, providing security in departmental stores etc. The color, shape, speed and directions of the moving object can be recognized in a detection and tracking system.

Object detection is a technique for detecting foreground pixels in an image. It involves background subtraction and extraction of object from an image. Various object detection methods such as background subtraction, temporal difference, and statistical methods, Eigen background subtraction have been used recently [1]. A performance comparison of moving object detection techniques is carried out in video surveillance system [2]. A single object is detected based on segmentation using threshold, background subtraction model and background estimation techniques [2]. In the present work, background subtraction technique followed by morphological operations is applied on a video.

Object tracking technique tracks the moving object from first frame up to last frame of the video. Object tracking is performed by using different types of techniques, such as blob matching tracking algorithm [3], Kalman filter [4], robust algorithm [5], and iterative method [6]. A new algorithm is implemented by using feature point tracking approach based on matching process [6]. Object tracking is done in a monocular image sequence by feature point tracking approach based on iterative method and objects are extracted [7].

Background pixels are extracted and subtracted from an image or a video is known as background subtraction. It is performed based on several methods that are adaptive background mixture model [8], Bayes classification of background [9], state of art method and statistical background subtraction [10]. An evaluation of background subtraction techniques for video surveillance has been proposed. Here, the background is extracted from video frames, then subtracted from every frame and compared with the background threshold [11]. If the basic threshold value is greater than background threshold, it is assumed that the pixel is treated as foreground otherwise it is background. This approach is followed in the present work.

Morphological operations are used for the analysis of binary images. The operations like erode and dilate are applied on background subtracted images. Erode is used for removing noises in an image sequence and dilate for enlarging the boundaries of foreground regions in an image sequence [12].

The details of object detection object extraction and object tracking techniques are given in section 2 of this paper. The system design and implementations are provided in section 3. The results of the system are shown in section 4. The present work is concluded finally.

2. METHODOLOGY

There are three different phases performed in the present work, namely object detection phase, object extraction phase and finally object tracking phase. The first phase consists of background subtraction followed by morphological operations. The theory of background subtraction is briefly given in section 2.1. A brief introduction of morphological operation is given in section 2.2. The description of object extraction using different threshold constitutes the phase 2 of the present work and is given in section 2.3. Object tracking is performed as third phase of the work and provided in section 2.4.

2.1 Background subtraction

Background subtraction is a one of the technique of image processing. In general, background pixels are detected and subtracted. The foreground pixels are detected and extracted based on thresholding. A statistical background modeling has been proposed for foreground detection. This method identifies the main challenges of background subtraction in the field of video surveillance [10].

In the present work, background subtraction is performed by background subtraction with Alpha method. The foreground and background pixels are first identified. It is given in following Eqn,

$$\left|I_{t(x,y)-}B_{t(x,y)}\right| > T$$

where, T is a pre-defined threshold. The background image B_t is updated by using following Eqn,

$$B_{t+1=}\alpha I_t + (1-\alpha)B_t\alpha$$

where, α is an adaptation coefficient [1].

2.2 Morphological operation

An image detection technique is proposed based on morphological edge detection and background differencing for real-time traffic analysis [13]. In the present work, morphological dilate and erode is applied on background subtracted image. Erode operation is used for removing noises in an image and dilate operation is used for enlarging the boundaries of foreground regions in an image.

2.3 Threshold Methods

Object extraction involves extracting the foreground object from an image sequences. Among the various techniques, threshold based techniques are simple and straight forward to apply. The three types of threshold techniques, namely Maximum Entropy Threshold, Mixture Modeling Threshold, and Otsu Method are applied in present work. The object extraction is necessary to detect the foreground object in an image.

2.3.1 Maximum entropy threshold

Maximum Entropy (ME) method is mainly used for object extraction and object recognition [14]. This method is used for extract the object from an image sequence by calculating gray level values for each pixel in an image sequence.

Using ME for automatic image annotation, the ME-based image segmentation approach is implemented to segment a gray-scale face image [15]. This approach uses Maximum Entropy Thresholding (MET) value of 2D image. MET operations are done by the following Eqn,

$$h_{ij=P_{ij}=\frac{f_{ij}}{N},} \qquad 0\le i,j\le L-1$$

where i is the gray level value of a pixel; j is the average gray level value of a pixel's neighborhood; f_{ij} is the number of the pixels, of which the gray level values are i and the average gray level values of their neighborhoods are j; ME-based method yielded very good threshold value and segmented the given image very well.

2.3.2 Mixture modeling threshold

Maximum entropy threshold is one of the thresholding methods. This method is used to extract the object from an image sequence by calculating mean and standard deviation for each pixel in an image sequence.

In adaptive background mixture models for real-time tracking [9], each pixel is classified based on Gaussian distribution. This thresholding process is done by following Eqn,

$$h_{model(g)} = n_B e^{-(g-\mu_B)2/2\sigma_B^2} + n_O e^{-(g-\mu_O)2/2\sigma_O^2}$$

- ❖ μ_B and σ_B is mean and standard deviation of background pixels
- ❖ μ_O and σ_O is mean and standard deviation of foreground pixels

Mixture modeling threshold technique assumes two classes, one for the foreground and another one for the background. The mean and standard deviation of these two classes are calculated and added. The result is assigned as mixture modeling threshold value.

2.3.3 Otsu threshold

Otsu's method is used to convert gray level image to a binary image. The two clusters are obtained by Otsu method based on threshold value and the statistical measures are optimized [16]. The automatic thresholding of gray-level images via two-dimensional. This method provides good segmentation of the object in an image.

In Otsu's method, background pixels belong to one class and foreground pixels belong to another class. The weight and variance of the two classes are calculated and added. This resultant value constitutes the Otsu threshold value. It is shown in following Eqn,

$$\sigma_\omega^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

$\omega_1(t)$ is weights of background pixels and $\omega_2(t)$ is weight of foreground pixels. $\sigma_1^2(t)$, $\sigma_2^2(t)$ are the inter class variance of the two classes. Otsu shows that minimizing the intra-class variance is the same as maximizing inter-class variance. It is shown in following Eqn,

$$\sigma_b^2(t) = \sigma^2 - \sigma_\omega^2(t) = \omega_1(t)\omega_2(t)[\mu_1\,(t) - \mu_2\,(t)]$$

where ω_i are the class probabilities and μ_i is the class mean. The class probability $\omega_1(t)$ is computed from t:

$$\omega_1(t) = \sum_0^t p(i)$$

while, the class mean $\mu_1\,(t)$ is:

$$\mu_1^t = \frac{\left|\sum_1^t p(i)x(i)\right|}{\omega_1}$$

2.4 Object tracking

Object tracking monitors the changes of objects in a video sequence, including its presence, position, size, shape, etc. K. Sethi and R. Jain have proposed finding trajectories of feature points in an image sequence and identifying the same point in more than one image [7]. In the present work, object tracking process is performed by using feature point tracking approach.

In any video frame, some pixels are visible and some other pixels are not visible. Thus an image is having two types of pixels, foreground pixels and background pixels. The foreground pixels are considered from the object of the image and background pixels are taken from the background of the image.

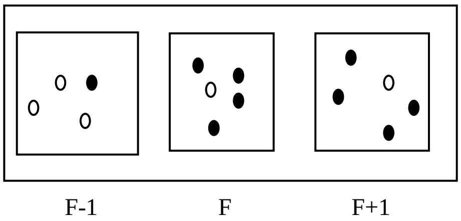


Fig 1: Frames of video

(See Figure 1) shows three frames (F-1, F, F+1) of the video. They represent previous frame, current frame and next frame. The frames are having both visible pixels and invisible pixels. Visible foreground pixels in current frame are compared with visible foreground pixels of previous frame. Single foreground visible pixel is present in current frame and previous frame means that it is easy to perform matching process and easy to identify the moving pixel. However, if multiple visible pixels present in current frame and previous frame, the sorting process based on intensity value of the pixels is performed.

When high intensity value of pixel is detected, those detected pixels are compared with pixels of previous frame. The moving pixel is thus identified. The moving pixel position is measured in current frame (q) and previous frame (p).

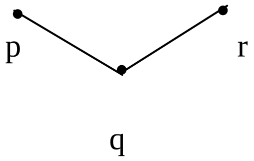


Figure 2: Connected Points (p, q, r: pixel positions in previous, current, and next frames)

The three points at p, q and r are connected by a line (see Figure 2). There is some distance value between two points. After extracting these feature points from the images, the feature points representing the same physical object point are identified in the whole image sequence by establishing correspondences between points of video frames. In object tracking, detecting and tracking of the moving pixels are performed. The calculations of distance between pixels are carried out.

The detected pixels are connected from the first frame up to last frame, now the path is generated. Distances between pixels are calculated by using Euclidian distance function; it is shown in following Eqn,

$$\sqrt{(\boldsymbol{x_c} - \boldsymbol{x_p}) + (\boldsymbol{y_c} - \boldsymbol{y_p})}$$

x_c ,y_c is x and y value of current frame, x_p ,y_p is x and y value of previous frame of the video.

3. SYSTEM DESIGN AND IMPLEMENTATION

The main activities of the present systems are organized as three phases are shown (see Figure 3).

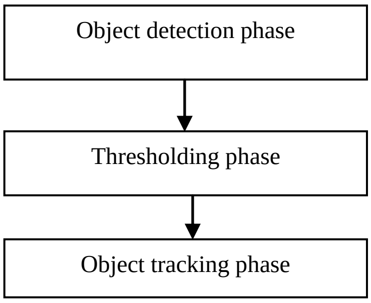


Fig 3: Three phases of the present work

In first phase, object detection is performed. Its flow chart is presented (see Figure 4). The image sequence (RGB) is taken an input by the system. The image sequence is converted into gray scale sequence. Then background subtraction is performed as detailed in section 2.1. The morphological operations are carried out to remove the noise. This operation is detailed in section 2.2. The background subtracted image sequence is obtained.

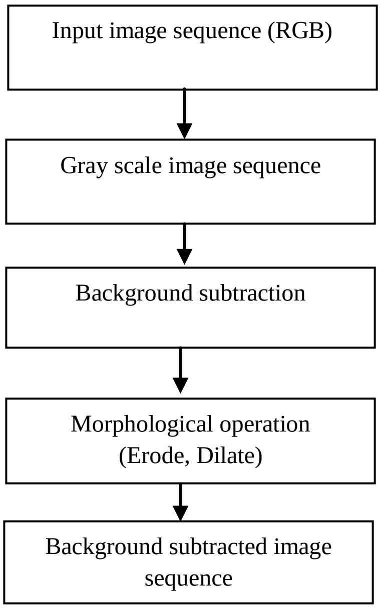

Input image sequence (RGB)
Gray scale image sequence
Background subtraction
Morphological operation (Erode, Dilate)
Background subtracted image sequence

Fig 4: Flowchart of object detection

In second phase, object extraction is performed. Three different threshold methods are applied on background subtracted image sequence. These methods were given in section 2.3. The results of three methods will be presented (see Figure 13, section 4). The flow chart of selection of threshold method is shown (see Figure 5).

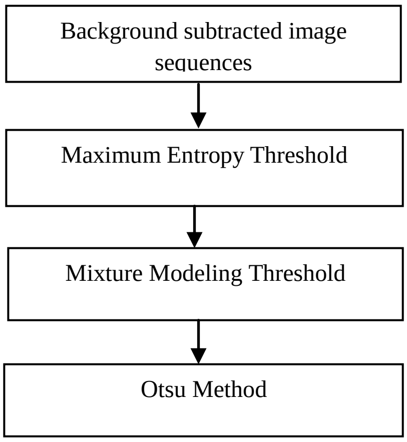

Background subtracted image sequences
Maximum Entropy Threshold
Mixture Modeling Threshold
Otsu Method

Fig 5: Flowchart of object extraction phase

In third phase, object tracking is performed. The flow chart of object tracking is shown (see Figure 6). In this phase, the path of moving object is generated, using feature point tracking approach (see section 4). The pixel positions are identified from the memory and displayed. The distance between the pixels are computed and displayed.

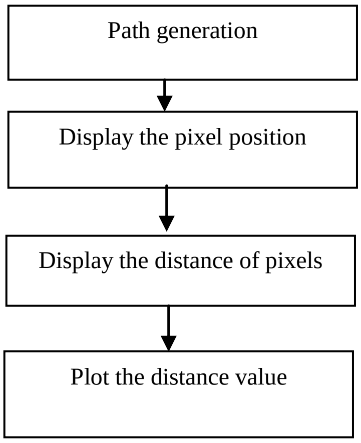


Fig 6: Flowchart of object tracking

The object detection and tracking system is implemented in ImageJ. ImageJ is implemented entirely in Java and is thus largely platform-independent, running without modification under Windows, Mac OS, or Linux. Java's dynamic execution model allows new modules ("plugins") to be written as independent pieces of Java code that can be compiled, loaded, and executed "on the fly" in the running system without the need to even restart ImageJ. It can display, edit, analyze, process, save and print 8-bit, 16-bit and 32-bit images. It can read many image formats including TIFF, GIF, JPEG, BMP, DICOM, FITS and "raw". It supports "stacks", a series of images that share a single window.

Plugins are small Java modules for extending the functionality of ImageJ by using a simple standardized interface. Plugins can be created, edited, compiled, invoked, and organized through the Plugin menu in ImageJ main window. Plugins can be grouped to improve modularity, and plugin commands can be arbitrarily placed inside the main menu structure. Also, many of ImageJ built-in functions are actually implemented as plugins themselves. Technically speaking, plugins are Java classes that implement a particular interface specification defined by ImageJ.

AVI video is converted into image sequences by using Virtual Dub software. Virtual Dub contains several menus, such as file, edit, audio, video etc. AVI video is displayed in Virtual Dub, from file menu (*File->Open video file*). This pane displays the AVI video; it shows (see Figure 7).

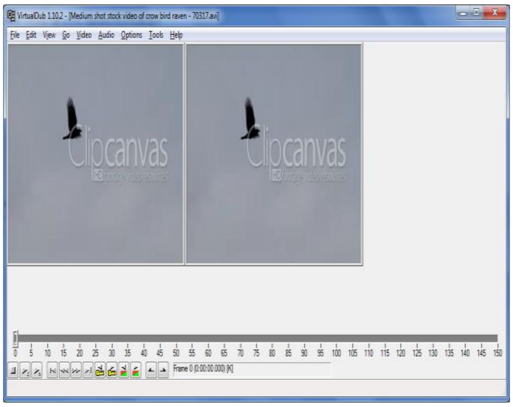

VirtualDub 1.10.2 - [Medium shot stock video of crow bird raven - 70317.avi]
File Edit View Go Video Audio Options Tools Help
Clipcanvas
Frame 0 (0:00:00.000) [K]

Fig 7: AVI video in Virtual Dub

The AVI video is split into number of image sequences (File->Export->Image sequence). Here, output format window is displayed, it contain file name, image format and file location. Now, the image sequence is stored in Input folder. This output format window is shown in Fig 8.

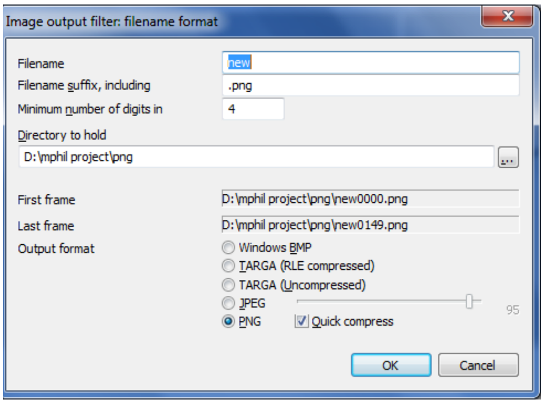
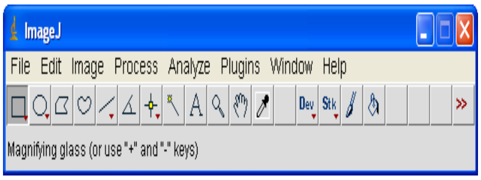


Image output filter: filename format

Filename: new
Filename suffix, including: .png
Minimum number of digits in: 4
Directory to hold: D:\mphil project\png
First frame: D:\mphil project\png\new0000.png
Last frame: D:\mphil project\png\new0149.png
Output format:
○ Windows BMP
○ TARGA (RLE compressed)
○ TARGA (Uncompressed)
○ JPEG 95
● PNG ☑ Quick compress

Fig 8: Image output filter

ImageJ window menus are File, Edit, Image, Process, Analyze, Window and Help. Each Menu performs unique operations. These menus are shown (see Figure 9).



Magnifying glass (or use "+" and "-" keys)

Fig 9: GUI for ImageJ

File menu (File>Import->Image sequence) is used to open input image sequence. Here, the input file is selected. Then, selection open window is displayed to specify the number of images, starting image, increment value etc. The input image sequence is displayed. This RGB image sequence is converted into gray scale image sequence by using Image menu (Image->Type->8 bit gray scale).

The coding for the object detection and tracking system is developed as ImageJ plugins. Java class of the coding is loaded into plugin folder of ImageJ. Then automatically, the present system coding file is displayed in plugin menu of ImageJ. There are five different types of plugins developed and performed in the present system, namely Background subtraction, Maximum entropy threshold, Mixture modeling threshold, Otsu method and Path tracking. This process is shown (see Figure 10).

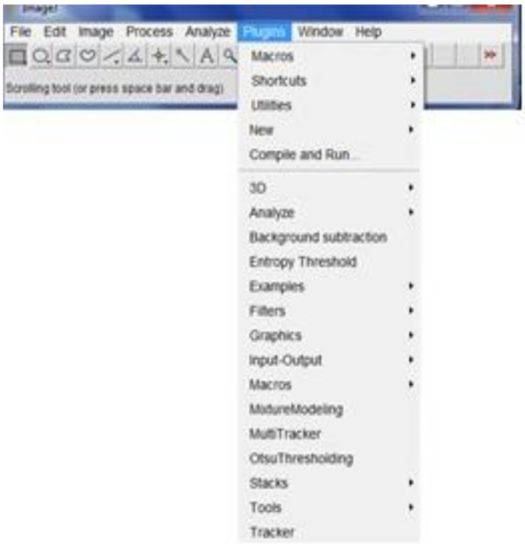


Fig 10: Plugins for object detection and tracking system

When background subtraction sub menu is clicked Java code of the phase 1 is executed. The results are presented and discussed (see section 4).

Maximum entropy threshold, Mixture modeling threshold and Otsu methods is performed in segmentation phase of present work. Maximum entropy threshold sub menu is selected to segment the object from an image sequence. It produces result of object extraction with some noise. When Mixture modeling threshold sub menu is clicked for extract the object, this method also produces noisy output. Otsu method sub menu is clicked for object extraction the object from an image sequence. It produces image sequence without noise output. These results will be discussed (see Figure 12).

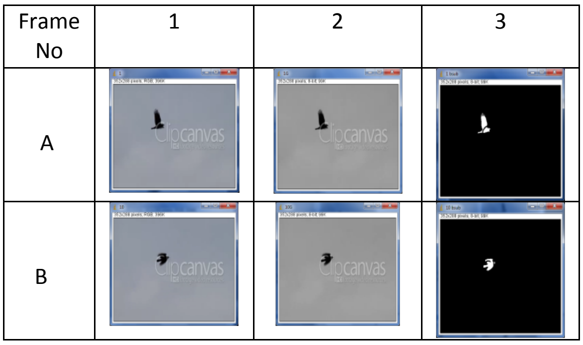
Third phase of present work is path tracking. It can be invoked by feature point tracking approach. This sub menu is selected to see the track of the moving object. It is shown (see Figure 13).

4. RESULTS AND DISCUSSION

The results of the system are given in detail in this section.

4.1 Object detection phase

The results of object detection are presented (see Figure 11). Here, RGB image sequence is converted into gray scale image sequence. Then the gray scale images are converted into binary image.



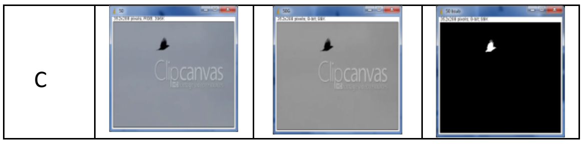


Fig 11: Results of object detection phase

(See Figure 11), column 1 denotes the input RGB image sequence (A1, B1, and C1). These image sequences are converted into gray scale image sequences shown in column 2 (A2, B2, C2). Column 3 shows the output of background subtraction (A3, B3, and C3). Row A is the 1st frame, row B indicates 10th frame, C denotes 50th frame of the video.

4.2 Thresholding phase

In next phase, the results of object extracted background subtracted images are presented (see Figure 12). It involves various threshold techniques. Threshold operation is performed before tracking of the moving objects. These background subtracted images are converted into threshold images.

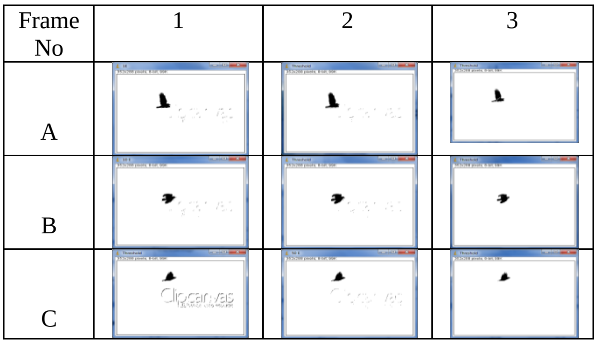


Fig 12: Result of Thresholding phase

(See Figure 12), column 1 (A1, B1, C1) shows the results of Maximum Entropy threshold, column 2 (A2, B2, C2) shows the results of Mixture Modeling and Column 3 (A3, B3, C3) shows the result of Otsu Method. Three types of threshold techniques were detailed in section 2. These techniques are applied to background subtracted image sequences (A3, B3, C3 in (see Figure 3) and the results are shown in (see Figure 12).

On seeing the images (see Figure 12), the noise is well removed in the images (see Column 3 of Figure 13) obtained by applied Otsu method. However, there is some noise present in the outputs of Maximum Entropy threshold and Mixture Modeling threshold (see Column 1 and Column 2 of Figure 12).

4.2.1 Mean and standard deviation of threshold methods

Maximum entropy and mixture modeling threshold method produces some noisy output and mean and standard deviation value is high. In otsu method to provide without noise output and the mean and standard deviation value is NaN. It is shown (see Table 1).

Table 1: Measurement of Mean and Standard Deviation

Threshold method	Mean	Standard Deviation
MaximumEntropyThreshold	251.31	30.44
MixtureModelingThreshold	251.97	27.602
Otsu Method	NaN	0

4.3 Object tracking phase

The third phase of the work is to find the track of moving object. Here, feature point tracking approach is applied. High intensity value of pixel is detected and those detected pixels are compared with those pixels of previous frame. The pixel position of the moving object are found out and plotted in new ImageJ window. It is shown (see Figure. 13). In present work video contain 150 frames. However, three image sequences are shown (see Figure 11 and Figure 12), for want of space.

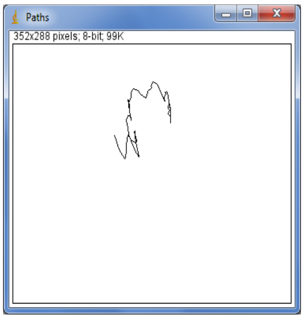


Fig 13: Path tracking

The position of pixel in each frame is obtained by using pixel matching technique. It is displayed (see Table 2).

Table 2: Positions of the pixel

Frame Number	X1	Y1
1	128.42	101.71
2	130.41	106.45
3	132.41	112.04
4	134.74	117.46
5	138.70	122.94
6	141.47	127.63
7	142.58	126.83
8	143.27	123.90
9	143.27	119.95
10	144.72	113.81
11	145.22	104.12
12	147.64	101.95

The moving pixels are identified in this process. The distance of pixel is calculated by using Euclidian distance function.

Current frame moving pixel position is same with moving pixel position of previous frame in the sense the distance value of pixel is 0. The pixel position is different in the sense that distance pixel is having some value not 0. This distance values are displayed (see Table 3).

Table 3. Distance between pixels

Frame Number	X1	X2	Y1	Y2	Distance
1	128.4	101.71	128.4	101.71	0.00
2	130.4	106.45	130.4	106.45	0.00
3	132.4	112.04	132.4	112.04	0.00
4	134.7	117.4	144.7	107.4	20.22
5	138.7	101.95	138.7	101.95	0.00
6	141.4	127.63	141.4	127.63	0.00
7	142.5	126.83	142.5	126.83	0.00
8	143.2	123.90	156.2	115.90	22.56
9	143.2	119.95	143.2	119.95	0.00
10	144.7	113.81	144.7	113.81	0.00
11	145.2	104.12	145.2	104.12	0.00
12	147.6	101.95	153.6	96.95	16.45

In plotting window, x coordinate value indicates number of frame and y coordinate value indicates distance of the pixel between two frames. Whenever frame is having some distance value, the peak will be increased otherwise peak will be 0. This plotting window displays the cleared result of distance variations; this result is shown (see Figure 14).

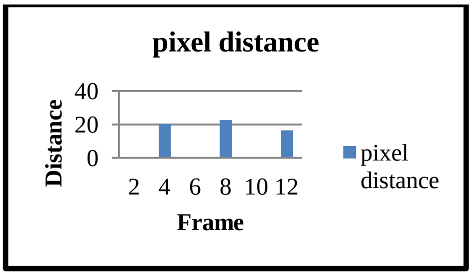


Fig 14: Plot the distance value

5. CONCLUSION

In this paper, a work on moving objects detection and tracking is performed based on various threshold methods. Object detection is performed by using background subtraction with Alpha method and morphological operations (Erode, Dilate). Object tracking is performed by using point tracking approach for tracking the moving object. To extract the detected object is carried out using three threshold techniques, namely Maximum Entropy threshold, Mixture Modeling Threshold and Otsu Method. The system yields best object extraction results during the work. 150 frames of a video are used for testing the system.

REFERENCES:

[1] R1. Joshi, Kinjal A., and Darshak G. Thakore, 2012. "A Survey on Moving Object Detection and Tracking in Video Surveillance System, "International Journal of Soft Computing and Engineering (IJSCE) ISSN", 2231-2307.

[2] Dheeraj Agrawal, Nitin Meena, 2013. "Performance Comparison of Moving Object Detection Techniques in Video Surveillance System" The International Journal of Engineering And Science (IJES), Volume. 2, Issue. 01, 240-242.

[3] Vision System for Relative Motion Estimation from Optical Flow: Sergey M. Sokolov, Andrey A. Boguslavsky, Felix A. KuftinKeldysh Institute for Applied Mathematics RAS Moscow, Russia. 2004.

[4] M. Yachida, M. Asada and S. Tsuji, 1981. "Automatic analysis of moving images", IEEE Trans. Pattern Anal. Mach. Intel., vol. PAMI-3, 12 -19.

[5] Visser, R., Sebe, N., & Bakker, 2002. Object recognition for video retrieval. In Image and Video Retrieval, Springer Berlin Heidelberg, 262-270.

[6] Häusler, G., and D. Ritter. 1999. "Feature-based object recognition and localization in 3D-space, using a single video image." Computer Vision and Image Understanding 73.1, 64-81.

[7] I. K. Sethi and R. Jain, 1987. "Finding trajectories of feature points in a monocular image sequence", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 9, 56 -73.

[8] C. Stauffer and W. E. L. Grimson, 1999. "Adaptive background mixture models for real-time tracking," in Proc. IEEE Conf. Computer Vision and Pattern Recognition.

[9] Elgammal, A., Dura swami, R., Harwood, D., Davis, L, 2002. Background and foreground modeling using nonparametric kernel density for visual surveillance. Proceedings of the IEEE 90, 1151-1163.

[10] T. Bouwmans, F. El Baf, and V. B, 2010. Statistical Background Modeling for Foreground Detection: A Survey, volume 4 of Handbook of Pattern Recognition and Computer Vision, chapter 3. World Scientific Publishing.

[11] Sebastian Brutzer, Benjamin H¨oferlin, 2011. "Evaluation of Background Subtraction Techniques for Video Surveillance" Gunther Heidemann Intelligent Systems Group, Universit¨at Stuttgart, Germany.

[12] J. Song, R. L. Stevenson, and E. J. Delp, 1989. "The Use of Mathematical Morphology in Image Enhancement", Proceedings of the 32nd Midwest Symposium on Circuits and Systems, Urbana-Champaign, IL, 67-70.

[13] Fathy, M., and Siyal, M.Y., 1995. "An image detection technique based on morphological edge detection and background differencing for real-time traffic analysis", Pattern Recognition Letters, 1321-1330.

[14] Sushil Kumar, et al., AMEA, in 2012. "2D Maximum Entropy Method for Image Thresholding Converge with Differential Evolution" Advances in Mechanical Engineering and its Applications (AMEA) Vol. 2, No. 3, 189-192.

[15] Jeon, J., and Manmatha, 2004. "Using maximum entropy for automatic image annotation" Springer link, Lecture Notes in Computer Science The, Vol. 3115/2004.

[16] J. Z. Liu and W. Q. Li, 1993. "The automatic thresholding of gray-level pictures via two-dimensional Otsu method," IEEE Int. Computer Vision and Pattern Recognition.