Temporal Tracking on Videos with Direction Detection

Shajeena Johnson
Department of Computer Science and Engineering, James College of Engineering and Technology, India

Abstract: *Tracking is essentially a matching problem. This paper proposes a tracking scheme for video objects on compressed domain. This method mainly focuses on locating the object region and predicting (evolving) the detection of movement, which improves tracking precision. Motion Vectors (MVs) are used for block matching. At each frame, the decision of whether a particular block belongs to the object being tracked is made with the help of histogram matching. During the process of matching and evolving the direction of movement, similarities of target region are compared to ensure that there is no overlapping and tracking performed in a right way. Experiments using the proposed tracker on videos demonstrate that the method can reliably locate the object of interest effectively.*

Keywords: *Motion vector, distance measure, histogram, block matching, DCT, tracking.*

Received August 19, 2014; accepted April 2, 2015

1. Introduction

Object tracking is one of the fundamental problems considering attention in the field of computer applications research, due to its applications in road traffic control, military guidance, surveillance system, visual navigation of robots and so on.

Visual features (like color, texture and shape) and motion details are the two major sources of information in video that can be used to track moving objects. Extraction of these two types of information can be done either in the pixel domain or in the compressed domain.

Despite of the recent progress in both pixel-domain and compressed-domain video object tracking the need for a framework with both reasonable accuracy and reasonable complexity still exists. The potential accuracy of the pixel domain approach is higher and also requires higher computational complexity.

Nowadays most of the video content are available in the compressed form, decoding is required in order to generate pixel domain information. Mean while the, "compressed-domain" approaches make use of the data from the compressed video bit stream, such as Motion Vectors (MVs), block coding modes, motion-compensated prediction residuals or their transform coefficients, etc.,

The most important advantage of compressed domain methods in practical applications is their lower computational cost. As a result, compressed-domain methods are thought to be more suitable for real-time applications although some of them are still characterized by high complexity. This paper presents a compressed-domain object tracking method that uses only MV's and block coding motion information to perform fast and fairly accurate tracking.

The remainder of this paper is organized as follows: section 2 briefly summarizes related work on object tracking, section 3 is devoted for analysing the object tracking problem in the compressed domain. Section 4 proposes the algorithm for object tracking and its direction of movement in video scenes. Section 5 presents the experimental results and provides insight into the utility and robustness of the proposed approach. Section 6 concludes this paper.

2. Background

Researchers have tried different approaches for object tracking. The main, non-mutually exclusive categories identified are: region-based tracking, active contour-based tracking, feature-based tracking, model-based tracking and body-part based tracking.

The method in [10] proposes an approach to develop a real-time object tracking system using histogram matching and absolute frame subtraction. Particle filtering based visual tracking approaches are used in [18, 19] it deals with non-linear dynamic system and can handle the multimodal status of the observation noise. Approach in [18] uses multiple features in combination with the particle filter algorithm in real time object tracking.

Contour-based approaches do not make use of the spatial and motion information of the entire object and rely on the information closer to the boundary of the video object [4, 6]. Contour-based methods can achieve a high tracking precision, but their robustness is usually not better than that of region-based methods.

There are some tracking methods that use both region and contour information [21].
A variation of conventional mean shift algorithm in combination with Kalman filter uses adaptive bandwidth and kernel weights are used in video-based target tracking [2, 16].

Background subtraction is a commonly used technique for segmenting object of interest in static scenes. The idea behind is to subtract a background model image from the current frame [17, 23].

More recently, there are some work aiming to recognize the category of the object incorporating high-level offline models and low-level online tracking models [3, 8]. In this work once an object is discovered, the tracking results are continuously fed forward to the upper-level video-based recognition scheme, in which the category of the object and its location can be recognized.

Some of the previous work targets on detecting objects of interest which is a convoy, using Multi-Target Tracking algorithm as well as modelling of objects of interest with graphic models. Many methods are designed for specific applications (e.g., using object models) [3, 7, 20] and many other methods employs features for tracking performance [1, 5, 15, 22].

The compressed domain algorithms [9, 12] exploit motion vectors or Discrete Cosine Transform (DCT) coefficients instead of original pixel data as resources in order to reduce computational complexity of object detection and tracking.

One of the advantage of the proposed approach compared to the state-of-the-art algorithm is that here the computational time is reduced. As this approach uses compression techniques instead of 64 pixels, only one pixel is used. Also most of the papers uses three frames to compute motion vector, in this approach only two frames are used to estimate the motion information.

In the proposed approach to detect the movement only the region of interest is searched and not the entire image. This saves the computation time. Again in most of the traditional algorithms, noise removal is done in a separate step. But in the proposed approach, as the DCT technique is used noise removal is done automatically.

3. Object Tracking Analysis

3.1. Feature Extraction

In static video object tracking the frame difference can be used to separate the moving object from the background. This specific theory of frame difference is done by calculating the same pixels feature value in two continuous frames, the difference of feature value δi between these two frames is calculated as:

$$\delta i = |F_{i+1} - F_i| \quad (1)$$

Here F_i denotes the current frame and F_{i+1} the successive frame. The possible movement area is assumed as region of interest. Hence the region of interest is estimated as

$$\epsilon = \delta i > \eta \quad (2)$$

Where η denotes the threshold value, which is calculated by,

$$\eta = \frac{max(\delta i) - min(\delta i)}{2} \quad (3)$$

So the frame difference can help us track more exactly in the static videos.

3.2. Motion Estimation

The motion compensation consists of taking a macro block from the current picture and determining a spatial offset in the reference picture at which a good prediction of the current macro block can be found. The offset is called the motion vector, see Figure 1. It consists of two components, x and y coordinates Δx and Δy.

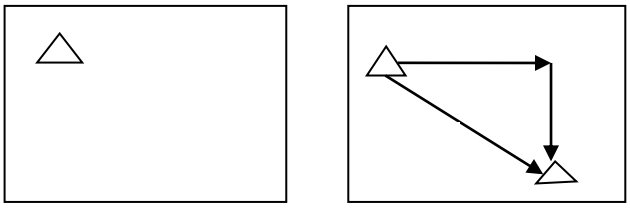


Figure 1. Image showing motion vector.

3.3. Distance Measure

The distance between two points is the length of the path connecting the points. In the plane, the distance between points (x_1, y_1) and (x_2, y_2) is given by the Pythagorean Theorem,

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (4)$$

In general, the distance between points x and y in a Euclidean space Rn is given by:

$$d = |x - y| = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \quad (5)$$

3.4. Image Compression

The DCT helps separate the image into parts (or spectral sub-bands) of differing importance (with respect to the image's visual quality).

The DCT is identical to the discrete Fourier transform: it transforms a signal or image from the spatial domain to the frequency domain.

The one-dimensional forward Discrete Cosine Transform (1-D DCT) of N samples is formulated by

$$F(u) = \sqrt{\frac{2}{N}} C(u) \sum_{x=0}^{N-1} f(x) cos\left[\frac{\pi(2x+1)u}{2N}\right] \quad (6)$$

The general equation for a 2D (N by M image) DCT is defined by the following Equation:

$$G_{ij} = \frac{1}{\sqrt{2n}} CiCj \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} Pxy \cos\frac{(2y+1)j\pi}{2n} \bullet \cos\frac{(2x+1)i\pi}{2n} \quad (7)$$

3.5. Histogram Equalization

The histogram of an image normally refers to a histogram of the pixel intensity values. It shows the number of pixels in an image at each different intensity value found in that image using a graph. An 8-bit grayscale image is said to have 256 different possible intensities, and so the histogram will provide a graphical display of 256 numbers showing the distribution of pixels amongst those grayscale values.

The image is scanned in a single pass and a running count of the number of pixels found at each intensity value is kept. This is then used to construct a suitable histogram.

Histograms have many uses. The most common one is to decide upon the value of threshold to use when converting a grayscale image to a binary one using threshold value. If the image is appropriate for thresholding then the histogram will be bi-modal i.e., the pixel intensities will be gathered around two well-separated values. A proper threshold for separating these two groups will be found somewhere in between the two peaks in the histogram. If the distribution is not so then it is doubtful that a good segmentation can be produced by thresholding.

3.6. Block Matching

Detecting the moving objects using two successive frames can be done by a procedure called block matching. Here the image frame is divided into non-overlapping square blocks. Each block from the source frame is verified for a corresponding block in the destination frame, by moving the source frame over a predefined region of pixels in the destination frame. At each move, the Euclidean distance between the gray values of the two blocks is computed. The shift which gives the smallest Euclidean distance is regarded as the best possible match.

4. The Proposed Algorithm

4.1. Outline of the Proposed Algorithm

In this algorithm, object tracking is performed by temporal tracking of a rectangle around the object at a reference frame. The algorithm is fully-autonomous in the sense that there is no human intervention and system itself does the low-level tasks, like motion detection and tracking. The input is the video sequence taken at the scene where surveillance is performed.

4.2. Low-Pass Filter for Image Smoothening

The first step is to capture a live video and take picture frames from it, upon which tracking techniques will be done to identify a moving object in the environment.

After which a low pass filter that is a mean filter is applied on the frames. This helps in smoothing the image as well as to reduce noise in images. Let Sxv represent the set of coordinates in a rectangular sub image window of size *m* x *n*, centered at point (*x*, *y*). The arithmetic mean filtering process computes the average value of the captured image *g*(*x*, *y*) in the area defined by Sxy. The value of the restored image at any point (*x*, *y*) is simply the arithmetic mean computed using the pixels in the region defined by S. In other words,

$$\hat{f}(x,y) = \frac{1}{mn} \sum_{(s,t)\in S_{xy}} g(s.t) \quad (8)$$

This operation can be implemented using a convolution mask in which all coefficients have value 1/mn. Now each input tile in Red Green Blue (RGB) space colour frame is transformed to YCbCr space. The transform takes an RGB input value with each component in the range [0-255] and transforms it into *Y*, *Cb*, and *Cr*, in the ranges [0.0, 255.0], [-128.0, 127.0], and [-128.0, 127.0], respectively. The matrix equation for this conversion is shown in the following equation:

$$[YCbCr] = [RGB] \begin{bmatrix} 0.299 & -0.168935 & 0.499813 \\ 0.587 & -0.331665 & -0.418531 \\ 0.114 & 0.50059 & -0.081282 \end{bmatrix} \quad (9)$$

Y: Luminance; *Cb*: Chrominance-Blue; and *Cr*: Chrominance-Red are the components. Luminance is same as the grayscale version of the original image. It is not necessary to keep all the information that is now represented in these color frames (*Cb* and *Cr*).

4.3. The Temporal Tracking

The aim of temporal tracking is to locate the object of interest in the successive frames based on the information about the object at the reference and current frames. To achieve this partition the Y plane of the image of size m x n into non-overlapping 8x8 blocks. For each block DCT coefficients are calculated, converting the original 8 x 8 array of pixel values into an array of coefficients according to Equation (6). Extract Direct Current (DC) coefficients which results in m/8 x n/8 matrix F_i for the ith frame.

Find the absolute difference, δi between the successive frames F_i and F_{i+1}. As per the procedure discussed in the section 3 and using the Equation (2) the area where the movement has occurred is the region of interest. The width of the region of interest $€_w$ can be calculated using Equation (10).

$$€_w = R\text{-}L \quad (10)$$

Where R is the rightmost column of Region of Interest (ROI) and L is the leftmost column of ROI.

The height of the region of interest $€_h$ can be calculated using Equation (11):

$$€_h = T\text{-}B \quad (11)$$

Where T is the topmost row of ROI and B is the bottommost row of ROI.

Let S_b be a block in ROI of F_i. Let D_b be a block in ROI of F_{i+1}. The initial width of the block B_w is computed by Equation:

$$B_w = €_w/3 \quad (12)$$

Similarly block height B_h can be calculated by equation:

$$B_h = €_h/3 \quad (13)$$

4.4. Temporal Difference Histogram

To approximate the direction of the object motion, the temporal difference histogram of two successive frames is defined. Coarseness and directionality of the frame difference of the two successive frames can be derived from the temporal difference histogram.

Let HS_b be the histogram of values in block S_b. Let HD_b be the histogram of values in block D_b. Let Ed denote the computed Euclidean distance of HS_b and HD_b. If the Euclidean distance Ed is less than t2, then the block S_b in F_i is matched with the block D_b in F_{i+1}.

So the movement is identified. If more than one source block is identified to move to a particular destination block then overlapping is identified, in order to avoid overlapping the size of source block is increased step by step until there is no overlapping in the identified object.

In order to confirm S_b is moved to D_b otherwise D_b is moved to S_b is tracked using the proposed approach.

4.5. The Ultimate Search

Let B_j, B_k represent blocks at location j and k of frames F_i and F_{i+1} respectively. Let F_i B_j represent block at location j in F_i

Movement of block from location B_j to B_k can be identified by checking, if F_i B_j is equal to $F_{i+1}B_k$ and $F_{i+1}B_j$ is equal to F_i B_k ,then F_i B_j is moved from B_j to B_k. If F_i B_k is equal to $F_{i+1}B_j$ and F_i B_j equivalent to $F_{i+1}B_k$, then F_i B_k is moved from B_k to B_j, The entire process is given in a step by step procedure as:

1. Capture the frames from the video clip.
2. Apply low pass filter on the captured frame.
3. Transform the captured RGB frame to YCbCr color space.
4. Divide the Y plane of the YCbCr frame into 8x8 blocks.
5. Find DCT coefficients for each block.
6. Extract DC coefficients which results in m/8 x n/8 matrix F_i for the ith frame.
7. Find the frame difference $\delta_i = |F_{i+1} - F_i|$.
8. Detect the area of movement(ROI) by checking if $\delta_i > t$.
9. Divide the ROI into uniform blocks.
10. Search each block S_b from (F_i in ROI) in (ROI of F_{i+1}) at D_b.
11.
    a) Find histogram of S_b, ie.,H_{sb}.
    b) Find histogram of D_b, ie., H_{db}.
    c) Find Euclidean distance of H_{sb} and H_{db} as E_d.
12. Match if $E_d < t_2$, movement is detected.
13. Detect the number of overlapping blocks while block matching, if number of overlaps is greater than one the size of the block is increased. As the block size is increased, the number of blocks gets reduced and the block matching process gets repeated from step 9.

Two way search to identify the source and destination: Let B_j, B_k represent blocks at location j and k of frames F_i and F_{i+1} respectively. Let F_i B_j represent block at location j in F_i. Movement of a block from location B_j to B_k can be identified by checking:

- *Case*1: if ($F_i B_j = F_{i+1}B_k$ && $(F_{i+1}B_j == F_i B_k)$) Then $F_i B_j$ is moved from B_j to B_k.
- *Case*2: $F_i B_k = F_{i+1}B_j$ && $(F_i B_j == F_{i+1}B_k)$

5. Results

To evaluate the performance of the proposed method, experiments were carried out using IDL6.3 on Windows platform. The tracking results are considered for various settings including person riding a bicycle, person in a motor bike and moving of a vehicle. The videos used for tracking were taken by stationary camera and the method works well with frames overlaid with noise and even with dark scenes.

Figure 2. Image with block wise divided ROI.

In Figure 2 the image can be seen where the Region of Interest is uniformly divided into 9 blocks. The number of blocks used helps to maintain the accuracy and robustness of tracking.

Figure 3. Image with overlapping motion vectors.

Here we see the effect of step 13 in Figure 3, where the estimated motion vector gets overlapped with existing motion vectors.

As the blocks overlaps while block matching, and matches at more than one location the size of the block is increased, this is shown in Figure 4.

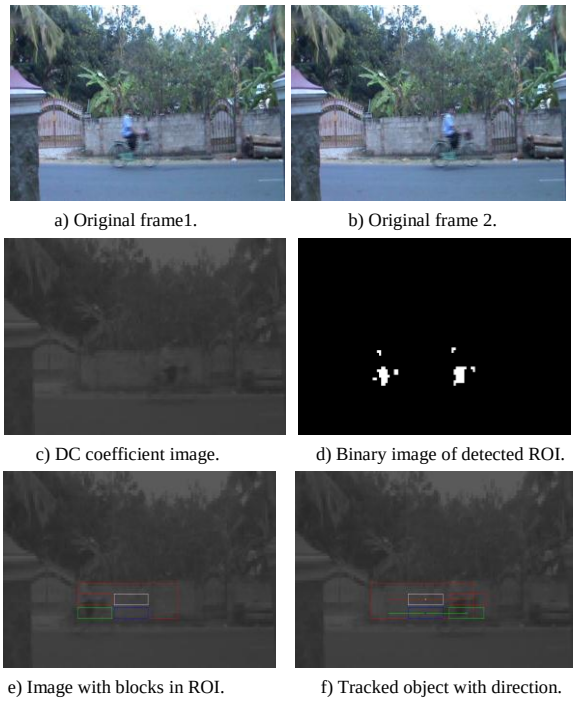
Figure 4. Image with block wise divided ROI (Block size increased).

The whole ROI is divided into 4 blocks. Similarly, as long as the number of overlapping is greater than one the process of division of ROI is repeated.

Figure 5. Tracked image with its direction of movement.

Thus after repeated division of ROI the problem of overlapping is overcome and the object is tracked with its direction of movement. The arrow indicated in the Figure 5 shows the object movement. This method handles objects of various sizes but it was not tested in crowded scenes.

To make the algorithm more understandable, some intermediate results of object tracking and direction detection are shown in Figure 6.



a) Original frame1. b) Original frame 2.

c) DC coefficient image. d) Binary image of detected ROI.

e) Image with blocks in ROI. f) Tracked object with direction.

Figure 6. Tracking a moving object.

Tracking performance evaluation of the proposed methodology is done by calculating sensitivity (Recall), Specificity, error percentage and accuracy, based on the number of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN) which is given as:

$$\text{Sensitivity} = TP/(TP+FN) \quad (14)$$

$$\text{Specificity} = TN/(TN+FP) \quad (15)$$

$$\text{Accuracy} = (TP+TN)/(TP+TN+FP+FN) \quad (16)$$

The performance of the proposed method is compared with other tracking methods like Extreme Learning Machine (ELM) [13] and Support Vector Machine (SVM) [14] and the comparison table, chart is shown in Table 1 and Figure 7 respectively.

Table 1. Comparison of several methods.

Method	Specificity %	Sensitivity %	Accuracy %
Proposed	96.75	88.96	96.63
SVM	97.98	94.17	92
ELM	98.45	94.77	92.8

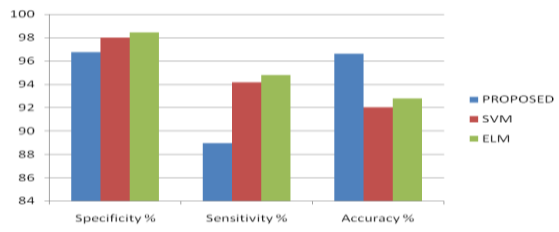


Figure 7. Performance comparison chart.

These plots illustrate the current performance of our system. The blue bars indicate the measures of the proposed system. Computation time, error percentage, accuracy, sensitivity and specificity are tabulated in Table 2 for different samples.

Table 2. Quantitative evaluation of test results.

Method	Specificity %	Sensitivity %	Accuracy %
Proposed	96.75	88.96	96.63
SVM	97.98	94.17	92
ELM	98.45	94.77	92.8

The experimental results show that the proposed method can efficiently track moving objects in indoor and noisy outdoor environment with better accuracy.

7. Conclusions

In this paper, an adaptive block-based approach is proposed for tracking moving objects along with direction prediction. Tracking is based on temporal matching and using a histogram, the object movement is identified. The method also tracks the direction of movement of the object of interest.

The problem of overlapping in block matching is solved by increasing the size of blocks.

Experimental results have shown that this approach obtains more accurate tracking results and tracks objects of various sizes and with large variations in illuminations even in outdoor surveillance scenarios.

Even though significant progress has been achieved for object tracking, there are still many challenging tasks requiring further investigation such as occlusion handling and tracking multiple objects.

References

[1] Amer A., "Voting -Based Simultaneous Tracking of Multiple Video Objects," *IEEE Transactions on Circuits and Systems for Video Tech*, vol. 15, no. 11, pp. 1448-1462, 2005.

[2] Chen Q., Sun Q., Heng P., and Xia D., "Two-Stage Object Tracking Method Based on Kernel and Active Contour," *IEEE Transactions on Circuits and Systems for Video Tech*, vol. 20, no. 4, pp. 605-609, 2010.

[3] Fan J., Shen X., and Wu Y., "What Are we Tracking : A Unified Approach of Tracking and Recognition," *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 549-560, 2013.

[4] Frost D. and Tapamo J., "Detection and Tracking Of Moving Objects in A Maritime Environment Using Level Set With Shape Priors," *EURASIP Journal on Image and Video Processing*, vol. 1, no. 42, pp. 1-16, 2013.

[5] Gao Y. and Dai Q., "View-Based 3-D Object Retrieval: Challenges and Approaches," *IEEE Multimedia*, vol. 21, no. 3, pp. 52-57, 2005.

[6] Hariharakrishnan K. and Schonfeld D., "Fast Object Tracking Using Adaptive Block Matching," *IEEE Transactions on Multimedia*, vol. 7, no. 5, pp. 853-859, 2005.

[7] Jiang H., Fels S., and Little J., "Optimizing Multiple Object Tracking and Best View Video Synthesis," *IEEE Transactions on Multimedia* vol. 10, no. 6, pp. 997-1012, 2008.

[8] Khan Z. and Gu I., "Nonlinear Dynamic Model for Visual Object Tracking on Grassmann Manifolds With Partial Occlusion Handling," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 2005-2019, 2013.

[9] Khatoonabadi S. and Bajie I., "Video Object Tracking in the Compressed Domain Using Spatio-Temporal Markov Random Fields," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 300-313, 2013.

[10] Mehta M., Goyal C., Srivastava M., and Jain R., "Real Time Object Detection and Tracking: Histogram Matching and Kalman Filter Approach," *in Proceedings of the 2nd International Conference on Computer and Automation Engineering*, Singapore, pp. 796-801, 2010.

[11] Pollard E., Rombaut M., and Pannetier B., "Situation Assessment: An End-to-End Process for the Detection of Objects of Interest," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2195-2210, 2013.

[12] Shajeena J. and Ramar K., "A Novel Way of Tracking Moving Objects in Video Scenes," *in Proceedings of International Conference on Emerging Trends in Electrical and Computer Technology*, Nagercoil, pp. 805-810, 2011.

[13] Selvy T., Devi R., Keerthini S., and Umamaheshwari S., "A Proficient Extreme Learning Machine Approach for Tracking and Estimating Human Poses," *International Journal of Engineering and Computer Science*, vol. 4, no. 3, pp. 10908-10913, 2015.

[14] Selvy T., "An Improved GA-MILSVM Classification Approach for Diagnosis of Breast Lesions from Stain Images," *International Journal of Advances in Engineering and Technology*, vol. 4, no. 2, pp. 216-227, 2012.

[15] Tang F. and Tao H., "Probabilistic Object Tracking With Dynamic Attributed Relational Feature Graph," *IEEE Transactions on Circuits and Systems for Video Tech*, vol. 18, no. 8, pp. 1064-1074, 2008.

[16] Tang Z., Sun C., and Liu Z., "The Tracking Algorithm for Maneuvering Target Based on Adaptive Kalman Filter," *The International Arab Journal of Information Technology*, vol. 10, no. 5, pp. 543-459, 2013.

[17] Varcheie P., Sills-Lavoie M., and Bilodeau G., "An Efficient Region-Based Background Subtraction Technique," *in Proceedings of Canadian Conference on Computer and Robot Vision*, Windsor, pp. 71-78, 2008.

[18] Wang Z., Zhao H., Shang H., and Qiu S., "An Improved Particle Filter for Multi-Feature

Tracking Application," *in Proceedings of IEEE International Conference on Imaging Systems and Techniques*, Manchester, pp. 522-527, 2012.

[19] Xi T., Zhang S., and Yan S., "Robust Visual Tracking Approach with Adaptive Particle Filtering," *in Proceedings of 2th International Conference on Communication Software and Networks*, Singapore, pp. 549-553, 2010.

[20] Yang T., Zhang Y., Tong X, Zhang X., and Yu R., "A New Hybrid Synthetic Aperture Imaging Model for Tracking and Seeing People Through Occlusion," *IEEE Transactions on Circuits and Systems for Video Tech*, vol. 23, no. 9, pp. 1461-1475, 2013.

[21] Ying-hong L., Yi-gui P., Zheng-xi L., and Ya-li L., "An Intelligent Tracking Technology Based on Kalman and Mean Shift Algorithm," *in Proceedings of 2th International Conference on Computer Modeling and Simulation*, Sanya, pp. 107-109, 2010.

[22] Zhao P., Zhu H., Li H., and Shibata T., "A Directional-Edge-Based Real-Time Object Tracking System Employing Multiple Candidate-Location Generation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 3, pp. 503-517, 2013.

[23] Zhou D., Zhang H., and Ray N., "Texture Based Background Subtraction," *in Proceedings of the International Conference on Information and Automation*, Changsha, pp. 601-605, 2008.

Shajeena Johnson M. Tech. is working as Asst. Professor in James College of Engineering and Technology, Kanyakumari District, India in the Dept. of Computer Science and Engineering. She has got a teaching experience of nearly 8 years. Her areas of interests are Image Processing and Medical Imaging.