ISSN: 0976-9102(ONLINE)
DOI: 10.21917/ijivp.2014.0138

# BLOCK-BASED TRACKING WITH TWO WAY SEARCH

**J. Shajeena[1] and K. Ramar[2]**

[1]*Department of Computer Science and Engineering, James College of Engineering and Technology, India*
E-mail: shajeenaelmo@gmail.com
[2]*Einstein College of Engineering, India*
E-mail: kramar.einstein@gmail.com

*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 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, Euclidean Distance, Histogram, Block Matching, DCT***

## 1. INTRODUCTION

During the recent years video object tracking has attracted more 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. Wide range of methods have been proposed and improved, from the simple and rigid object tracking under the condition of a static camera, to the non-rigid object tracking under the condition of a moving camera.

The task of analyzing videos to detect certain unusual activities is termed as video surveillance. Visual features (such as color, texture and shape) and motion information 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.

The potential accuracy of the pixel domain approach is higher and also requires higher computational complexity. Nowadays most of the video content is available in the compressed form and as a result it requires decoding in order to generate pixel domain information.

Nowadays most of the video content are available in the compressed form, decoding is required in order to generate pixel domain information. Meanwhile 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. In reality, not all but some of the information from the bit stream needs to be decoded.

However the compressed domain is referred so because they are fully decoded by the bit stream as long as pixel values are not recovered completely for all the frames. The lower accuracy is due to the shortage of full pixel information. The most important advantage of compressed domain methods in practical applications is their lower computational cost. This is possible because of the fact that part of decoding can be avoided, a small portion of data needs to be processed compared to pixel-domain methods, and some of the information produced during encoding (e.g., MVs) can be reused.

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 MVs 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 our approach. Section 6 concludes this paper.

## 2. BACKGROUND

Many 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 [2] 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 [3][15], it deals with non-linear dynamic system and can handle the multimodal status of the observation noise.

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][21]. 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 uses both region and contour information [5].

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 [6].

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 [7] [8].

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 [10][17]. 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 can be recognized and determined using the off-line model.

Some work target on detecting objects of interest which is a convoy, using Multi-Target Tracking algorithm as well as modeling of objects of interest with graphic models [16]. Many methods are designed for specific applications (e.g. using object models) [12][14][17], and many other methods employs features for tracking performance [11][18][19][20].

The compressed domain algorithms [1][9] exploit motion vectors or 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.

Movement in the proposed approach is detected by searching only the region of interest (ROI) 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 EXTRACT FEATURES

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 $\delta i$ between these two frames is calculated as,

$$\delta i = |F_{i+1} - F_i| \tag{1}$$

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

$$€ = \delta i > \eta \tag{2}$$

where, $\eta$ denotes the threshold value, which is calculated by,

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

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

### 3.2 MOTION VECTOR

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. It consists of two components, $x$ and $y$ coordinates $\Delta x$ and $\Delta y$.

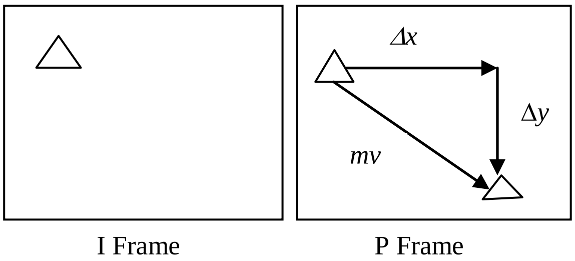

Fig.1. Image showing motion vector

### 3.3 EUCLIDEAN DISTANCE

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

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

In general, the distance between points $x$ and $y$ in a Euclidean space $R^n$ is given by,

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

### 3.4 DISCRETE COSINE TRANSFORM

The discrete cosine transform (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 similar 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,

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

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

$$C_{ij} = C_i C_j \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} P_{xy} \tag{7}$$

$$\theta_{xy} = \cos\frac{(2y+1)j\pi}{2n} . \cos\frac{(2x+1)i\pi}{2n} \tag{8}$$

$$G_{ij} = \frac{1}{\sqrt{2n}} C_{ij} . \theta_{xy} \tag{9}$$

The basic operation of the DCT is as follows:

1) The input image is $N$ by $M$
2) $f(i,j)$ is the intensity of the pixel in row $i$ and column $j$
3) $F(u,v)$ is the DCT coefficient in row $k_1$ and column $k_2$ of the DCT matrix
4) For most images, much of the signal energy lies at low frequencies, these appear in the upper left corner of the DCT

5) Compression is achieved since the lower right values represent higher frequencies, and are often small - small enough to be neglected with little visible distortion
6) The DCT input is an 8 by 8 array of integers. This array contains each pixel's gray scale level
7) 8 bit pixels have levels from 0 to 255

### 3.5 HISTOGRAM

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 graphically display 256 numbers showing the distribution of pixels amongst those grayscale values.

Histograms can also be taken of color images, either individual histogram of red, green and blue channels can be taken, or a 3-D histogram can be produced, with the three axes representing the red, blue and green channels, and brightness at each point representing the pixel count. The exact output from the operation depends upon the implementation, it may simply be a picture of the required histogram in a suitable image format, or it may be a data file of some sort representing the histogram statistics.

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. One of the more common is to decide what value of threshold to use when converting a grayscale image to a binary one by thresholding. If the image is suitable for thresholding then the histogram will be bi-modal i.e. the pixel intensities will be clustered around two well-separated values. A suitable threshold for separating these two groups will be found somewhere in between the two peaks in the histogram. If the distribution is not like this then it is unlikely 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 OVERVIEW 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 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 $S_{xv}$ represent the set of coordinates in a rectangular sub image window of size $m \times n$, centered at point $(x, y)$. The arithmetic mean filtering process computes the average value of the corrupted image $g(x, y)$ in the area defined by $S_{xy}$. 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,

$$\hat{f}(x,y)=\frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t) \tag{10}$$

This operation can be implemented using a convolution mask in which all coefficients have value $1/mn$. Now each input tile in 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:

$$\begin{bmatrix} Y & Cb & Cr \end{bmatrix}=\begin{bmatrix} 0.299 & -0.168935 & 0.499813 \\ 0.587 & -0.331665 & -0.418531 \\ 0.114 & 0.50059 & -0.081282 \end{bmatrix} \tag{11}$$

where, Y: Luminance, Cb: Chrominance-Blue, and Cr: Chrominance-Red are the components. Luminance is very similar to the grayscale version of the original image. We really do not need 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 $m \times n$ into non-overlapping 8 × 8 blocks. For each block DCT coefficients are calculated, converting the original 8 × 8 array of pixel values into an array of coefficients according to Eq.(6). Extract DC coefficients which results in $m/8 \times n/8$ matrix $F_i$ for the $i^{th}$ frame.

Find the absolute difference, $\delta_i$ between the successive frames $F_i$ and $F_{i+1}$. As per the procedure discussed in the section 3 and using the Eq.(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 Eq.(10).

$$€_w = R\text{-}L \tag{12}$$

where, $R$ is the rightmost column of ROI and $L$ is the leftmost column of ROI.

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

$$€_h = T\text{-}B \tag{13}$$

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 Eq.(14).

$$B_w = €_w/3 \tag{14}$$

Similarly block height $B_h$ can be calculated by Eq.(15).

$$B_h = €_h/3 \tag{15}$$

### 4.4 TEMPORAL DIFFERENCE HISTOGRAM

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

Let $H_{sb}$ be the histogram of values in block $S_b$. Let $H_{db}$ be the histogram of values in block $D_b$. Let $E_d$ denote the computed Euclidean distance of $H_{sb}$ and $H_{db}$. If the Euclidean distance $E_d$ is less than $t_2$, 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 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 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 8 × 8 blocks.
5) Find DCT coefficients for each block
6) Extract DC coefficients which results in $m/8 \times n/8$ matrix $F_i$ for the $i^{th}$ 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 entire 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 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.

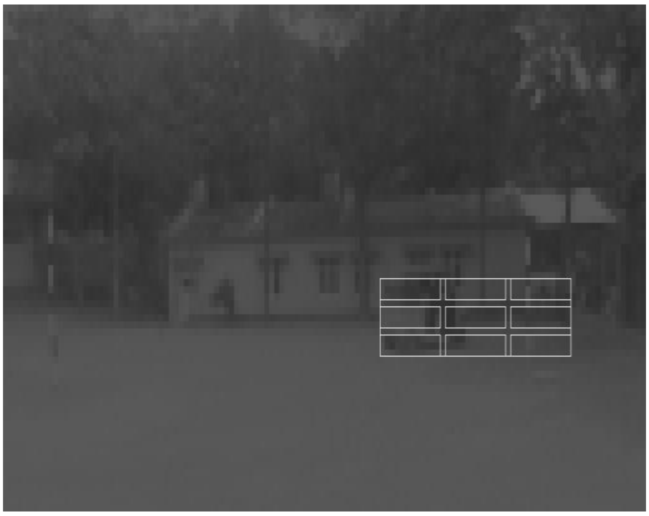

Fig.2. Image with block wise divided ROI

In Fig.2, Region of Interest is uniformly divided into 9 blocks. The number of blocks used helps to maintain the accuracy and robustness of tracking.

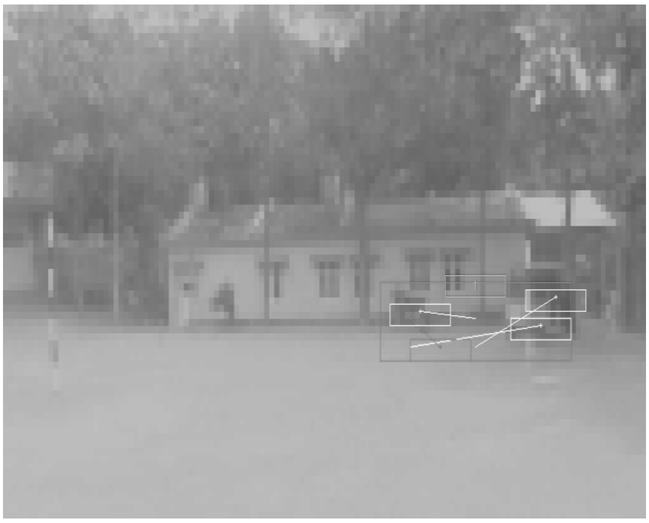

Fig.3. Image with overlapping motion vectors

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

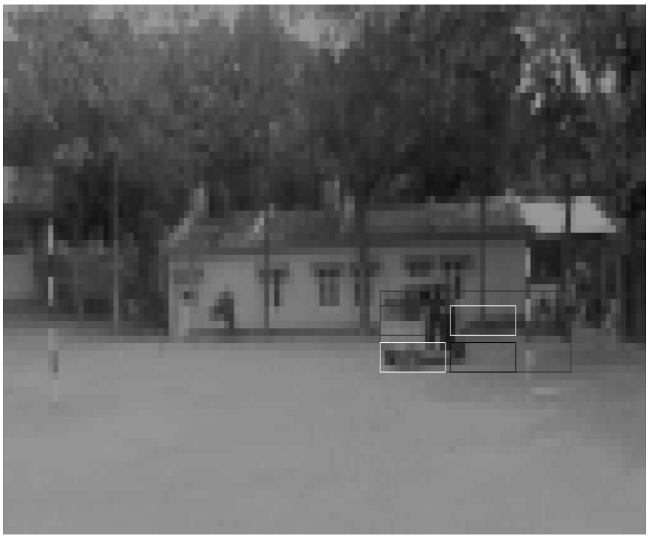

Fig.4. Image with block wise divided ROI (Block size increased)

As the blocks overlaps while block matching, and matches at more than one location the size of the block is increased. The whole Region of Interest 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.

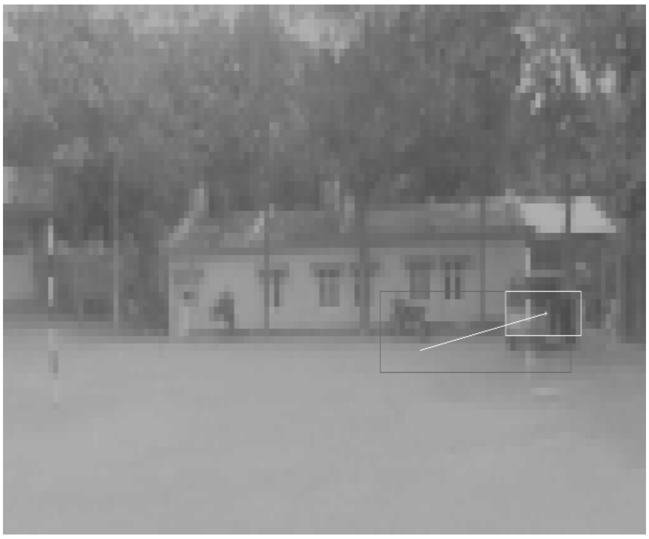

Fig.5. Tracked image with its direction of movement

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

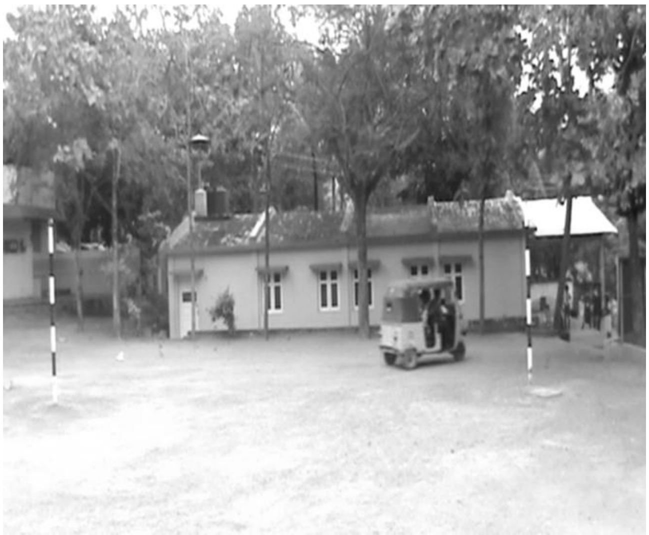

Fig.6(a)

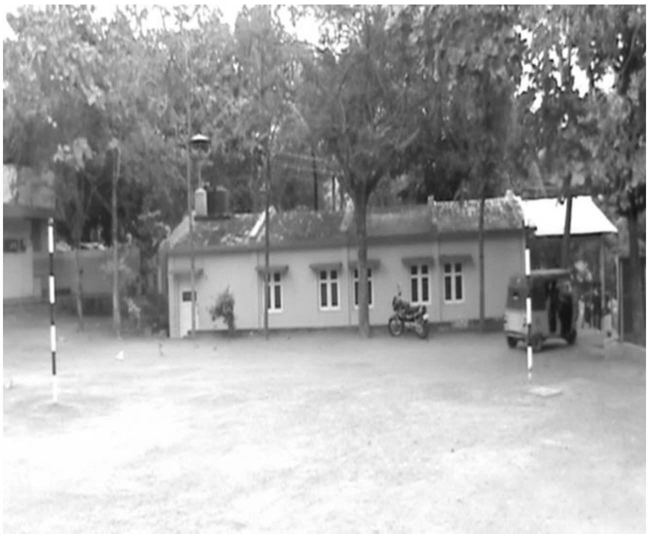

Fig.6(b)

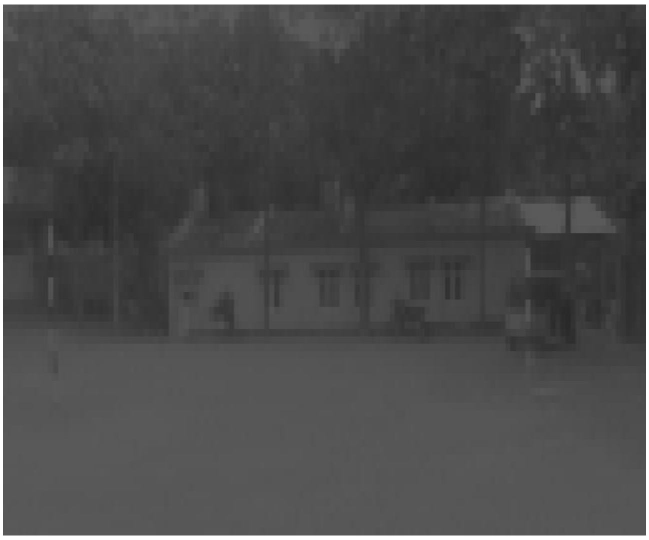

Fig.6(c)

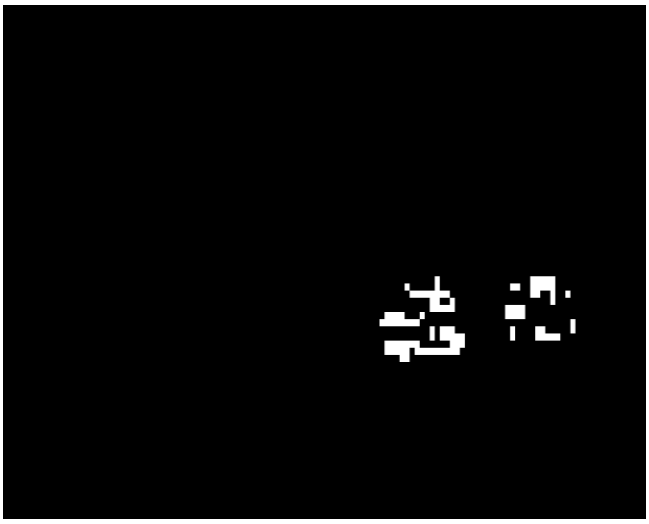

Fig.6(d)

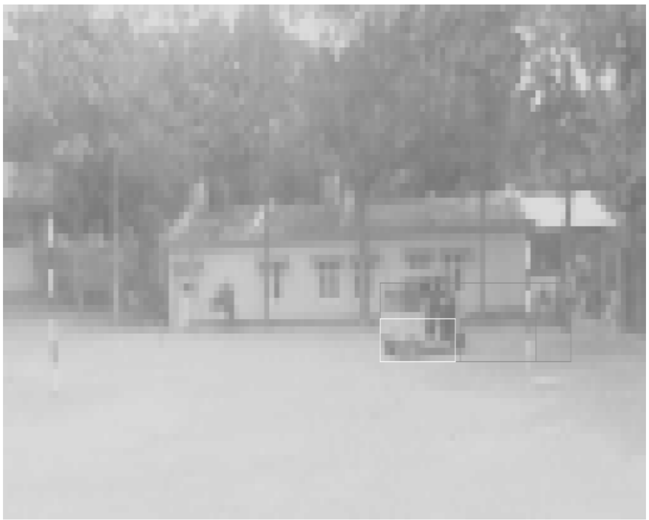

Fig.6(e)

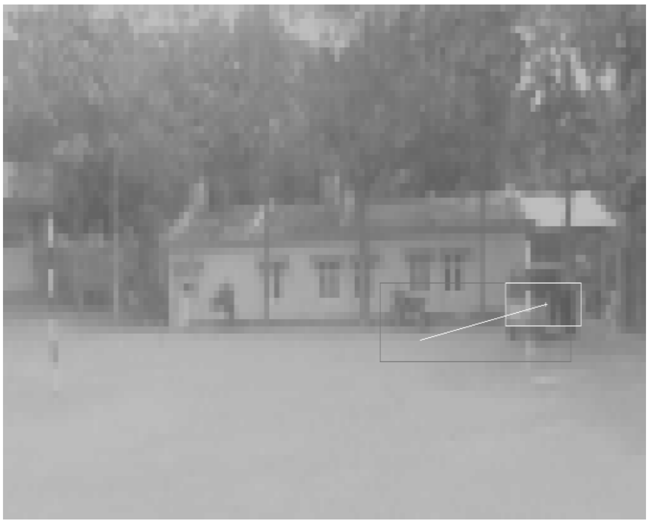

Fig.6(f)

Fig.6. Tracking a moving object: (a). Original frame1, (b). Original frame2, (c). DC coefficient image, (d). Binary image of detected ROI, (e). Image with blocks in ROI, (f). Tracked object with direction

## 6. CONCLUSION

In this paper, an adaptive block-based approach is proposed for tracking moving objects. 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] J. Shajeena and K. Ramar, "A novel way of tracking moving objects in video scenes", *IEEE International Conference on Emerging Trends in Electrical and Computer Technology*, pp. 805-810, 2011.

[2] Madhur Mehta, Chandni Goyal, M .C. Srivastava and R. C. Jain, "Real time object detection and tracking: Histogram matching and kalman filter approach", *Second IEEE International Conference on Computer and Automation Engineering*, Vol. 5, pp. 796-801, 2010.

[3] Xi Tao, Zhang Shengxiu and Yan Shiyuan, "Robust Visual Tracking Approach with Adaptive Particle Filtering", *Second IEEE International Conference of Communication Software and Networks*, pp. 549-553, 2010.

[4] Karthik Hariharakrishnan and Dan Schonfeld, "Fast Object Tracking Using Adaptive Block Matching", *IEEE Transactions on Multimedia*, Vol. 7, No. 5, pp. 853-859, 2005.

[5] Li Ying-hong, Pang Yi-gui, Li Zheng-xi and Liu Ya-li, "An Intelligent Tracking Technology based on Kalman and Mean Shift Algorithm", *IEEE Second International Conference on Computer Modeling and Simulation*, Vol. 1, pp. 107-109, 2010.

[6] Qiang Chen, Quan-Sen Sun, Pheng Ann Heng and De-Shen Xia, "Two-Stage Object Tracking Method Based on Kernel and Active Contour", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 20, No. 4, pp. 605-609, 2010.

[7] Parisa Darvish Zadeh Varcheie, Michael Sills-Lavoie and Guillaume-Alexandre Bilodeau, "An Efficient Region-Based Background Subtraction Technique", *IEEE Canadian Conference on Computer and Robot Vision*, pp. 71-78, 2008.

[8] Dongxiang Zhou, Hong Zhang and Nilanjan Ray, "Texture Based Background Subtraction", *IEEE International Conference on Information and Automation*, pp. 601-605, 2008.

[9] Sayed Hossein Khatoonabadi and Ivan V. Bajie, "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] Zulfiqar Hasan Khan and Irene Yu-Hua Gu, "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.

[11] Feng Tang and Hai Tao, "Probabilistic Object Tracking With Dynamic Attributed Relational Feature Graph", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 18, No. 8, pp. 1064-1074, 2008.

[12] Tao Yang, Yanning Zhang, Xiaomin Tong, Xiaoqiang Zhang and Rui Yu, "A New Hybrid Synthetic Aperture Imaging Model for Tracking and Seeing People Through Occlusion", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 23, No. 9, pp. 1461-1475, 2013.

[13] Weiming Hu, Xi Li, Wenhan Luo, Xiaoqin Zhang, Stephen Maybank and Zhongfei Zhang, "Single and Multiple Object Tracking Using Log-Euclidean Reimannian Subspace and Block-Division Appearance Model", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 34, No. 12, pp. 2420-2440, 2012.

[14] Hao Jiang, Sidney Fels and James J Little, "Optimizing Multiple Object Tracking and Best View Video Synthesis", *IEEE Transactions on Multimedia*, Vol. 10, No. 6, pp. 997-1012, 2008.

[15] Anlong Ming, Huadong Ma and Charles X. Ling, "Object Tracking in Frame – skipping Video Acquired Using Wireless Consumer Cameras", *International Journal on Advanced Robotic Systems*, Vol. 9, No. 131, pp. 1-11, 2012.

[16] Evangeline Pollard, Michele Rombaut and Benjamin Pannetier, "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.

[17] Jialue Fan, Xiaohui Shen and Ying Wu, "What Are we Tracking: A Unified Approach of Tracking and Recognition", *IEEE Transactions on Image Processing*, Vol. 22, No. 2, pp. 549-560, 2013.

[18] Pushe Zhao, Hongbo Zhu, He Li and Tadashi Shibata, "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.

[19] Aishy Amer, "Voting – Based Simultaneous tracking of Multiple Video Objects", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 15, No. 11, pp. 1448-1462, 2005.

[20] Yue Gao, "View-based 3-D Object Retrieval: Challenges and Approaches", *IEEE Multimedia Magazine*, Vol. 21, No. 3, pp. 52-57, 2014.

[21] Duncan Frost and Jules-Raymond Tapamo, "Detection and tracking of moving objects in a maritime environment using level set with shape priors", *EURASIP Journal on Image and Video Processing*, Vol. 2013, No. 42, pp. 1-16, 2013.