I J C T A, 9(41) 2016, pp.919-932 ISSN: 0974-5572

FAST AND PRECISE METHOD FOR OBJECT DETECTION IN REAL-TIME VIDEO APPLICATION

PrathameshGangal*, and VishalSatpute**

Abstract: For any modern surveillance system, object detection is prime business. The object detection includes both, locomotive as well as static objects detection. The major targets associated with object detection strategies are to quickly detect the objects and keep a track of them. To deal with these issues, in this paper, we have proposed a simple and elegant technique using variance based approach, which is similar to the human vision system for targeting both speed and accuracy aspects. Detection of edge information is a crucial parameter to detect moving or non-moving objects. The proposed mechanism uses Discrete Wavelet Transform for edge detection which indirectly helps to ease the guessing of object location's initial estimate in frame. This will be followed by the variance based approach which is useful in exact calculations of the locations of the object and for its tracking. As stated earlier, the DWT is used in this work, but, specifically, Haar wavelet is used here due to its ease of implementations on 2D frames and also due to its inherent properties. To evaluate the performance of the proposed system, it has been tested on various videos and its performance is compared with various existing techniques prevalently used.

Key Words: *2D-DWT; variance; object detection and tracking; Haar;surveillance; video processing*

1. INTRODUCTION

With increase in anti-social and terrorist activities, it is very essential to have some kind of mechanism, which will tackle the security aspects of the public at large. To do so the tool available is video surveillance. With popularity of various video surveillance products, it is being used widely now a day in the public places such as at airports, railways stations, banks, ATM, etc. where many people are allowed to access different services. With tremendous improvements in the field of VLSI, chip designing and in imaging devices, the resolution of images or video captured is increased very abruptly. With the progress in resolution, it directly improved the quality of videos captured and at a very low cost. Further, it fueled expectations from automated video surveillance systems. With the increase in resolution, now systems need to handle and operate on a huge amount of data, which leads to decrease in the speed of operation of system. Generally, it is observed that, for a video surveillance system, there exists three different jobs to be exercised i.e. first object detection, then tracking of the moving objects and finally storing of thesevideos. This leads to development of multiple technological fields including movement detection,

* Department of Electronics & Communication Engineering, VNIT Nagpur, 440010, India. Email: prathamesh.gangal@gmail.com
** vrsatpute@ece.vnit.ac.in

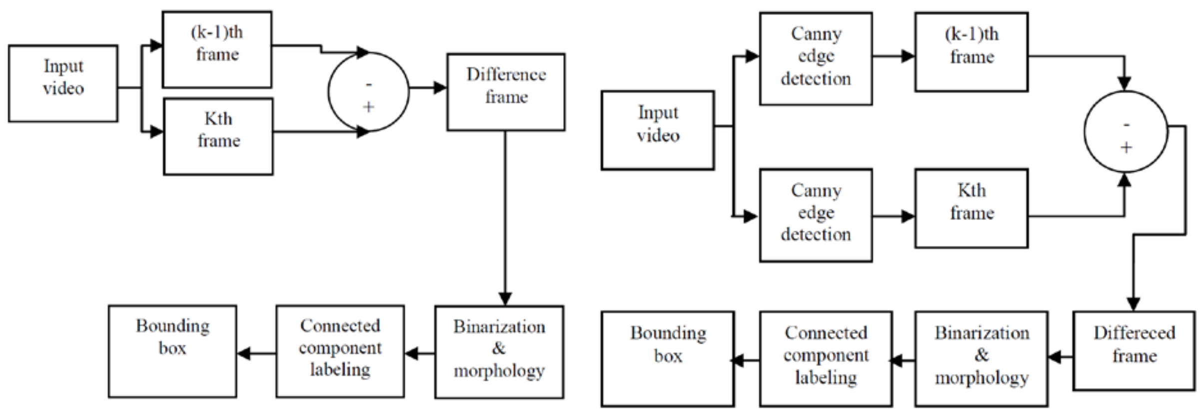
object tracking and behavioral prediction, etc. The most tedious job amongst them in the video surveillance is object detection. As for human beings it is very easy to identify the objects, there movements besides their behavioral analysis in a very quick manner since the human vision and thinking was developed for millions of years. To make the automated video surveillance so robust will need a lot of efforts and research. As humans detect the object based on relative difference between foreground object and background pixels, this principle need to be used to detect the objects by an automated system. To do so there exists multiple methods and techniques available in the literature but, a lot of them suffer from accuracy, computational complexity, or from timing requirements point of view winning in any one or may be in at the most two parameters. Again, after object detection, tracking of these detected moving objects is a much-pertained task to be accomplished. In some of the conventional object detection approaches, like statistical algorithm or optical flow algorithm, accuracy is very promissory but, computational complexity as well as timing requirements are very high which fails the system for its suitability in live video surveillance applications. In contrary, a very fast method available called as differencing of frame operation/process and/or subtraction of background operation/process, which produces the results very rapidly, but suffers from accuracy point of view [1-3]. Hence, it is also not suitable for applications where accuracy is prime concerns. There are certain other issues with these methods, i.e. for detecting stationary objects, the worthy selection is background subtraction. This technique works perfect under the conditions that camera movement is stationary and ambient lighting changes is almost constant. But, maintaining such conditions every time may not be possible at every deployed place. On the same way, the Frame differencing technique is most suitable for detection of moving objects. Thus, it is obvious that there exists a trade-off between accuracy and computational timing requirements. Desired choice is to have more accuracy and in less time. With the improvements in video resolution, it is becoming difficult to reduce timing requirements for object detection, this is creating a major hurdle in object detection process, and hence, some alternate strategy is need of the hour. The variance based approach for detecting object is projected to deal with these issues. This technique gives better accuracy and in lesser time. DWT has information of both spatial and frequency domain in its components and it is found that its higher frequency components carry edge information [3-6]. This edge information is very useful for initial estimate of the object and with the variance approach finally it leads to quick identification of these objects.Proposed algorithm's performance is trialed with competitive equivalent methods to find its suitability for real time applications. As stated earlier, DWT is used in this work, but, specifically, Haar wavelet is used here due to its ease of implementations on 2D frames and also due to its inherent properties.

2. PREVALENTLY USED OBJECT DETECTION ALGORITHMS

There exists a variety of algorithms available for object detection out of which the mostly preferred algorithms are selected and preferred here. They also acts as benchmarks for comparison of performance of the proposed variance based algorithm. These existing algorithms are discussed here in brief.

2.1 Frame differencing and Background subtraction method-

Frame difference and Background subtraction are the simplest and fastest algorithms available for object detection. These algorithms are simplest algorithms available in which a simple subtraction operation is to be carried out. In background subtraction method, the current video frame is subtracted from reference video frame for detection of any changes occurring on the current frame with respect to that of the reference frame. Those changes will be reported as the changes observed in the video and will be used for object detection job. On the other hand, in frame differencing method, the subtraction of frames are carried out in a successive manner i.e. the current frame is subtracted from the previous frame and this process is continued by changing current frame to previous frame and new frame will become current frame. Resultant output will be the changes observed in current frame with respect to the previous frame. These methods are very simple to implement and are executed at a real time. However, major problem associate with these methods is linked with lighting conditions of video. These methods are highly susceptible to noise. Slight variations in lighting conditions or noise available in video will lead to fall in accuracy levels and in practical conditions; it is too difficult to maintain lighting conditions under prefect control. Even though thesemethods are simple and fast but cannot be used effectively for practical considerations [2]. This process is represented in block diagrammatical way in figure 1(a).



(a) (b)

Fig.1. (a) Frame differencing (b) Frame differencing with edge detection method

2.2 Frame differencing with edge detection method-

It is an improved method of previous frame differencing method to minimize the noise and illumination variation effect by applying edge detection process. In this method, edges of objects are found out by using Canny edge detector. This method improved the accuracy to some extend but due to additional edge detection process, speed of operation was lowered down than that of previous method [2]. This process is represented in block diagrammatical way in figure 1(b). In this method, edge detection of current and previous frame is computed followed by carrying frame differencing as previously done.

In both these methods, to minimize the effect of noise and lighting conditions, at the end thresholding operation is carried out. These operations will reduce effect of noise and lighting conditions on the results. Morphological operations are followed to ensure removal of available noise particles nearby object area. This guarantees the correctness of detection quality. [2] There exists many techniques for edge detection purpose out of which authors have proposed Canny edge detector operator as it gives accurate and consistent edge width which will guarantee the correctness of the detection quality. This process includes following steps [2] [3].

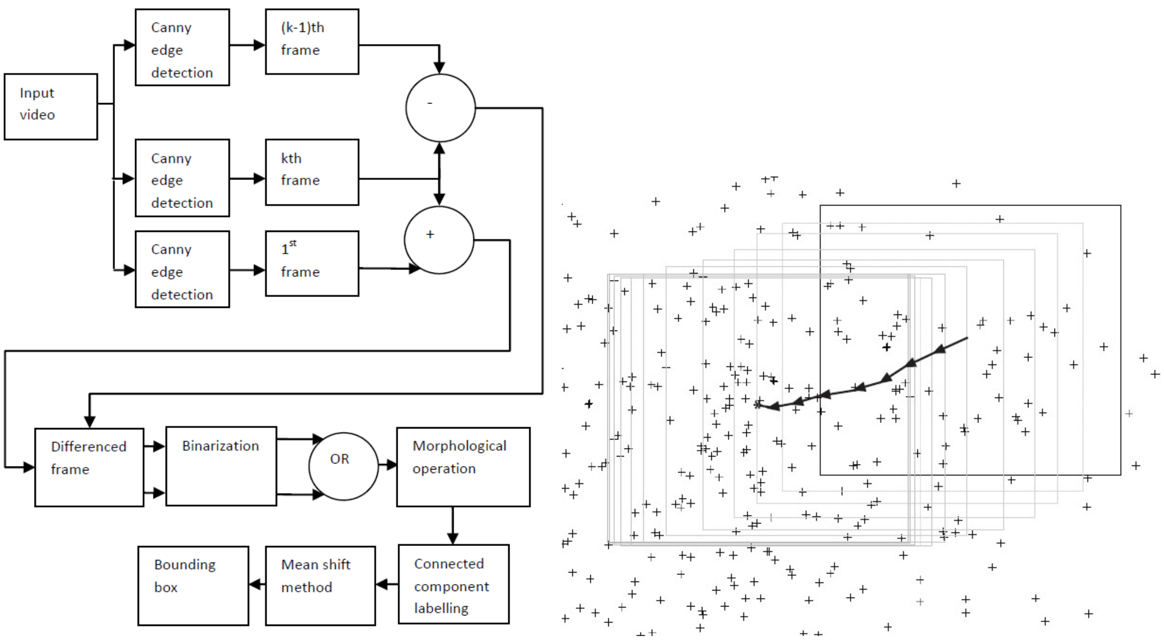
1. Noise removal using Gaussian filter.
2. Magnitude and direction gradient calculation.
3. Suppressing non maximum pixels along the norm of edge using edge direction information to makeedge as thin as possible.
4. Double threshold processing is to be carried out if required.

2.3 Mean-shift method-

Even though frame differencing methods are very promising with respect to computational timing requirements, but suffers from the problem of accuracy due to which more advancement in previous method is needed for betterment in detection process of the object. To do this, authors have proposed a method termed as mean shift. This method has suggested changes in the process of object localization. In this method, at the beginning, background subtraction and frame differencing both methods are carried out, so that moving as well as stationary objects can be captured. Then, thresholding operation of both frames is carried out followed by OR operation on both these threshold resultant frames. All these steps may introduce some amount of noise which will affect further processes and effectively the accuracy hence, to eliminate the noise effects, morphological operations were required. Finally, mean shift technique is carried out to get dense cluster of object pixels. Mean shifttechnique works on the basic principle of finding centroid of object pixels which is explained as follows [7-8].

1. With certain initial location, choose a search window with details of its type and shape.
2. Compute centroid of search window.
3. This computed centroid will form new location of search window. Hence, relocate it.
4. For convergence, repeat steps 2,3until window stops relocating.

This method improved the accuracy statistics but is more time consuming due to repetitive processes of mean calculations. This method gives more accuracy compared to existing mechanisms, but major retiring factors of it is that it requires a huge amount of computational time and is not suitable for real time applications. Mean shift method is represented in block diagrammatic way in figure 2(a). Process of mean calculations and convergence is shown in figure 2(b).



(a) (b)

Figure 2. (a) Mean shift method (b) Mean value calculations

2.4 Guassian Mixture Model (GMM)-

GMM is parametric probability density function, which is represented as weighted sum of Gaussian component densities. GMMs are commonly employed as a parametric model of the probability distribution of continuous measurements or features in many biometric systems. By using GMM background model, frame pixels are deleted from the required video to achieve the expected results. Application of background subtraction involves a multiple factors. These factors involve developing an algorithm which will be able to detect required objects. In order to give an inner understanding of this algorithm used for background subtraction, the algorithm follows the steps as.

1. Firstly, each input pixels is compared to mean 'μ ' of the associated components. If value of this pixel is close enough to a chosen component's mean, then, that component is referred as the matched component. To be a matched component, it is essential to have the difference between the pixel and

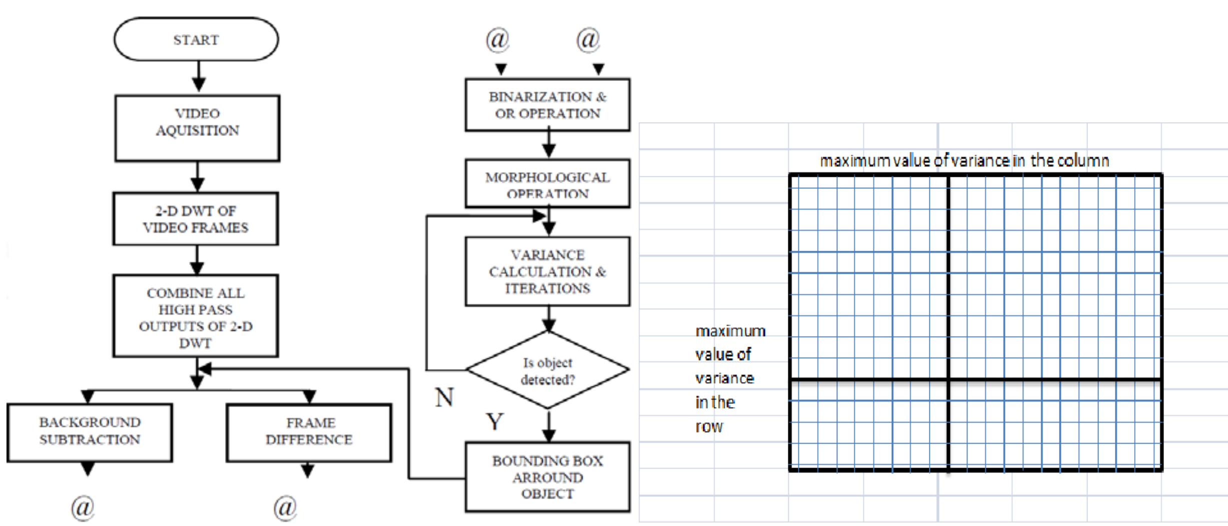
the mean must be less than the component's standard deviation which is a scaled version by factor D in the algorithm.

2. Secondly, updating the Gaussian weight, mean and standard deviation to reflect new computed pixel value. In relation to the components which are non-matched, the weights 'w' decreases while mean and standard deviation remains unchanged. It is fully dependent upon the learning component 'p' with the relation to how fast they change.
3. Thirdly, here we identify which components are parts of the background model. To perform this task, a threshold value is applied to the component weights 'w'.
4. Finally, in the last step the foreground pixels are determined. One important point to note here is that the pixels which are identified as foreground pixels, does not match with any components decided as background.

This method is complex and lot many computations are expected to be done on each pixel so as to identify objects. The proposed variance method has an advantage of both better accuracy and also lesser computational timings.

3. PROPOSED METHOD: VARIANCE BASED APPROACH FOR OBJECT DETECTION

The proposed method includes an alternate strategy based on human vision system to identify and detect moving as well as stationary objects under different conditions. This method not only gives edge information but also used for object detection and for tracking moving objects. The proposed method uses DWT as a basis tool for getting edge information through high pass components of DWT. This edge information is very useful for an initial estimate of object location and helps in extracting shape of object. The whole process of proposed algorithm is presented here and is shown in figure 3(a) as flow chart.



(a) (b)

Figure 3. Proposed algorithm (a) Flow chart (b) Variance Computations

3.1 Discrete Wavelet Transform-

DWTis used to transform a discrete time signal into its corresponding discrete wavelet domain. It transforms any series say, $X_0, X_1 \dots X_m$ into two different components viz. high-pass and low-pass coefficients [3-5]. Haar wavelet is referred here for entire work looking at its ease of implementation and has many underlying properties. DWT outputs the given frame into four different components namely, LL, LH, HL and HH, where L stands for Low-pass and H for High-pass. DWT uses filters for low-pass and high-pass information separation as given in equation (1) and (2).

$$\widetilde{H}_k = \sum_{m=0}^{\omega-1} X(2i-m).S_m(z) \tag{1}$$

$$\widetilde{L}_k = \sum_{m=0}^{\omega-1} X(2i-m).T_m(z) \tag{2}$$

Where wavelet filters are denoted by $S_m(Z)$ and $T_m(Z)$, ωdefines filter bank length, and k represents frame number [3]. Application of DWT is done in two steps leading to 4 components viz. LL, LH, HL & HH shown in figure 4(a)[6]. Except LL, rest of the components i.e. LH, HL and HH are termed as high-pass components [9]. These high-pass components contain information about sudden change in intensity i.e. nothing but edge information. Hence, these components are used for edge detection purpose and will help us to get an initial estimate of object's locations in corresponding frame. This decomposition is shown in figure4(b), 4(c) for an example frame.

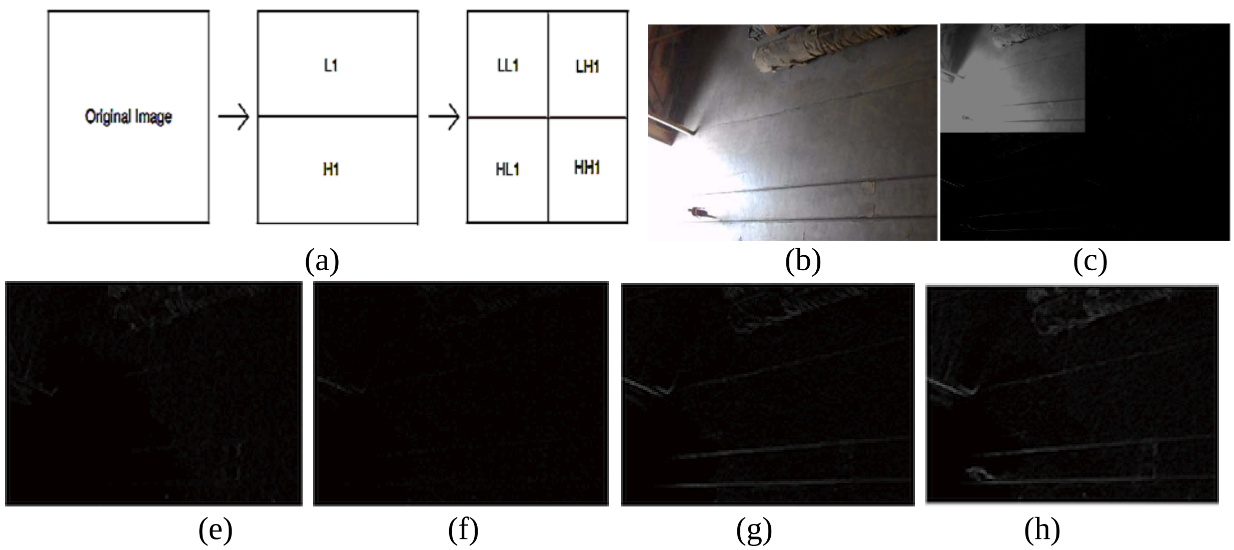


Figure 4. (a) DWT Decomposition (b) Original frame (c) Decomposition result (d)HL (e) LH (f) HH (g)Fusion of HL,LH and HH components.

3.2 Edge Detection using high pass components-

Edge detection of frames can be done by using all high pass coefficients of DWT decomposed output. Looking carefully at theseDWT outputs, one can very easily conclude that high-pass components LH contains horizontal, HL contains vertical and HH contains diagonal edge information. Considering only one or two component(s) will definitely lose the remaining information. Hence, to get all the information a simple trick is applied i.e. all these high pass outputs are fused together into a single component by spatially adding them together. This insures that full edge information is available in the fused component as expressed in equation (3).

$$\hat{F}(i,j) = \sum_{i=1}^{r/2} \sum_{j=1}^{c/2} (LH(i,j) + HL(i,j) + HH(i,j)) \tag{3}$$

Where r, c are used to indicate original frame size. Decomposed components in terms of row and columns are 1/4th of the original frame due to level-1 2D-DWT. With reference to figure4(c), LL, HL and HL parts are shown in figure 4(d), (e) and (f) respectively, and figure 4(g) represents result of fusion of these components. This edge detection process helps us in an initial estimate of object's location and hence the full frame search for object detection is reduced to a small and localized area. This improves speed of operation of the proposed algorithm.

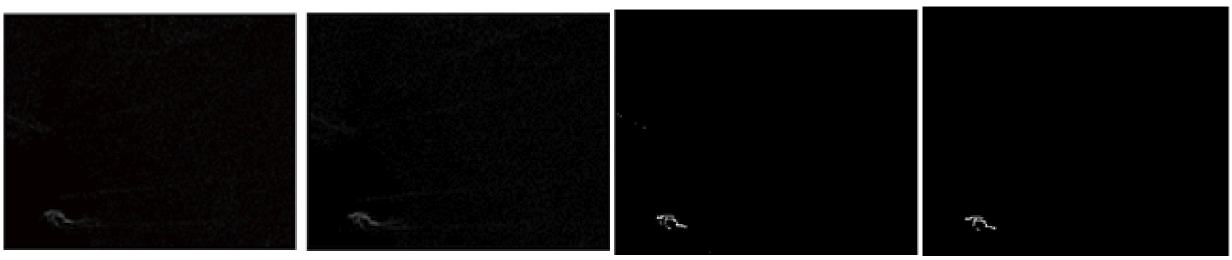
3.3 Details about Frame Differencing and Background Subtraction-

These two techniques are basis for object detection mechanisms and are frequently used as discussed. When camera position is relatively fixed and for stationary objects, background subtraction is used, in which the first frame is considered as background frame. Background frame is subtracted from current frame to estimate the object [1] [3]. However, on the contrary, when current frame is subtracted from subsequent frame, then we call it as frame differencing operation. Both the methods can be illustrated by using equations (4) and (5).

$$\{\widehat{BS}(i,j)\}_k = \{\hat{F}(i,j)\}_k - \{\hat{F}(i,j)\}_{k=0} \tag{4}$$

$$\{\widehat{FD}(i,j)\}_k = \{\hat{F}(i,j)\}_k - \{\hat{F}(i,j)\}_{k-1} \tag{5}$$

Where i, j represents spatial locations and can take values from 1 to r/2and 1 to c/2 respectively, k indicates frame number. Results of these operations are shown in figure 5(a) and (b).



(a) (b) (c) (d)

Figure 5. Process of Morphological Operation: (a) BS output (b) FD output (c)Noisy binarized output (d) Morphological operation result

3.4 ***Threshold Computation: Binarization-***

Computation of threshold is a process of converting an image into a specific range of pixels. When outputs of thresholding process are binary it is called as binarization. Using this process, binary output of corresponding frame is obtained by a threshold. All the values above this threshold are taken as white else black. Threshold selection is to be done for each frame as there may exist changes in the illumination conditions in the video. This drastically affects algorithm's performance. To perform computation of threshold, one needs to compute mean and standard deviation for each frame [10]. This determines threshold for binarizationof each frame. Equations(6) to (8) are used to compute mean, standard deviation, and threshold respectively.

$$\mu_k = \frac{1}{RC}\sum_{i=1}^{r/2}\sum_{j=1}^{c/2}\{\widehat{FD}(i,j)\}_k, \tag{6}$$

$$\sigma_k = \sqrt{\frac{1}{RC}\sum_{i=1}^{r/2}\sum_{j=1}^{c/2}(\{\widehat{FD}(i,j)\}_k - \mu_k)^2} \tag{7}$$

$$T_k = 0.06 * \sigma_k \tag{8}$$

In which, r, c are original frame sizes, μ_k is the mean of, $\sigma_k$represents standard deviation. Threshold (T) is taken as 0.06 times of standard deviation fork^{th} frame. Constant 0.06 is obtained experimentally [10].The OR operation is carried out on binarized form as given in equation (9).

$$\{RV(i,j)\}_k = \{BS(i,j)\}_k \; OR \; \{FD(i,j)\}_k \tag{9}$$

Where RV is resultant frame of OR operation w.r.t. figure 5(a) and (b) is shown in figure 5(c).

3.5 Morphological Operations-

Video RV is the result of edge detection, binarization etc. which introduces some noise in the frames because no thresholding method is as ideal as to be noise free. Change in background illumination may also cause to generate noise component as shown in figure 5(c). Hence to remove noise from frames, morphological operations are carried out on each frame of video RV. Opening and closing are performed to eliminate noise [4]. Equation (10) and (11) shows opening and closing operation.

$$A \circ B = (A \ominus B) \oplus B \tag{10}$$

$$A \bullet B = (A \oplus B) \ominus B \tag{11}$$

Where A is main image and B is structuring element. Closing operation is performed to eliminate issues created by opening operation such as loosing of object area [3,5]. The result of morphological operations for figure 5(c)is shown in figure 5(d).

3.6 Detection of objects-

Here we propose new method for localization of object in the frame area by computing the variance of rows and columns of corresponding frame [11].This operation localizes the object by finding non-zero value of variance in the frame.Variance is obtainedby equation (12) [12].

$$\sigma^2 = \frac{1}{\alpha^2} \sum_{x-1}^{m} \sum_{y-1}^{n} I^2{}_{sub}(x, y) \tag{12}$$

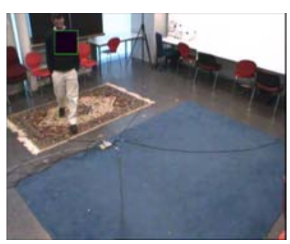
Where, $I_{sub}$is part of the frame under considerations, and α is total pixels there. Figure 3(b) illustrates process of object localization. To get initial estimate of the object's location, intersection point of the row and column is taken where variance is maximum. A window is formed around this intersection pixel point. To converge, the process is repeated by shifting the window center to thus computed point.To indicate the presence of object a bounding box is placedon object. A tracker can be initiated to keep a track of objects which helps security personnel for behavior analysis.

4. RESULTS AND DISCUSSIONS

This section presents details of various experiments carried out here. All algorithms described in this work have been implemented and executed using MATLAB.Various video datasets are used to test algorithm's performance. Computational resource used in this entire work is a desktop computer with i7-IV generation, with3.4GHz processor. The algorithms were tested on various videos downloaded from standard databases, andon locally captured videos [13-15].

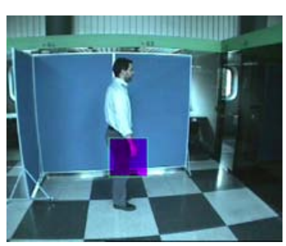
Lighting conditions are non-standard while capturing videos. Table-1 indicates video database details. Figure 6 shows detected objects usingthe proposed variance based method with pink bounding box placed at object's location.

(a) campus4-c1_1, Frame#262



(b) 4p-c0_1,Frame#368

(c) terrace-c2_1, Frame#220



(d) Visor_01, Frame#147 (e) Visor_04, Frame#524 (f)Visor_07, Frame#115

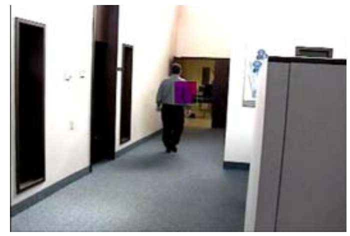
(g) cam4, Frame#25

(h) domotica_01, Frame#466

(i) Video8, Frame#422

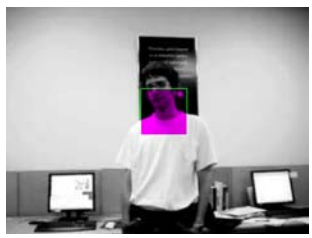
(j) backdoor3, Frame#267

(k) pedestrain_1, Frame#177



(l) cubical1, Frame#675

(m) menearbfp, Frame#201



(n) 02-SurfaceCover_video00012, Frame#16

(o) motinas_toni, Frame#147

(p) MVI_04, Frame#2

(q) visor_car, Frame#120

(r)winterDriveway, Frame#83

Fig. 6 Results for proposed algorithm

To evaluate performance of the proposed algorithm, it is compared with existing algorithms as discussed. All algorithms are executed on videos as specified in Table 1 hence uniformity in the comparison is maintained. Performance analysis is carried out based on two parameters "accuracy of detection" and "computational timings". The accuracy is computed using true-false analysis (TFA). The TFA parametersare,

- True positive (TP) indicates no. of frames of the video in which object is there and it got detected correctly.
- True negative (TN) represents object is not there in the video and algorithm has failed to detect it.
- False Positive (FP) indicates algorithm do not detect the object even though it is there.
- False Negative (FN) indicates algorithm detects the object even though it is not there.

Table-1: Video details

Human Database			
Video Name	**Resolution**	**No. of frames**	**Database**
campus4-c1_1	480*856	448	CV Lab
campus4-c2_1		440	
passageway1-c3_1		190	
terrace1_c3_2		409	
4p-c0_1		434	
terrace-c1_1		451	
terrace-c2_1		377	
visor_01	288*384	702	VISOR
video8		426	
visor_02	480*856	983	
visor_03		486	
visor_04	288*352	2308	
Tardini2		346	
visor_05	256*320	137	
visor_06	288*360	336	
visor_07		250	
visor_08		211	
cam4	240*340	104	
vdo17	480*640	58	
vdo37		67	
vd039		61	
domotica_01	272*368	756	
domotica_02		708	
domotica_03		252	
domotica_04		391	
backdoor3	480*856	462	CDNET
backdoor4		395	
Copymachine1		1050	
cubical1		1160	
cubical2		415	
cubical3		534	
office_1		416	
pedestrain_1		311	
peopleinshade_1		228	
sofa_1		478	
abandonedBox	288*432	167	
meanarbfp	480*640	543	CAVIAR
meanarbfp1		581	
motinas_toni	576*720	430	
motinas_toni_change_ill		680	
motinas_nikola_dark		910	
02-SurfaceCover_video00003	1080*1920	210	ALOV300+
02-SurfaceCover_video00012	240*320	532	
06-MotionSmoothness_video00016		1385	
	Total Frames	22218	
Non-human Database			
vdo1.mp4	360*480	54	IPCV_Lab
vdo2.mp4		131	
vdo3.mp4		67	
vdo4	1200*1600	58	
MVI_01	480*640	104	
MVI_02		22	
MVI_03		53	
MVI_07		29	
MVI_08		165	
MVI_04	480*856	84	

Human Database			
Video Name	Resolution	No. of frames	Database
MVI 05	240*320	74	
MVI 06.mpg	270*480	182	
cycle	368*656	99	VISOR
visor car	288*384	770	
winterDriveway	240*320	580	CDNET
	Total Frames	2472	

True negative condition may occur due to noise present in the video or very small part or very small object may be present. False negative may occur due to noise available in video, either because of non-standardlighting condition or thresholding. Table 2 shows detection statistics for all methods including the proposed one.

Table-2: Comparative analysis of detection statistics

Human Database								
Method	Detection Statistics				Percentage Detection			
	TP	TN	FP	FN	TP	TN	FP	FN
FD with Centroid	16162	3445	2466	142	72.753	15.508	11.101	0.639
FD with edge detection	5563	1474	13019	2159	25.042	6.635	58.605	9.719
Mean shift	16162	3453	2463	137	72.753	15.564	11.102	0.618
GMM	6275	3461	12349	130	28.247	15.580	55.589	0.585
Variance	16560	3523	2055	77	74.543	15.859	9.251	0.347
Non-Human Database								
FD with Centroid	1061	724	656	29	42.955	29.312	26.559	1.174
FD with edge detection	351	474	1364	281	14.211	19.190	55.223	11.377
Mean shift	1063	723	652	32	43.036	29.271	26.397	1.296
GMM	640	658	1076	96	25.911	26.640	43.563	3.887
Variance	1081	727	634	28	43.765	29.433	25.668	1.134

The bar-charts for TFA for respective databasesare shown in chart 1 and 2.

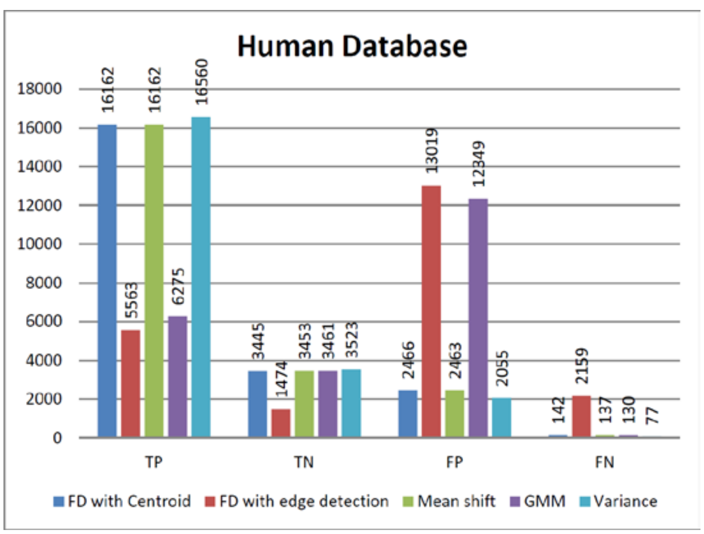


Chart-1: TFA Bar chart

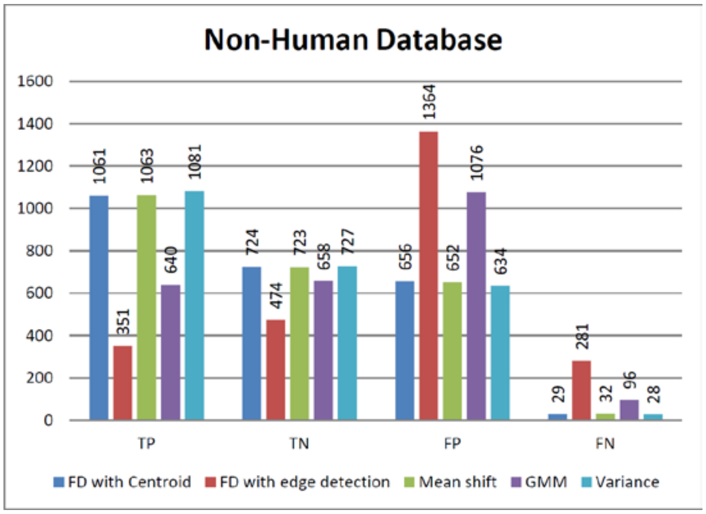


Chart-2: TFA Bar chart

Table 3 shows comparative timing analysis of all algorithms.

Table-3: Timing analysis

Human Database					
Video Name	**Frame Diff**	**Edge Detection**	**Mean Shift**	**GMM**	**Variance**
campus4-c1 1	**10.197**	**17.522**	**10.866**	**17.109**	**9.511**
campus4-c2 1	**9.440**	**17.096**	**10.279**	**16.605**	**9.349**
passageway1-c3 1	**4.069**	**7.821**	**4.832**	**7.114**	**4.851**
terrace1 c3 2	**8.512**	**15.340**	**9.443**	**15.510**	**8.313**
4p-c0 1	**9.037**	**16.246**	**10.067**	**16.084**	**8.861**
terrace-c1 1	**9.895**	**17.998**	**11.454**	**17.697**	**9.707**
terrace-c2 1	**8.037**	**14.685**	**9.086**	**14.028**	**7.760**
visor 01	**4.920**	**7.590**	**7.631**	**10.475**	**4.676**
visor 02	**22.966**	**42.590**	**27.061**	**43.473**	**22.283**
visor 03	**11.021**	**18.571**	**11.374**	**18.703**	**10.151**
visor 04	**15.874**	**24.014**	**23.006**	**35.097**	**15.136**
visor 05	**2.859**	**1.310**	**3.363**	**4.008**	**2.827**
visor 06	**7.198**	**3.758**	**8.600**	**9.751**	**7.093**
visor 07	**5.349**	**2.739**	**6.066**	**7.372**	**5.285**
visor 08	**5.241**	**2.681**	**6.025**	**7.285**	**5.160**
cam4	**1.401**	**0.991**	**1.099**	**8.496**	**0.615**
vdo17	**0.960**	**1.622**	**1.083**	**1.980**	**0.919**
vdo37	**1.143**	**2.025**	**1.244**	**2.459**	**1.128**
vd039	**1.108**	**1.873**	**1.329**	**2.176**	**1.061**
domotica 01	**4.977**	**7.427**	**7.515**	**10.545**	**4.762**
domotica 02	**4.774**	**7.039**	**6.501**	**10.001**	**4.391**
domotica 03	**1.729**	**2.670**	**1.969**	**3.822**	**1.630**
domotica 04	**2.786**	**4.318**	**4.280**	**5.723**	**2.675**
Tardini2	**2.417**	**3.546**	**3.643**	**5.087**	**2.297**
video8	**3.029**	**4.670**	**3.808**	**6.350**	**2.905**
backdoor3	**10.442**	**17.978**	**10.835**	**18.099**	**9.901**
backdoor4	**10.186**	**15.168**	**10.068**	**15.438**	**8.150**
Copymachine1	**26.148**	**44.570**	**29.817**	**47.676**	**25.089**
cubical1	**28.761**	**49.720**	**29.510**	**50.810**	**27.790**
cubical2	**8.683**	**15.690**	**10.000**	**15.778**	**8.527**
cubical3	**11.283**	**20.485**	**12.904**	**21.045**	**10.967**
office 1	**8.716**	**16.040**	**10.393**	**16.246**	**8.506**
pedestrain 1	**6.247**	**11.635**	**7.198**	**11.433**	**6.130**
peopleinshade 1	**5.004**	**8.840**	**5.604**	**8.978**	**4.817**
sofa 1	**10.674**	**18.281**	**11.920**	**19.256**	**10.008**
abandonedBox	**1.407**	**2.075**	**1.848**	**2.810**	**1.312**
meanarbfp	**15.329**	**15.663**	**17.820**	**21.915**	**15.132**

Human Database					
Video Name	**Frame Diff**	**Edge Detection**	**Mean Shift**	**GMM**	**Variance**
meanarbfp1	**16.416**	**16.677**	**18.556**	**23.486**	**16.114**
motinas_toni	**6.832**	**17.352**	**11.169**	**17.345**	**6.798**
motinas_toni_change_ill	**16.396**	**28.931**	**18.827**	**30.818**	**15.860**
motinas_nikola_dark	**22.702**	**40.777**	**29.432**	**42.251**	**22.101**
02-SurfaceCover_video00003	**34.787**	**47.999**	**32.833**	**54.764**	**31.444**
02-SurfaceCover_video00012	**10.772**	**4.335**	**13.139**	**14.840**	**10.696**
06-MotionSmoothness_video00016	**27.849**	**16.627**	**33.868**	**38.294**	**27.655**
Non-human Database					
vdo1.mp4	**0.557**	**2.474**	**1.637**	**1.461**	**2.021**
vdo2.mp4	**1.274**	**2.178**	**1.455**	**3.477**	**1.403**
vdo3.mp4	**0.695**	**1.234**	**1.823**	**4.587**	**0.668**
vdo4	**5.985**	**19.155**	**5.458**	**10.826**	**5.209**
MVI_01	**1.681**	**3.156**	**1.790**	**3.709**	**1.649**
MVI_02	**0.600**	**0.765**	**0.545**	**1.155**	**0.424**
MVI_03	**0.991**	**1.648**	**1.221**	**2.013**	**0.946**
MVI_04	**2.021**	**3.292**	**1.901**	**3.516**	**1.766**
MVI_05	**0.524**	**0.760**	**0.852**	**1.299**	**0.476**
MVI_06.mpg	**1.430**	**2.402**	**1.395**	**3.342**	**1.350**
MVI_07	**0.637**	**0.997**	**0.716**	**1.380**	**0.554**
MVI_08	**2.795**	**4.952**	**3.407**	**5.386**	**2.722**
Cycle	**1.352**	**2.324**	**1.600**	**2.615**	**1.351**
visor_car	**10.540**	**8.200**	**11.762**	**22.133**	**9.836**
winterDriveway	**3.100**	**4.720**	**4.210**	**6.716**	**2.998**

CONCLUSION

Variance based proposed algorithm is tested on video database given in table-1 and also compared with some well-known algorithms on the basis of detection rate and computational timingrequirements for object detection. Table 2 shows detection rate of frame differencing, frame differencing with edge detection, mean shift, Gaussian Mixture Model and variance methods respectively. It can be concluded from these tables that frame differencing with edge detection method has lowest detection rate. Simple frame differencing with centroid has better detection rate than frame differencing with edge detection whereas mean shift method is having better result than both. Here we can conclude that proposed variance method is best amongst all these methods on the basis of detection rate. It has detection accuracy of almost 90.402% for human and 73.198% for non-human databases. Table 3 shows timing requirements to detect objects by all methods for full database. Time required by proposed variance based method is lowest and GMM is largest for both the databases. Looking at table-2and table-3, one can conclude that the proposed algorithm is superior in terms of accuracy and timing than the prevalent algorithms.

REFERENCES

[1] MuyunWengGuoce Huang Xinyu Da. A New Interframe Difference Algorithm for Moving Target Detection. 3rd International Congress on Image and Signal Processing, Yantai, 16-18 Oct. 2010: 285-298.

[2] ZHAN Chaohui, DUAN Xiaohui, XU Shuoyu, SONG Zheng, LUO Min. An Improved Moving Object Detection Algorithm Based on Frame Difference and Edge Detection. Fourth International Conference on Image and Graphics, Sichuan, 22-24 Aug. 2007: 519-523.

[3] Rafael C. Gonzalez, Richard E. Woods. Digital Image Processing. 2nd edition, Prentice Hall.

[4] Arnab Roy, Sanket Shinde, Kyoung-Don Kang. An Approach for Efficient Real Time Moving Object Detection. Proceedings of International Conference on Embedded Systems & Applications, July 12-15, 2010, Las Vegas.

[5] Anil K. Jain. Fundamentals-of-Digital-Image-Processing. 1st edition, Prentice Hall.

[6] K. Sureshraju, Vishal R. Satpute, Kishor D. Kulat, Avinash G. Keskar. Image Compression Using Wavelet Transform: Compression Ratio and PSNR Calculations. National Conference on Computer Science and Informatics, Madurai, 23-24 April 2012: 195 -199.

[7] Yuanyuan Jia, Wei Qu. Real-time integrated multi-object detection and tracking in video sequences using detection and mean shift based particle filters. IEEE 2nd Symposium on Web Society, 16-17 Aug. 2010, Beijing: 738–743

[8] Ganesh Rakate. Human body tracking system based on DWT and mean shift algorithm on ARM-Linux platform. World Congress on Information and Communication Technologies, Trivandrum, Oct. 30-Nov. 2 2012: 715-720.

[9] Jerome M. Shapiro. Embedded Image Coding Using Zero Trees of Wavelet Coefficients. IEEE Transactions on signal processing VOL. 41 No.12 December 1993.

[10] Xiaoshi Zheng, Yanling Zhao, Na li, Huimin Wu. An Automatic Moving Object Detection Algorithm for Video Surveillance Applications. International Conferences on Embedded Software and Systems, Zhejiang, 25-27 May 2009: 541-543.

[11] Donald B. Percival. A Wavelet Perspective on the Allan Variance. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, Vol 63, Issue 4: 538 – 554.

[12] Vishal R. Satpute, Kishor D. Kulat, Avinash G. Keskar. A novel approach based on variance for local feature analysis of facial images. International Conference on Recent Advances in Intelligent Computational Systems, 22-24 Sep 2011: 210-215.

[13] Video Surviellance Online Repository (Home page from internert). R. Vezzani, R. Cucchiara, "Video Surveillance Online Repository (ViSOR): an integrated framework" in Multimedia Tools and Applications, vol. 50, n. 2, Kluwer Academic Press: 359-380, 2010.

[14] Available from: http://imagelab.ing.unimore.it/visor/

[15] AVSS 2007 International Conference on Advanced Video and Signal based Surveillance, London (U. K), 5-7 September 2007.

[16] Available from: http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html

[17] CAVIAR Test Case Scenarios. Video clips created July 11, 2003 and January 20, 2004.

[18] Available from: http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/