Comparison and Analysis of Moving Object Detection Techniques

Hitesh A. Patel
Lecturer, Computer Engineering Department,
B & B Institute of Technology, VV Nagar-388120, India.
hiteshpatelbbit@gmail.com

Paresh M. Tank
Lecturer, Computer Engineering Department,
B & B Institute of Technology, VV Nagar-388120, India.
pareshtank@gmail.com

***Abstract*— Systems for automated surveillance are essential for security applications. These systems are efficient at tracking, detecting, and classifying moving objects. The first and most important stage of a surveillance system is object detection. The precision of the detection phase has a significant impact on the surveillance system's overall performance. There are several methods for recognizing moving objects. This work presents a comparison of three object recognition techniques: Kernel Density Estimation [3], Approximate Median [7], and Temporal Frame Differencing [6]. CAVIAR [13] and PETS [14], two common surveillance video datasets, have been used to successfully evaluate these algorithms. The suggested techniques identify every moving object in films taken by stationary cameras positioned in moderately to highly complex both indoor and outdoor locations.**

***Keywords* — Visual surveillance, Temporal frame differencing, Approximate median, Kernel density estimation.**

I. Introduction

visual surveillance systems. Affordable, multi-sensor visual surveillance systems have been made possible by developments in processing power, large-capacity storage devices, and high-speed network infrastructure. Three main issues with these systems are the requirement for quick, dependable, and strong algorithms for tracking, identifying, and categorizing moving objects. Tracking items of interest across numerous frames while maintaining their correct identities is crucial to achieving this, as is first identifying and segmenting them from the backdrop [2]. Numerous disciplines, including as statistics, criminology, sociology, and traffic accident detection, and military applications, benefit from object detection algorithms [9].

The paper's second portion covers the several approaches that can be used to identify moving objects. The application of these detection methods is described in detail in the third section. The results of moving object detection are shown in the fourth section, which also assesses the three algorithms' time performance. The paper ends with some observations about the results and some directions for future research..

II. Related work

The main need for a visual surveillance system is the ability to identify moving objects in videos. The literature contains a variety of techniques for detecting moving objects, such as the following: Approximate Median (AM), Running Gaussian Average (RGA), Kernel Density Estimation (KDE), Temporal Frame Differencing (TFD), and Mixture of Gaussians (MoG).

A. Approximate Median

Image segmentation using image differencing is the first step of the Approximate Median approach, which was first presented by McFarlane and Schofield in [7]. To detect foreground pixels, the difference image is threshold after each subsequent frame is removed from a time-averaged reference image.

When it comes to fully separating things from the background, this technique works especially well. A background that adapts more slowly incorporates a longer visual scene history, yielding outcomes akin to buffering and processing multiple frames [5]. When compared to Temporal Frame Differencing, median filtering has proven to be highly robust, performs on par with more sophisticated techniques, and incurs just a slight increase in computational and storage expenses [12].

B. Running Gaussian Average

A background modelling method where each pixel position $(i,)$ is modelled independently was proposed by Wren et al. [11]. This technique fits the final n pixel values to a Gaussian probability density function (PDF). For every new frame at time t, a running (or online cumulative) average is calculated rather than recalculating the PDF from begin.

Following the identification of foreground pixels, small-sized regions are eliminated using morphological procedures like closing and opening. This method is susceptible to dynamic shifts, nevertheless, such as when abrupt changes in illumination or the appearance of previously obscured background regions are revealed by stationary objects [8].

$$\mu_t = \alpha I_t + (1 - \alpha)\mu_{t-1} \qquad (1)$$

In this case, I_t indicates the pixel value as of right now, while μ_t indicates the prior average. An experimentally determined weight that strikes a balance between responsiveness and stability in updates is the parameter α. The standard deviation σ_t of the Gaussian probability density function can be calculated similarly.

The running average method's effectiveness in terms of speed and memory utilization is one of its main advantages. The two parameters needed for each pixel are μ_t (mean) and σ_t (standard deviation), rather than a buffer holding the latest n pixel values.

At each frame time T, the pixel value I_t is classified as a foreground pixel if it satisfies a specific condition [8].

$$|I_t - \mu_t| > K\,\sigma_t \qquad (2)$$

C. *Kernel Density Estimation*

A non-parametric background subtraction method for density estimation is Kernel Density Estimation (KDE). This method creates a smooth approximation of the backdrop by averaging a known kernel density function over observed data points. The literature has examined a number of kernel functions with various characteristics; nevertheless, because of its continuity, differentiability, and location qualities, the Gaussian kernel is the most widely utilized.

In order to handle situations when the background is not completely static but contains little motions, such swaying tree branches, shifting vegetation, or subtle illumination changes, Elgammal [3] used KDE. Furthermore, spurious detections brought on by tiny camera displacements are successfully suppressed by this technique.

D. *Temporal Frame Differencing*

Lipton [6] developed the Temporal Frame Differencing approach, which uses the difference between two or three successive frames in a video clip to identify moving objects. This method works well in dynamic contexts and is computationally efficient.

Temporal Frame Differencing is limited in its ability to capture all crucial feature pixels, though. The fact that pixels with uniform intensity within an object might not be identified as "moving" pixels is a significant disadvantage [1]. Furthermore, items that stay still for extended periods of time cannot be detected using this method.

E. *Mixture of Gaussian*

A new adaptive online background mixture model was introduced by Stauffer and Grimson [10] and is capable of handling lighting variations, repetitive motions, clutter, adding or removing objects from the scene, and slowly moving objects with resilience.

Each pixel's values are determined using a combination of Gaussians in this method. Three to five Gaussians are typically employed. A combination of K Gaussian distributions is used to simulate the recent history of each pixel in the frame, {X1…….Xt}, and the values of a single pixel (e.g., scalars for gray values or vectors for color pictures) over time is referred to as a "pixel process." The following formula provides the likelihood of seeing the current pixel value [10].

$$P(X_t) = \sum_{i=1}^{k} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \qquad (3)$$

where $\mu_{i,t}$ is the mean value of $G_{i,t}$ and $\sum_{i,t}$ is the covariance matrix of $G_{i,t}$ and η is a Gaussian probability density function derived from equation 3, and $\omega_{i,t}$ is an estimate of the weight (what percentage of the data is accounted for this Gaussian) of the ith Gaussian ($G_{i,t}$) in the mixture at time t. K depends on the amount of memory and processing power that are available. At the moment, three to five are utilized. Additionally, the covariance matrix is presumed to have the following shape for computational purposes:

$$\Sigma_{k,t} = \sigma_k^2\, I \qquad (4)$$

A criterion that distinguishes between the foreground and background distributions is necessary for equation 3 to become a model of the background alone. This is how it is presented in [10]: first, the ratio of each distribution's peak amplitude (ω_i) to standard deviation (σ_i) is used to rank them. It is assumed that a distribution is more likely to belong to the background if it is higher and more compact. Then, as background, the first B distributions in ranking order that meet a given criterion are allowed [8].

$$\sum_{i=1}^{B} \omega_i > T \qquad (5)$$

III. Implementation

Distinguishing foreground items from immobile, non-moving background objects is the primary function of visual surveillance systems. Therefore, identifying moving objects in the films is the initial stage in a visual surveillance system. The identified items can be categorized into many groups, including vehicles, people, and more, and their movements can be monitored. Partial occlusions, shifting illumination, rapidly moving objects, background clutter, shadows, camera movement, and other issues are some of the difficulties that must be considered while putting object identification systems into practice.

Here, three techniques—approximate median, kernel density estimation, and temporal frame differencing—have been used to create the object detection module.

A method for temporal frame differencing has been proposed by Lipton et al. [6]. The frame at t-1 time has been regarded as the backdrop frame in this procedure. It has been computed how much the current frame differs from the background frame. A pixel is classified as foreground if the calculated absolute difference is higher than the threshold value; otherwise, it is classified as background. The two-frame differencing equation is provided below.

$$|I_t(x,y) - I_{t-1}(x,y)| > \tau \qquad (6)\ (3.1)$$

Foreground pixels are defined as those that satisfy equation 6.

The approximate median is the second approach that has been used. McFarlane and Schofield first proposed the technique in [7]. The way the approximate median approach operates is that the background pixel is increased by 1 if a pixel in the current frame has a value greater than the matching background pixel. Similarly, the background is decremented by one if the current pixel is smaller than the background pixel. In this manner, the backdrop gradually approaches an estimate in which half of the input pixels are larger than the background and the other half are smaller. This is how the background is computed[12].

The absolute difference between the current frame value of a pixel and the background frame value has finally been determined. Pixels are classified as foreground if the calculated difference is greater than the threshold value; otherwise, they are classified as background pixels. For each frame, this process has been repeated.

$$|I_t(x,y) - B_t(x,y)| > \tau \qquad (7)$$

The third approach that has been used is kernel density estimation, which was suggested by David Harwood and Ahmed Elgammal. Davis, Larry S. in [3].

Elmammal [3] claims that the model's fundamental feature for simulating the background is pixel intensity, or color. In order to estimate the density function of the pixel intensity distribution, the model maintains a sample of intensity values for every pixel in the image. As a result, the model can calculate the likelihood of any recently reported intensity value. The model is capable of handling scenarios in which the scene's background is cluttered and not entirely still, but rather exhibits slight movements brought on by shifting shrubs and tree branches. Because the model is updated often, it may adjust to changes in the scene's background [3].

The kernel density estimation methodology is a specific nonparametric method that estimates the underlying density, does not require storing all of the data, and is very general.
The underlying pdf is estimated using this method as [3].

$$f(x) = \sum_i \alpha_i K(x - x_i) \qquad (8)$$

Let $x_1, x_2, \ldots\ldots, x_n$ be a sample of a pixel's intensity values. With this sample, kernel density estimation may be used to estimate the pixel intensity at any intensity value. The likelihood of this observation is calculated as follows, given the observed intensity x_t at time t [3].

$$P_r(x_t) = \frac{1}{N} \sum_{i=1}^{N} K_\sigma (X_t - X_i) \qquad (9)$$

Where K_σ is a kernel function with bandwidth $\boldsymbol{\sigma}$. Kernel products can be used to generalize this estimate to color features [3].

$$P_r(x_t) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{d} K_{\sigma j} (X_{tj} - X_{ij}) \qquad (10)$$

The amount of memory and processing power available determines K. where $K_{\sigma j}$ is a kernel function with bandwidth σ_j in the *j*th color space dimension and x_t is a d-dimensional color characteristic. The density can be approximated as follows if the kernel function K is selected to be Gaussian [3].

$$P_r(x_t) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(x_{tj} - x_{ij})^2}{2\sigma_j^2}} \qquad (11)$$

According to this probability estimate, a pixel is deemed to be in the foreground if $Pr\ (x_t) < th$, where th is a global threshold that can be changed across all images to produce the appropriate percentage of false positives. The estimate in this case is predicated on the most current N samples that were utilized in the calculation. As a result, the model can be easily adapted by excluding older samples and including new ones [3].

Selecting an appropriate kernel bandwidth (size) is a significant problem that must be resolved when applying the kernel density estimation technique. Theoretically, the estimate will get closer to the actual density as the number of samples approaches infinity, making the bandwidth selection negligible. Practically speaking, selecting an appropriate bandwidth is crucial since the computation must be completed in real time and only a limited number of samples are used. An overly smoothed density estimate will result from a bandwidth that is too large, whereas a ragged density estimate will result from a bandwidth that is too little. Each pixel has a variable kernel bandwidth because the expected changes in pixel intensity over time vary depending on where in the image it occurs.Different kernel bandwidth is used for each color channel[3].

A pixel's kernel bandwidth σj^2 for the jth color channel can be estimated by computing the median absolute deviation over the sample for the pixel's successive intensity values. In other words, each color channel's median m of $|x_i - x_{i+1}|$ is determined separately for every subsequent pair $(\, x_i\, ,\ x_{i+1}\,)$ in the sample. Because different objects (such as the sky, branches, leaves, and mixtures when an edge passes over the pixel) are projected onto the same pixel at different moments, it is expected that pixel intensities throughout time will have jumps. This is the reason for using the median of absolute deviation. The pair $(\, x_i\, ,\ x_{i+1}\,)$ often originates from the same local-in-time distribution because variances between two consecutive intensity values have been detected; only a small number of pairs are anticipated to originate from cross distributions (intensity jumps). A few jumps shouldn't have an impact on the median, which is a reliable estimate [3].

The distribution for the deviation $(\, x_i\, ,\ x_{i+1}\,)$ is also Gaussian if the local-in-time distribution is Gaussian $N(\mu,\ \sigma^2\,)$, The quarter percentile of the deviation distribution is equal to the median of the absolute deviations because this distribution is symmetric [3]. That is provided as follows:

$$Pr(N(0,\ 2\sigma^2\,) > m) = 0.25 \qquad (12)$$

and therefore the standard deviation of the first distribution can be estimated as

$$\sigma = \frac{m}{0.68\sqrt{2}} \qquad (13)$$

In order to achieve more accurate median values, linear interpolation is utilized because the variances are integer gray scale (color) values [3].

IV. Results

The algorithms have been successfully implemented on the PETS [14] and CAVIAR [13] databases' standard surveillance footage. The algorithms can identify moving objects in both indoor and outdoor settings, according to the results.

OneStopMoveEnter1front.avi, an indoor film from the CAVIAR dataset, is utilized for testing. The footage is cut down to just 29 seconds. Tests are also conducted using outside video and the camera1.avi file from the PETS dataset. 320 x 240 resolution and 24 seconds have been substituted for the original video's 786 x 576 resolution and 112 seconds. Every fifth video frame is taken into account for processing.

Results for the PETS[14] dataset video camera1.avi are displayed in Figure 1 second column. The background of this outdoor video is complicated. The original video frames are displayed in the first row. The frames show people in motion, and trees that are waving. The temporal frame differencing method's results are displayed in the second row. All moving objects are detected using this method. A waving tree and some fake noise have been identified as moving objects. The set of "moving" pixels in frames 301 does not contain pixels that are uniformly bright inside a human. The approximate median findings are displayed in the third row. As the findings show, this approach produces qualitatively better results than the temporal frame differencing method. The kernel density estimate method's results are displayed in the fourth row. According to the findings, KDE's detection quality is inferior to that of the other two techniques. Some objects have not been fully detected, as may be seen visually. The reason for this is that the item is smaller than the chosen blob size. Additionally, some spurious noise has been found.

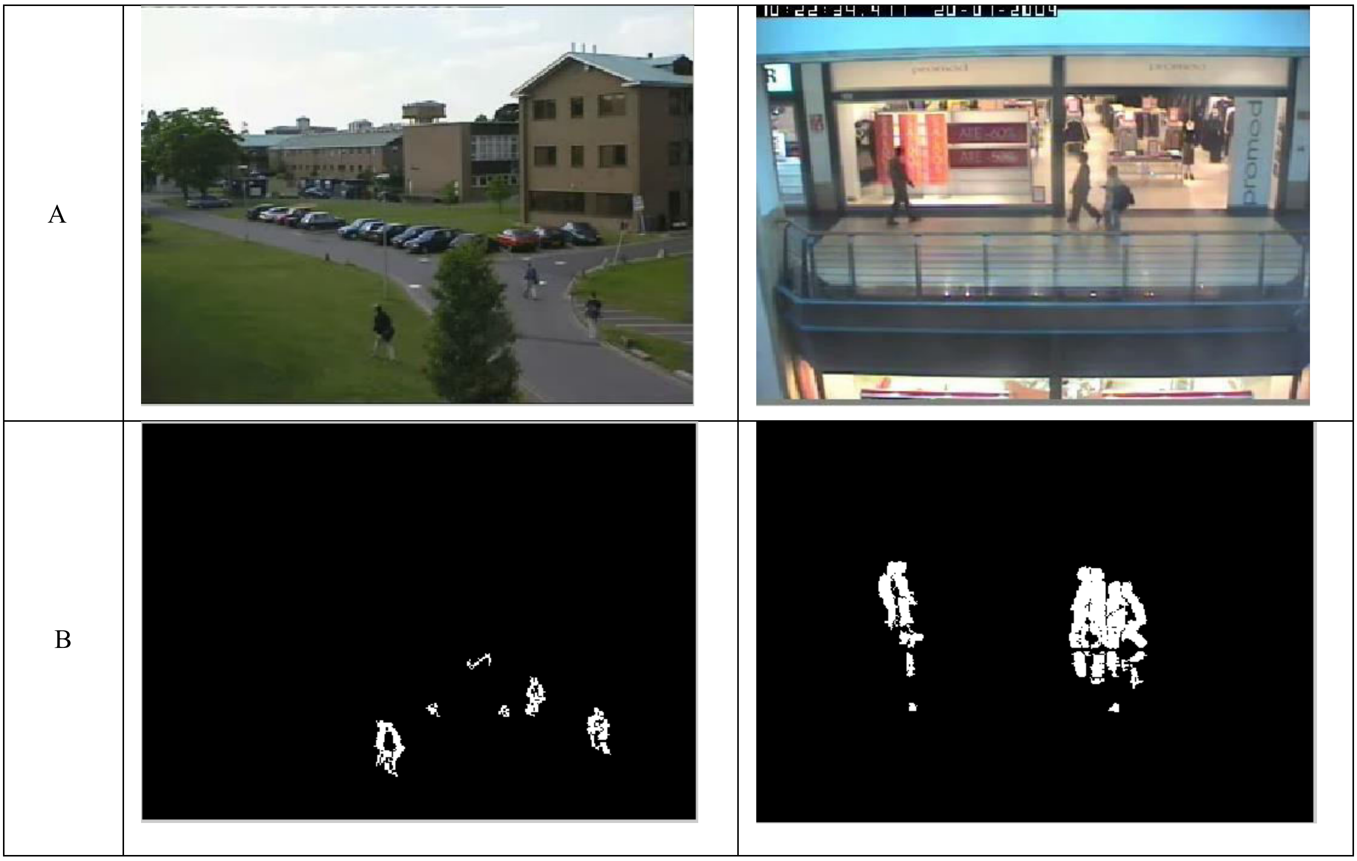
Results for the CAVIAR[13] dataset video OneStopMoveEnter1front.avi are displayed in Figure 1 third column. There are several people leaving and coming in at the same time in this indoor movie with a sizable hallway. The original video frames are displayed in the first row. The temporal frame differencing method's results are displayed in the second row. All people moving throughout the store are detected by this technology. The set of "moving" pixels does not contain pixels that are uniformly intense within a human. A portion of the human shadow was identified as a moving entity. The approximate median findings are displayed in the third row. Approximate median provides qualitatively good object detection results, as seen in outdoor video. The kernel density estimate method's results are displayed in the fourth row. As It can be seen from the findings, this approach produces better outcomes indoors than outdoors. Human shadows are detected by all three methods and must be suppressed from the output.

Time analysis for moving object detection techniques for movies from the CAVIAR[13] and PETS[14] datasets is displayed in Table 1. To distinguish foreground pixels from background, various threshold values have been chosen. To eliminate the output's misleading noise, several blob sizes have been chosen. The time required to process the entire video was represented by the total processing time. The number of frames handled in a second for various approaches is displayed in the last row. Time study shows that among the implemented methods, kernel density estimation is the slowest and temporal frame differencing is the fastest in terms of computing time.

Parameter	**Caviar Dataset Video OneStopMoveEnter1front.avi**			**Pets Dataset Video Camera1.avi**		
	TFD	***AM***	***KDE***	***TFD***	***AM***	***KDE***
Threshold	10	30	$1*10^{-9}$	10	15	$1*10^{-12}$
Blob size	20	10	100	20	10	100
Total processing time in seconds	5.89	9.45	22.55	3.47	4.83	11.98
No of frames processed per second	28.98	18.48	7.77	37.46	26.97	10.12

TABLE I.

EXECUTION TIME ANALSYS FOR MOVING OBJECT DETECTION METHODS



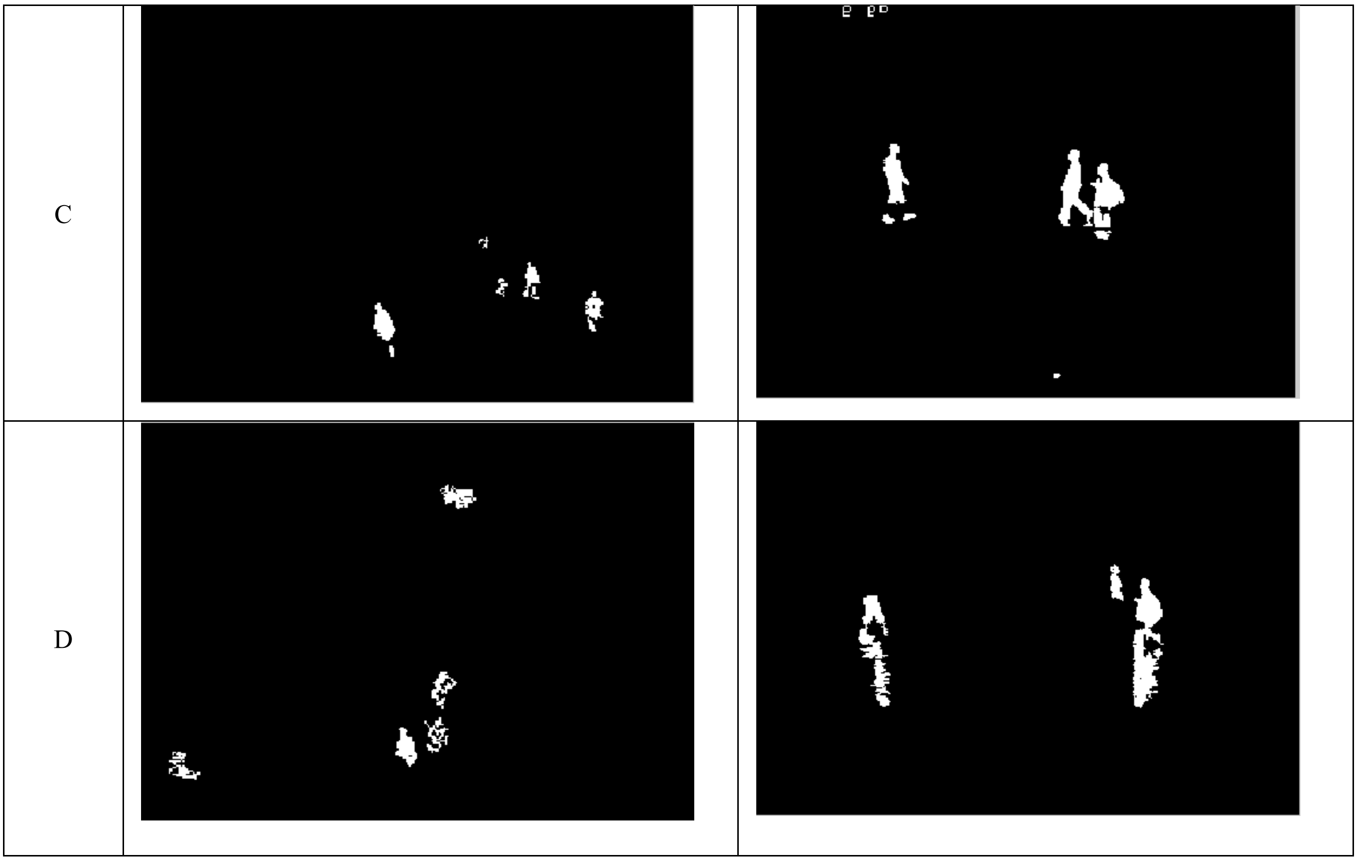


Fig 1. Original frames PETS [14] dataset sample (outdoor video) Original frames CAVIAR [13] dataset sample (indoor video)[[B] Output of temporal frame differencing [C]Output of approximate median [D] Output of kernel density estimation.

V. Conclusion

This study presents a visual surveillance system that can detect moving objects. Object detection using temporal frame differencing, approximate median, and kernel density estimation techniques has been effectively applied to the standard surveillance datasets of PETS[14] and CAVIAR[13]. Videos captured by still cameras are used by the system. The device is capable of detecting things both indoors and in indoor environments with fluctuating lighting levels and complex background. There has been much discussion on the current state of the art in modern work. The temporal frame differencing approach is the fastest of the implemented methods, according to the results. It is unable to identify moving pixels that are inside an object with uniform intensity. According to implementation, the approach with the slowest computing time is kernel density estimation. Although it can be improved, object detection using KDE in outdoor environments yields qualitatively poor results. Both kernel density estimation and temporal frame differencing identify an object's shadow as a moving component. The output results show that, out of all the developed methods, the approximate median is providing the best object detection results. The processing time of the approximate median is less than that of the kernel density estimation approach and more than that of temporal frame differencing.

VI. Future work

In order to achieve better outcomes, this work will continue and all implemented methods will be further improved.

Additionally, after moving objects have been detected, they can be categorized into any class for which the system is designed, including people, animals, automobiles, and more. Depending on the system's needs, the identified object can also be tracked. The typical color-based techniques found in the literature can also be used to execute the shadow reduction portion.

Applications in areas such as security, human-computer interaction, scene analysis and activity recognition, event detection, etc., can make use of the entire system with its detection, categorization, and tracking capabilities.

REFERENCES

[1] Collins R.T. et.al. ," A system for video surveillance and monitoring: VSAM final report". Technical report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, May 2000.

[2] Dedeoglu, Yigithan, " Moving object detection, tracking and classification for smart video surveillance." Diss. bilkent university, 2004.

[3] Elgammal A, Harwood D, Davis L., Non-parametric model for background subtraction, in: *European* Conference on Computer Vision, Dublin, Ireland, June 2000.

[4] Elgammal A., Duraiswami ramani, Harwood D, Davis L., "Background and Foreground Modeling Using Nonparametric Kernel Density Estimation for Visual Surveillance", Proceedings of the IEEE, vol. 90, no. 7, july 2002.

[5] Kumari Deepa, Shamik Tiwari, Deepika Gupta, Raina," Analysis on Adaptive Moving Objects via Robot Vision Implementations by Detection Techniques", International Journal of Scientific & Engineering Research, Volume 3, Issue 4, April-2012

[6] A. J. Lipton, H. Fujiyoshi, and R.S. Patil. Moving target classification and tracking from real-time video. In Proc. of Workshop Applications of Computer Vision, pages 129–136, 1998.

[7] McFarlane N, Schofield C., "Segmentation and tracking of piglets in images", British Machine Vision and Applications, BMVA 1995, pages 187-193, 1995.

[8] Piccardi,M. "Background subtraction techniques: a review." Systems, Man and Cybernetics, 2004 IEEE International Conference, Vol 4, pp.3099, 2004.

[9] Prasad Kauleshwar, Sharma Richa and Wadhwani Deepika," A Review on Object Detection in Video Processing", International Journal of u- and e- Service, Science and Technology Vol. 5, No. 4, December, 2012.

[10] Stauffer C., Grimson W.E.L," Adaptive background mixture models for real-time tracking" in: Computer Vision and Pattern Recognition, Santa Barbara, CA, June 1998.

[11] Wren C., A. Azarhayejani, T. Darrell, and A.P. Pentland, "Pfinder: real-time tracking of the human body," IEEE Trans. on Patfern Anal. and Machine Infell., vol. 19, no. 7, pp. 780-785, 1997.

[12] www.eetimes.com/General/PrintView/4017685- United States.

[13] http://groups.inf.ed.ac.uk/vision/caviar/caviardata1/

[14] http://www.hitech-projects.com/euprojects/cantata/ datasets_cantata/ dataset.htm.