INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume3, Issue1)
Available online at: www.ijariit.com

Hybrid Algorithm for Color Video Object Detection using Particle Filters

Arun Kumar
hooda.arun@gmail.com
Sat Priya Institute of Engineering and Technology, ECE
Maharishi Dayanand University, Rohtak.

Abstract— Colour can provide effective graphic features for tracking non rigid objects in real-time. However the colour of an object can vary over time dependent on the illumination, the visual angle to handle these appearance change a colour based target model must be adapted during temporally stable image observation. The proposed method of this dissertation gives new observation likelihood model with dynamic parameter setting. Experiments show our proposed method is more accurate and more efficient than the traditional colour histogram based particle filter. Integration of colour distribution into particle filters and shows how these distributions can be adopted overtime. A particle filter tracks several hypotheses simultaneously and weight them according to their similarity to the target model. As similarity measures between two colour distributions the popular Bhattacharyya coefficient is applied. In order to update the target model to slowly varying image conditions, Frames where the object is occluded or too noisy must be discarded.

Keywords— Weighted Histogram, Particle Filter, Trajectory, Colour, PDF, Prediction

I. INTRODUCTION

Video surveillance activities are grouped into three types, namely, manual activities, semi-automatic activities or fully-automatic activities. Manual video surveillance involves the analysis of video content by a human operator. Semi-automatic systems involve the use of automated processes for video processing tasks like motion detection, after 2 significant motions is detected and send them for analysis by human expert [1-2]. Fully automatic systems accept only the video sequences taken from a scene as input, and perform all tasks involving motion detection, object tracking, abnormal event detection automatically, without any intervention from humans. The present scenarios heavily implement manual systems for surveillance. These systems depend heavily on human visual inspection and serve the following key purposes: They provide a human operator with a huge amount of image sequences for analysis. The human operator detects potential threats and records them as evidence for investigative purposes. The systems suffer from major issues and to overcome these limitations of traditional surveillance methods, many researches are being conducted to develop automated systems for the real-time monitoring of people, vehicles and other objects using computer vision and artificial intelligence community. In spite of heavy researches being conducted, the field has not reached the state of maturity and requires more innovative and enhanced solutions [4-6]. This research is one another work that designs and implements an automatic video surveillance system to help alert security officers and prevent illegal activities.

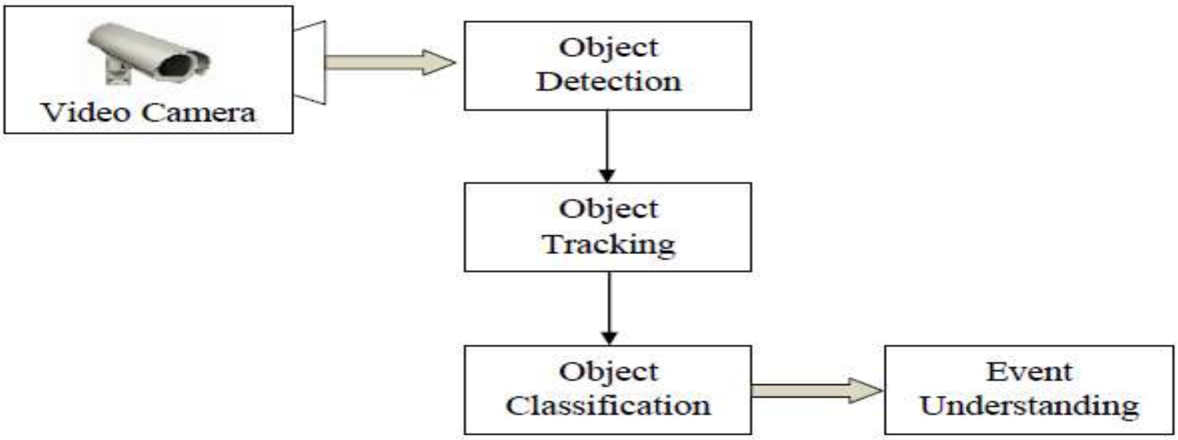

Fig1.1 General Architecture of AVS

The AVS are organized in a hierarchical fashion, with low level image processing techniques feeding to tracking and classification algorithms, which in turn, feed into higher level scene analysis and/or behaviour analysis modules The three subtasks, object detection, tracking and classification, converts the raw video image sequences into intelligent knowledge that can be used for event understanding [10-13]. The first step towards automated surveillance is detecting interesting objects in the video and this phase focuses on proposing solutions to this problem. The two main steps in object detection are object modelling and foreground extraction (background subtraction). Object detection is performed by building a representation of the scene called 'background model' and then finding deviations from the model for each input frame. Any significant change in an image region (foreground) from the background model is identified as moving objects. These pixels are marked for subsequent processing. This process is termed as background subtraction or foreground extraction.

II. LITERATURE SURVEY

Moving object detection plays an important role in video surveillance. Most high-level applications like abnormal event detection which is based on the success of moving object detection. Previously, motion object detection methods used object features like colour, intensity, edges and texture, which did not consider inter and intra frame relationships, which reduced the detection effectiveness [15]. To solve this problem, later researches moved towards motion features which increased the detection accuracy but comes with time complexity (Broojeni and Charkari ,2009). Video sequence images often require the need to differentiate between the objects in the image and their shadows. Shadow detection is a crucial task in many applications of video surveillance like scene understanding and interpretation, object segmentation, object recognition and tracking. This task is difficult because shadows have similar dynamics to the objects it is casted by.

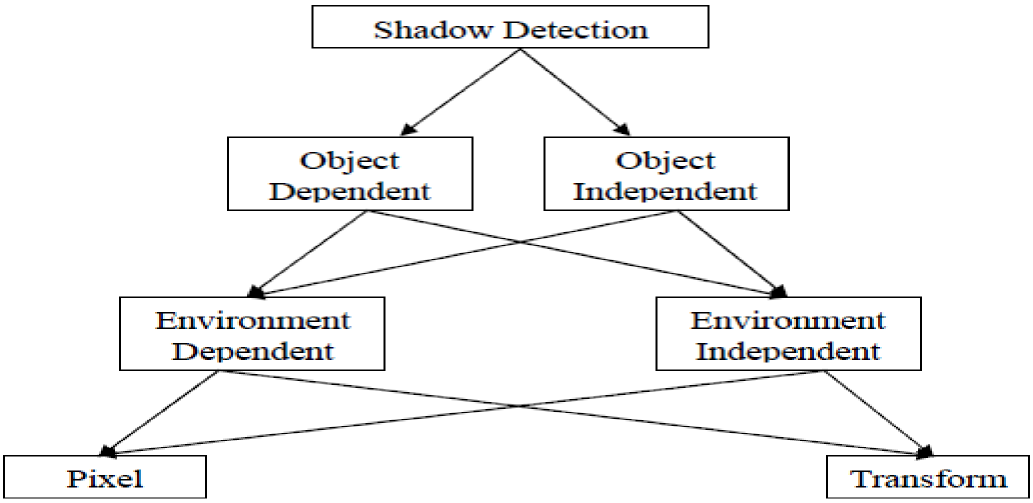

Fig2. Taxonomy of Shadow Detection Algorithms

Comparison of Object Detection Techniques:

Methods	Advantages	Disadvantages
Pixel Level Methods	Simple and has easy implementation steps	All pixel-based techniques can perform well when the foreground objects are moving, but are likely to fail when the time interval of exposure of the background is less than that of the foreground (Reddy *et al.*, 2011) and when foreground and background have similar structure and features.
Frame Differencing Methods	Strong adaptability for a range of dynamic environments	Shows errors in obtaining complete outline of moving objects. As a consequence, the detection accuracy is very low.
Hybrid Methods	High accuracy	Large computational complexity

III. METHODOLOGY/PLANNING OF WORK

The "filtering" problem is the process of estimating a system's current state which is hidden, based on past and present observations. This is represented by the probability density function $p(x_t|xt-1, z0{:}t)$. Depending on the application, the state at time t, x^t and observation at time t, z_t can be different entities. For the case of visual tracking, the state can be different like position, velocity, orientation, or scale of an object, and the observation can be the colour histogram or edge orientation histogram and may be contours of the image data [3,17].

In general, the object tracking problem can be modelled by the state-space depiction

State Equation:

$$X_t = f_t\,(x_{t-1}, v_t)\ (1) \quad \ldots\ldots\ldots\ldots\ldots..1$$

Measurement Equation:

$$Z_t = h_t\,(x_t, n_t)\ (2)\ldots\ldots\ldots\ldots\ldots\ldots\ldots2$$

Where f_t is the state equation (transition equation) at time t, and h_t is the measurement equation at time t. These are non-linear (NL) and time varying in general. The system noise and the measurement noise are denoted by v_t and n_t respectively.

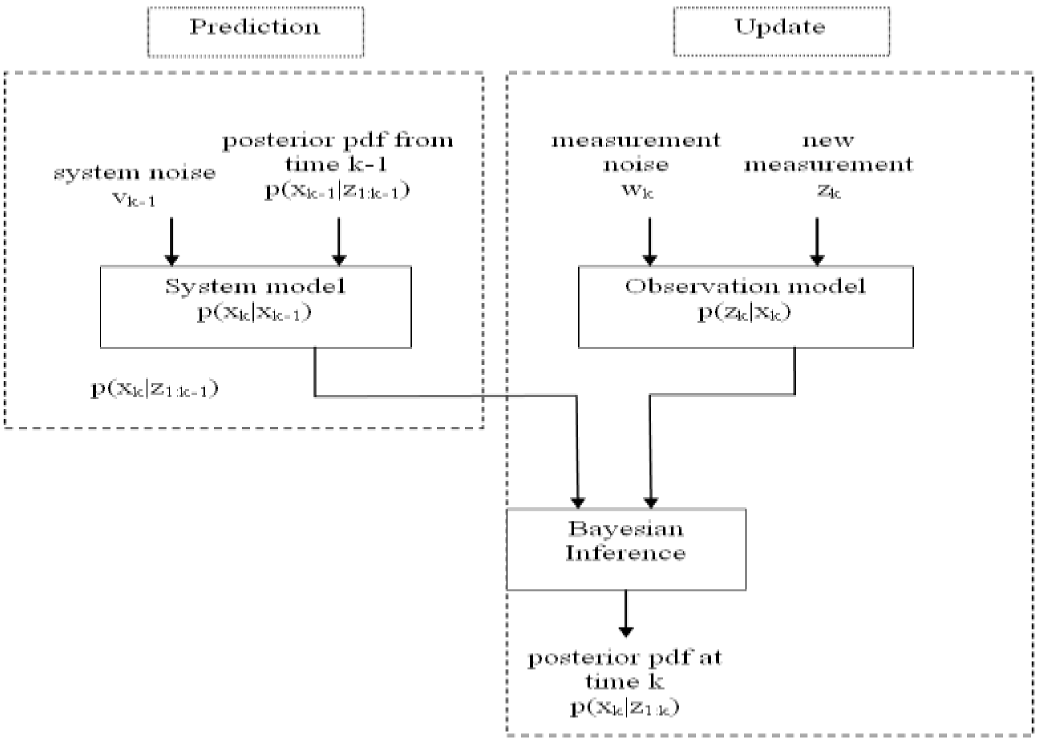

Fig 3. One iteration of the prediction and update. The goal is to find the posterior PDF at time k.

The key idea of particle filtering is to represent the posterior PDF by a set of discrete samples know as particles. A sample is also considered to as a particle due to of its discrete nature and its distinct representation by the probability density function (PDF). Each particle depicts a hypothesis of state and after that it is drawn randomly from the prior density. After a particle is drawn, it is then transmitted according to the transition model via free space or cable. Each propagated particle is verified by a weight assigning using likelihood model. The weight characterizes the quality of a specific particle. A large weight and small weight will be assigned to a good particle and a bad particle respectively.

The posterior PDF is constructed recursively by the set of weighted random samples {$x_t^{(i)}$, $w_t^{(i)}$ where i = 1 to N and N is the total number of particles. At each time t, the particle filtering algorithm repeats a 2 stage procedure: update and prediction

• **Prediction**: Each particle $x_t^{(i)}$ evolves independently according to the state model (1), which include addition of random noise in order to simulate the unknown disturbance. The step provides an approximation of the prior PDF:

$$p(x_t) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(x_t - x_t^{(i)})$$

• **Update:** Each particle's weight is evaluated based on the recent measurement in accordance with measurement model (likelihood model) (2). The posterior probability density function (PDF) at time t in the form of a discrete approximation which can be written as

$$p(x_t|z_{1:t}) \approx \sum_{i=1}^{N} w_t^{(i)} \delta(x_t - x_t^{(i)})$$

Where

$$w_t^{(i)} = \frac{\mathcal{L}(z_t|x_t^{(i)}) p(x_t^{(i)}|x_{t-1}^{(i)})}{q(x_t^{(i)}|x_{t-1,z_t})}$$

$$\sum_{i=1}^{N} w_t^{(i)} = 1.$$

Distance Measure A model histogram is the weighted colour or edge orientation histogram of the object to be tracked. The model histogram is constructed during the initialization (first frame at time t = 1) of the system. In subsequent frames, at every time t, there are N particles that represent N hypothetical states which need to be calculated. The observation likelihood model is used to assign a weight associated to a specific particle depending upon how exactness the object histogram q and the histogram $p(s_t)$ of the region described by the i^{th} particle s(i) t are.
To evaluate the similarity between the model histogram, q and the particle's histogram, $p(s_t^{(i)})$, where $s_t^{(i)}$ is the i^{th} particle at time t, we employ the Bhattacharyya coefficient ρ.

$$\rho[p(s_t), q] = \sum_{u=1}^{m} \sqrt{p_u(s_t) q_u}$$

Fig4.The pixels which are marked by a cross are used to build the histogram. Left: One half of the pixels are used and Right: One 4^{th} of the pixels are used.

IV. **RESULT**

Software MATLAB 2013b

It is powerful software that provides an environment for numerical computation as well as graphical display of outputs. In Matlab the data input is in the ASCII format as well as binary format. It is high-performance language for technical computing integrates computation, visualization, and programming in a simple way where problems and solutions are expressed in familiar mathematical notation.

- Acquisition, Analyzing &Visualization
- GUI (graphical user interface) building environment.
- Mathematical and Computational functions
- Engineering complex drawing and scientific graphics
- Modelling and simulating problems prototyping
- Analyzing of algorithmic designing

The proposed tracker employs the Bhattacharya distance to update the priori distribution calculated by the particle filter (PF). The target regions are represented by ellipses so that a sample as given as:

$$St = \{x, y, x' y' Hx, Hy\ Hx' Hy' y\}$$

Where x, y represents the location of the ellipse,x', y'. the motion Hx, Hy represent the length of the half axes and Hx', Hy' represents the corresponding scale changing.

Fig.5 Represents Ymean of the input voltage vertical axis represents the Y mean length of input video.

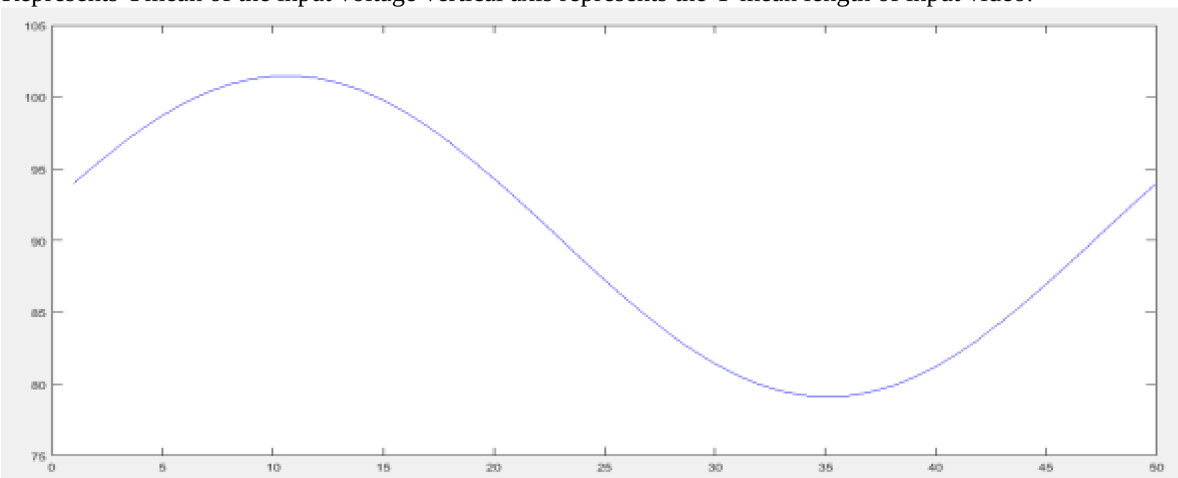

Fig.5 Y mean ellipse of video

Fig.6 Represents Xmean of the input video ellipse vertical axis represents the Y mean length of input video

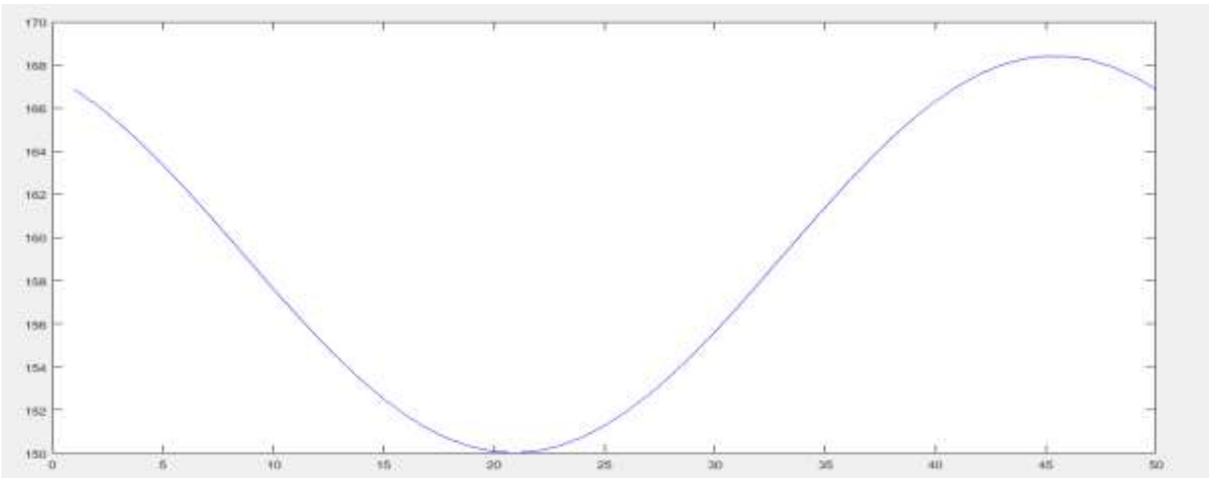

Fig.6 X Mean length of video

Fig.7 Represents Probability density function for colour ellipsoid in this we recognize colour tracking from colour ellipsoid based technique vertical axis represents

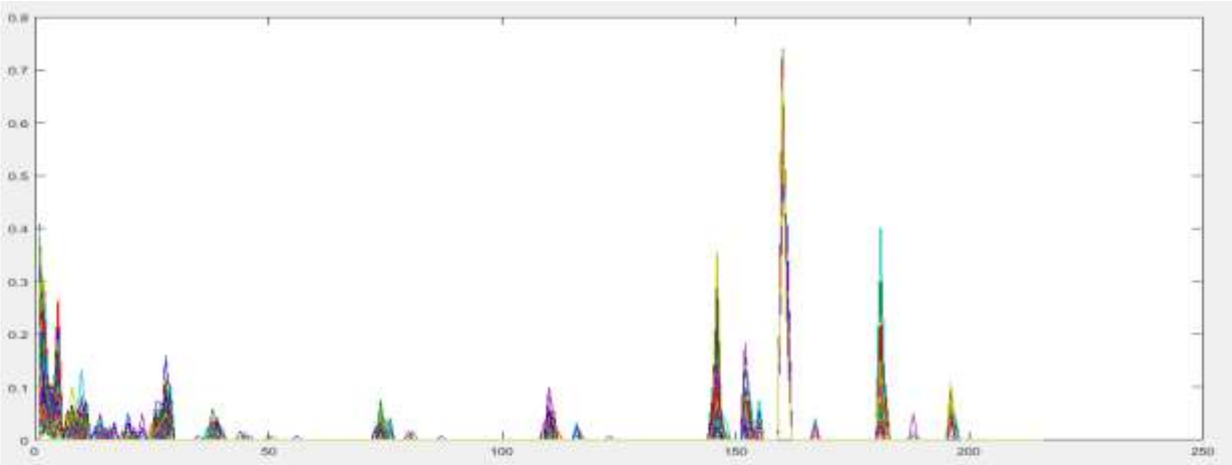

Fig.7 Probability density function colour ellipsoid

Fig.8 represents likeliyhood characterstics which represent the current likelihood model is based on color histograms.While this seems robust against deformations, rotationand partial occlusion of the target, it completely ignoresother features like the shape. It would be interestingto see how the filter would perform if the likelihoodwould be based on mutual information as defined by Violaand Wells instead. This would probably meanthat wewould loose much of the robustness, but in a controlled environmentwhere we have more guarantees about the shapeand color of the target it could prove useful.

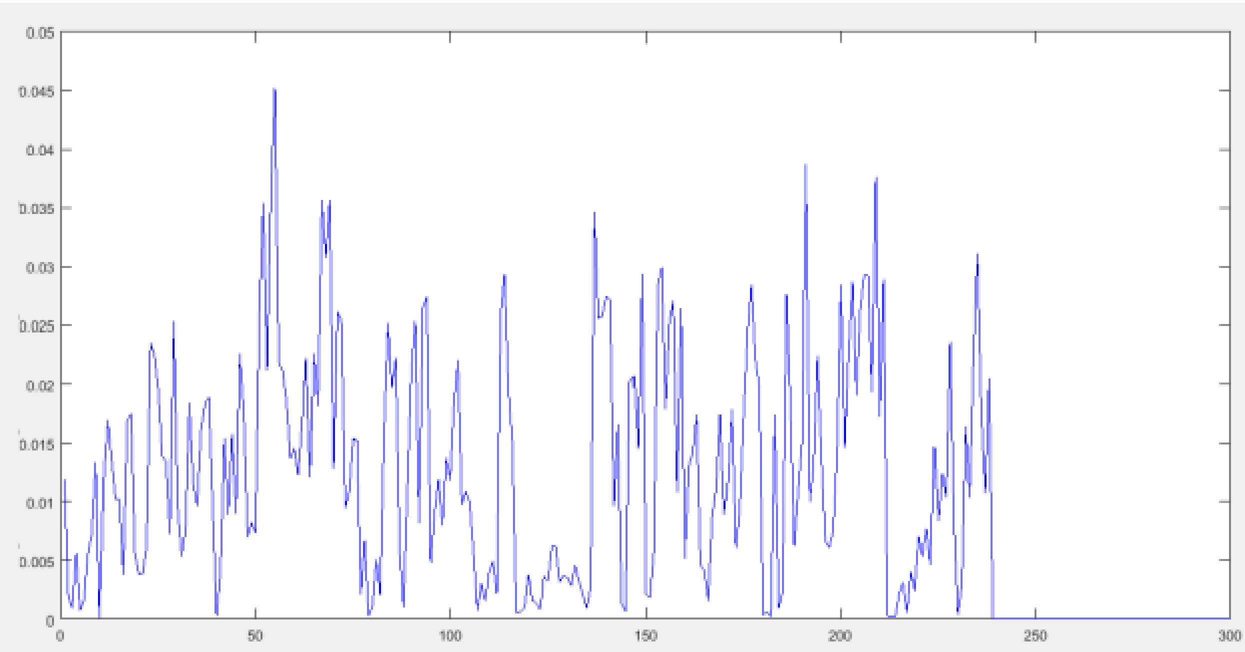

Fig.8 Likelihood characterstics which represent the current likelihood model is based on color histograms

Fig.9 represents the feather characterstics of color histogram based particle filter

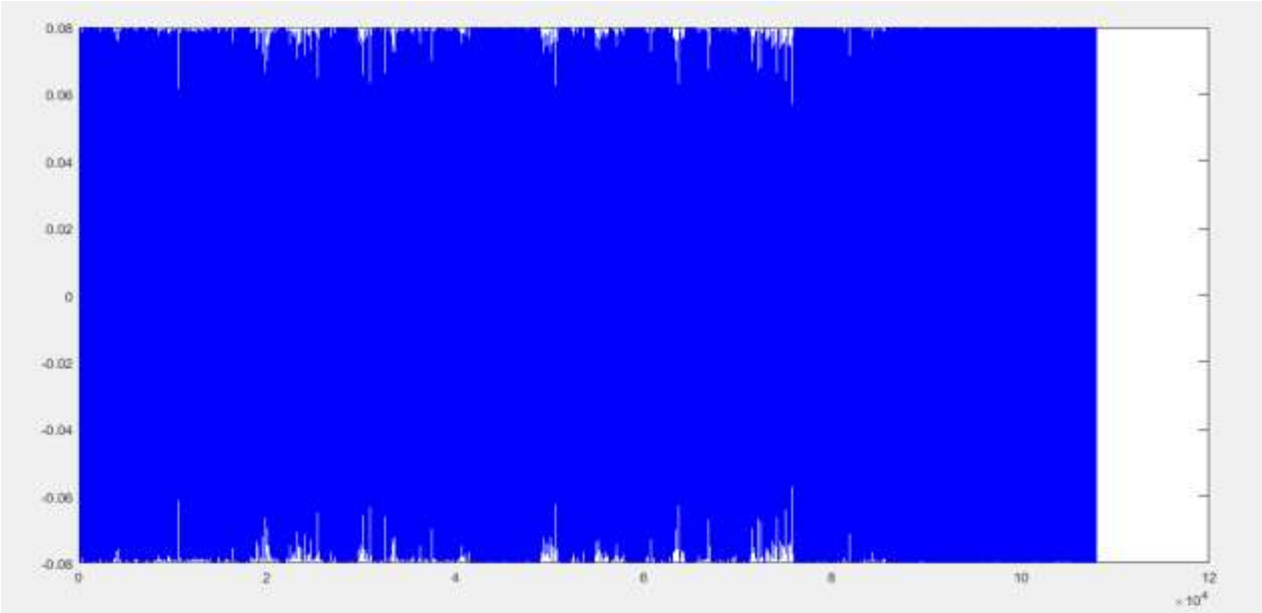

Fig.9 feather characteristics of input video

Fig.9 and Fig.10 both represents colour trajectory of moving helicopter vertical axes represents trajectory path on vertical axes and horizontal trajectory path represents x axis vertical dimension of video is 240 and horizontal dimension is 320 sizes. The trajectory path is 400 numbers of frames

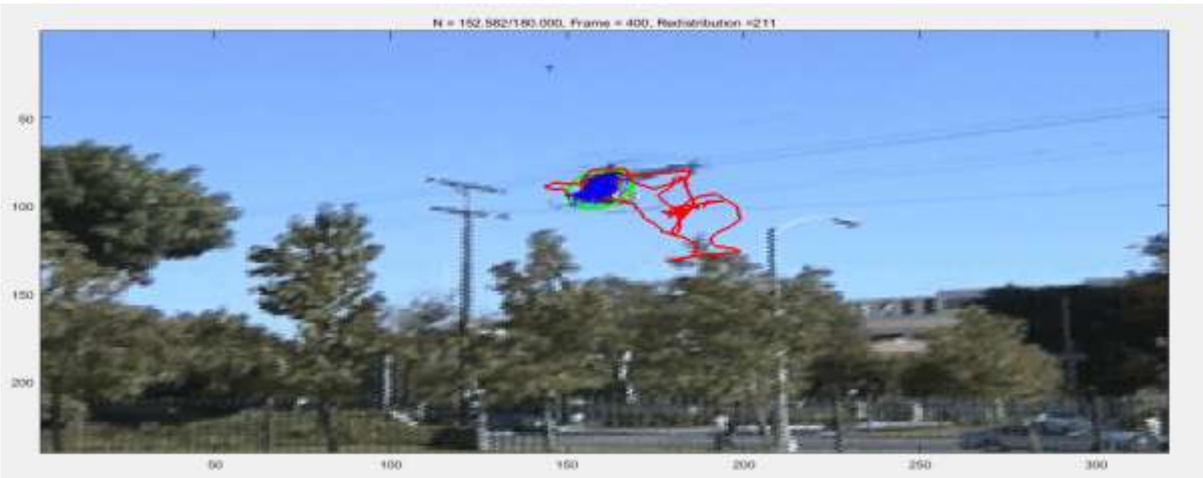

Fig.9 colour helicopter trajectory

Fig.4.7 represents the trajectory path which is doted by blue line and it represents the motion helicopter region and it represents that the object is 320*240 dimensions.

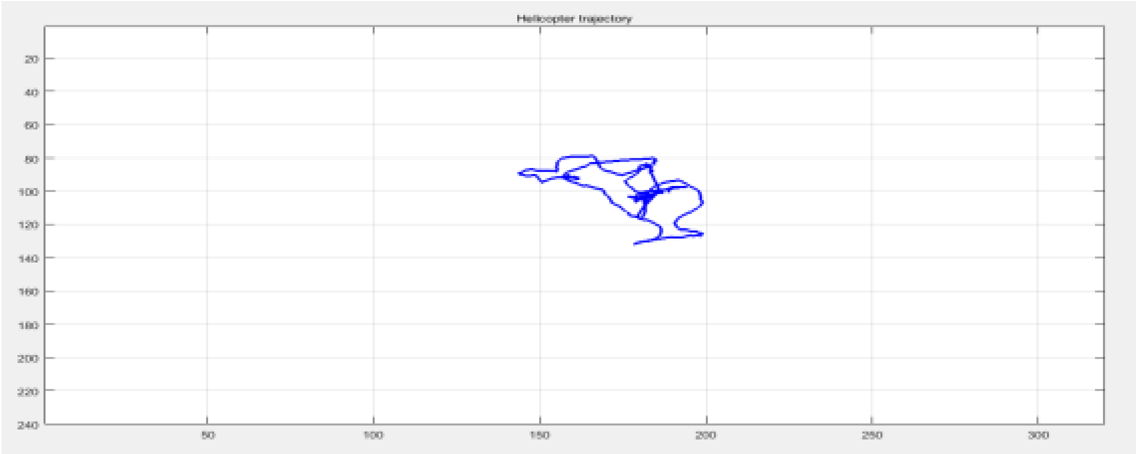

Fig10 Helicopter trajectory of input video

Fig.11 represents ellipse angle versus frames. Ellipse angle represents vertical axes and horizontal axes represent Frames. After 150 frame the value is confined to 0.5. A colour histogram constructed using all pixels in the ellipse is computed and is used as a control for comparison. Two histograms using only one half and one quarter of the pixels are constructed and are compared with the control histogram

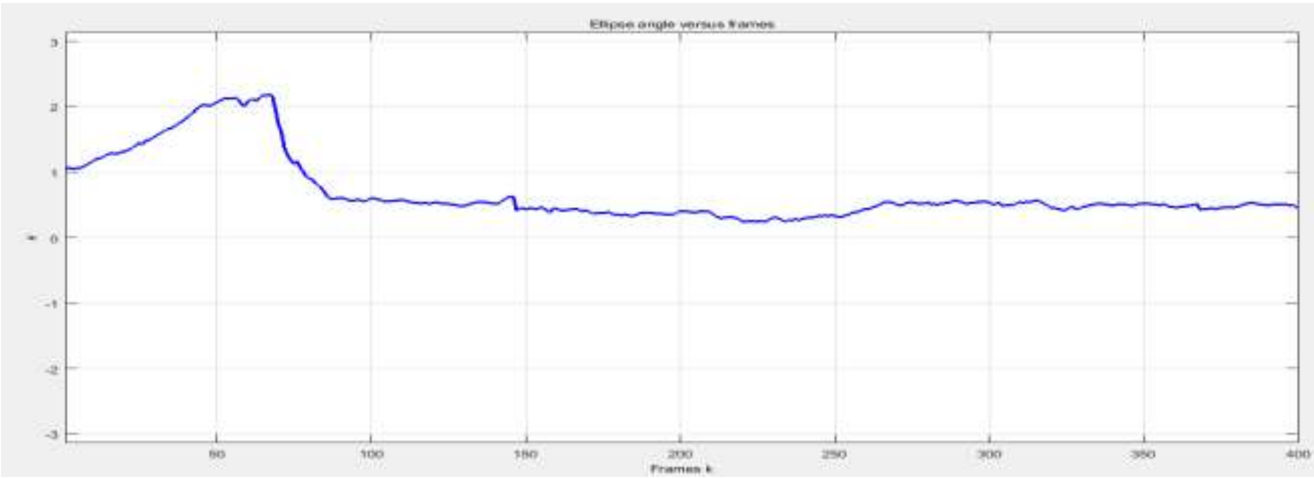

Fig.11 Ellipse angle versus Frames

Fig.12 represent colour feature of object vertical axis C1, C2, C3 are the histogram dimensions and horizontal dimension are vector colour in all edges. We model colour features or distinctive attribute of an elliptical region R, where R can be a region which surrounded the object to be tracked. A colour histogram is commonly used for object tracking because of their robustness to partial or incomplete occlusion and is rotation and also scale invariant. They are more flexible in the types of object so that they can be utilized to track, including rigid and non-rigid object. The bin index where the colour component at xi falls into, and δ is the Kronecker delta function (KDF). In our experiment, 8× 8 × 4-bin histogram is constructed or erected for every region R in HSV colour space. Hue saturation value color space is used to reduce the sensitivity to illumination

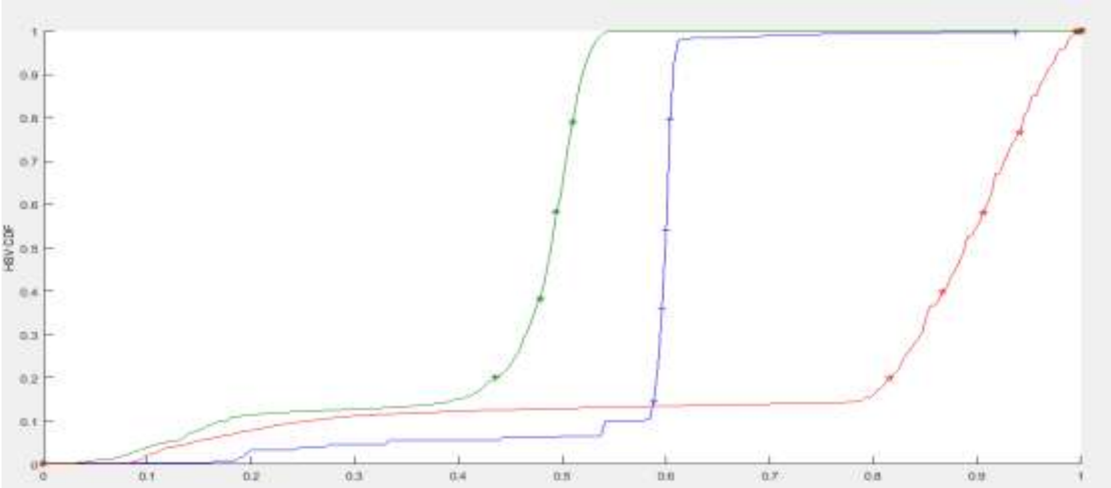

Fig.12 HSF/CDF Histogram

CONCLUSION

In this dissertation we introduced two new concepts for using particle filtering for tracking. An efficient algorithm for computation of the colour histograms was described and a new observation likelihood function was presented. The new approaches decreased the computational complexity by 50% when the checkerboard pattern was used for histogram construction. The method was more robust to cases such as occlusion and illumination change when multiple features and dynamic parameter setting were used. We have presented an algorithm based on PF's for detecting objects in video. The observation density required by the PF is based on a set of rectangular features selected by an ad boost procedure as introduced in [6]. The approach allows estimating the probability of face presence in the current frame given the history of all observed frames. This allows accumulating the likelihood of object presence over several frames. The resulting system is therefore less sensitive to false detections, while being able to detect faces in difficult cases. In future automatic initialization of reference window can be used to detect an object; similarly multi object tracking can be done.

REFERENCES

[1]. Y. Nakashima ,N. Babaguchi ,J. Fan, Detecting Intended human objects in Human captured videos. 978-1-4744-7030-3/10 IEEE.

[2]. Jack Czyz. Object detection in video via particle filtering proceeding of 18th International conference on pattern recognition (ICPR06)0-7695-0/06,IEEE.

[3]. B.Ristio, SArulampalam,N. Gordon. Beyond kalman filtering :Particle filter for tracking applications Arten house 2004.

[4]. M. Jaward ,L.Mihaylava ,Multiple Object tracking using particle filters ,IEEEAC paper #1280,IEEE2006

[5]. S.Fazli,H. moradi, H.Bouzari,Particle filter based object tracking with sift and color features ,2009 IEEE second international conference on machine vision.

[6]. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non- linear/non-gaussian bayesian tracking," IEEE Transactions of Signal Processing, 2002, pp. 174–188.

[7]. K. Nummiaro, E. Koller-Meier, and L. V. Gool, "Color features for tracking non-rigid objects," Special Issue on Visual Surveillance Chinese Journal of Automation, 2003, pp. 345–355.

[8]. D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," Computer Vision and Pattern Recognition, vol. 2, 2000, pp. 142–149.

[9]. P. P´erez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," Proceedings of the 7th European Conference on Computer Vision-Part I. London, UK: Springer-Verlag, 2002, pp. 661–675.

[10]. M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," International Journal of Computer Vision, vol. 29, no. 1, 1998, pp. 5–28,.

[11]. J. Lichtenauer, M. Reinders, and E. Hendriks, "Influence of the observation likelihood function on particle filtering performance in tracking applications," IEEE International Conference on Automatic Face and Gesture Recognition, 2005, pp. 767–772.

[12]. T. Xiong and C. Debrunner, "Stochastic car tracking with line- and color-based features," IEEE Transactions on Intelligent Transportation Systems, vol. 5, no. 4, Dec 2004, pp. 324–328.

[13]. H. Sugano and R. Miyamoto, "A real-time object recognition system on cell broadband engine," Proceedings of Pacific-Rim Symposium on Image and Video Technology, 2007, pp. 932–943.

[14]. Y. Rui and Y. Chen, "Better proposal distributions: object tracking using unscented particle filter," Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2001, pp. 786–793.

[15]. P. Brasnett, L. Mihaylova, D. Bull, and N. Canagarajah, "Sequential monte carlo tracking by fusing multiple cues in video sequences," Image and Vision Computing, vol. 25, no. 8, 2007, pp. 1217–1227.

[16]. C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," International Conference on Computer Vision, vol. 1, 2005, pp. 212–219.
[17]. Q. Wang and J. Liu, "Visual tracking using the kernel based particle filter and color distribution," Interna-tional Conference on Neural Networks and Brain, vol. 3, 2005, pp. 1730–1733.

