ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 5, Issue 6, June 2016

Particle Filter Based Object Tracking in Video

Swati S. Jadhav, Rohita P. Patil

***Abstract*—Object tracking has been emerging as a demanding research area in the domain of video surveillance. The accurate object tracking is achieved by overcoming difficulties caused by object deformation, occlusion and illumination variation. Object tracking using Particle filter algorithm could overcome these challenges so it becomes the recent area of research. The particle filter update the state-space dynamic model using approximation of posterior probability density function. It is achieved using the finite set of weighted samples known as particles. The object tracking process requires huge computation. The design is implemented on FPGA using synthesizable state machine. The states correspond to particle generation, likelihood estimation, resampling and particle update modules of the particle filter algorithm.**

Index Terms—*object tracking, computer vision, Particle filter,occlusion, deformation.*

I. INTRODUCTION

Object tracking has number of application in the domain of video surveillance, video indexing, vehicle navigation,human-computer interface,robot controland object classification. In general object tracking is process to create temporal correspondence among detected objects in consecutive frames. Object tracking process can be divided into number of tasks as Object representation, Feature Selection, Object Detection and Tracking.

A recursive Bayesian filter is statistical process of dynamic state estimation. In particle filtering using the state space approach different object properties are modelled. This results in posterior probability distribution function of the object state vector. Kalman filter and particle filter come under category of Bayesian filter. Kalman filter is best possible estimator for linear state space model where noises of the model are Guassian. Particle filter provides object state estimation even for non-linear state space model with non-Guassian noises. So the particle filter covers wide range of applications.

The particle filter method is based on three main operations: Sampling, Update and Resampling. The probability density function represented approximately as set of random-samples known as particles. The weights are assigned to the particles and get updated as per system dynamics.

Manuscript received June, 2016.

***Swati S. Jadhav**,Master of Engineeering Student, Dept. of Electronics and Telecommunication,Smt. KashibaiNavale College of Engineering, Vadgaon(BK),Pune,SavitribaiPhule Pune University, India.*

***RohitaP. Patil**, Professor,Dept. of Electronics and Telecommunication,Smt. KashibaiNavale College of Engineering, Vadgaon(BK),Pune,SavitribaiPhule Pune University, India.*

The proposed system consider object histogram as tracking feature. The random point are considered initially to prepare templates. These points are nothing but particles. This template is used as reference to update particle values in subsequent frames for tracking the object. The computation is based on likelihood function which is used to predict the location of object. During the process there might be the sample impoverishment problem which can be resolved by resampling algorithm.

This paper is organized as follow; section II describes related work. Section IIIprovides the mathematical analysis of particle filter algorithm. Section VI describes system architecture for object tracking using particle filter. Section V contains the FPGA implementation and simulation results. Section VI ends the paper.

II. RELATED WORK

Resampling is important step of particle filter algorithm as it prevents degeneracy of propagated particles. The resampling methods are reviewed in [1]. Evolutionary computing using genetic operators such as crossover and mutation addresses the problems of particle degeneracy and sample impoverishment [2]. In [3] the distance based histogram is calculated for RGB planes separately with flexibility for selecting object size, type of video sequence and number of particles.The problem of inaccurate tracking due to illumination variation is resolved in [4] with the help of local color entropy feature.

Efficient hardware implementation is achieved in [5] using Multiple Candidate Regeneration (MCR) algorithm; where MCR is recursive process analogous to prediction and update steps of particle filter algorithm.The parallel architecture for weight calculation and histogram computation is proposed in [6]. The design is based oncomputation of Bhattacharya coefficient for Region of Interest to estimate center of the target object.Hardware/software co-design has been proposed in [7].The software part contains computation of weight of particles andis implemented on NIOS-II, while hardware circuit is used for particle update step. Considering constraints on the resources like memory bandwidth and operation cycles a low cost real time object tracking system is achieved using dual cache architecture [8]. The robotic applications for object trackingand its FPGA implementation is proposed by [9]. The application includes construction automation using Bayer image patterns.

The applications of object tracking require real time response. The algorithms should be designed for efficient hardware implementation such that it should not require external memory module and applicable for different image sequences.

III. PARTICLE FILTER ALGORITHM

Particle filter is a Monte Carlo method used to compute a Bayesian state estimate recursively by updating a posterior pdf of the state of the state at each time *t* based on available information up to *t*. The relation betweenobservable state variableY_t, the unobserved or latent variable s_t and noise v_t is given by state transition equation as:

$$s_t = \gamma(s_{t-1}, Y_{t-1}, v_t) \qquad (1)$$

The associative transition density of (1) is $f(s_t|s_{t-1}, Y_{t-1})$.The measurement equation is,

$$y_t = \delta(s_{t-1}, Y_{t-1}, u_t) \qquad (2)$$

The associative density of (2) is $f(y_t|s_t, Y_{t-1})$ and u_t is measurement error or measurement noise. Consider the initial density$f(s_0)$. The filtering and likelihood evaluation proceed recursively. The steps involved in particle filter algorithm are:

Prediction:

$$f(s_t|Y_{t-1}) = \int f(s_t|s_{t-1}, Y_{t-1}), f(s_{t-1}|Y_{t-1})ds_{t-1} \qquad (3)$$

Forecasting:

$$f(y_t|Y_{t-1}) = \int f(y_t|s_{t-1}, Y_{t-1}), f(s_t|Y_{t-1})ds_t \qquad (4)$$

The likelihood estimation $\hat{f}(y_t|Y_{t-1})$ is derived from (4) as:

$$f(y_t|s_t^{1,i}, Y_{t-1}) = \frac{1}{\sqrt{(2\pi)\Sigma u}} \exp\frac{(X_t - x_t^{1,i})'\Sigma u(X_t - x_t^{1,i})}{2}$$

Averaging over particles yield the Likelihood function as:

$$\hat{f}(y_t|Y_{t-1}) = \frac{1}{N}\sum_{i=1}^{N} f(y_t|s_t^{1,i}, Y_{t-1}) \qquad (5)$$

The posterior weight w_t^i obtained from the prior weight w_{t-1}^i as:

$$w_t^{0,i} = \frac{f(y_t|s_t^{1,i}, Y_{t-1})}{\hat{f}(y_t|Y_{t-1})} \qquad (6)$$

Update:

$$f(s_t|Y_t) = \frac{f(y_t, s_t|Y_{t-1})}{f(y_t|Y_{t-1})} = \frac{f(y_t|s_t, Y_{t-1})f(s_t|Y_{t-1})}{f(y_t|Y_{t-1})} \qquad (7)$$

The update step is followed by likelihood step and the process repeats. The parameter Σu is standard deviation.

IV. OBJECT TRACKING SYSTEM

The system block diagram of object tracking is shown in Figure 1. The camera is used to input video signal to FPGA board. The video frames are stored in buffer. The resolution of image from video signal and number of particles are considered for furthercomputation.

A. Particle Generation

The operation of particle generation is the initialization step. It is applicable for very first frame of the video sequence. The pixels are randomly selected from the image, which is termed as particles. The state space model is built from the location and probable direction of object's motion. More the number of particles more will be accuracy of object detection. But the number of particles are constrained with resources available for computation and time required for computation. For this FPGA implementation total sixteen particles are used. The random generator module is used to generate particles taking resolution into account.

B. Particle Propagation

The particles are propagated according to motion model.It is assumed that if the object is moving towards the right direction then it will be move in right in subsequent frames. Similarly if object is moving towards left direction then it will move in left in subsequent frames.

C. Likelihood Estimation

The likelihood function is used for estimation of new values from the prior data. The likelihood function finds the difference between current positions of object with respect to the reference location. The computation is performed for all the particles and used for updating the particle location.

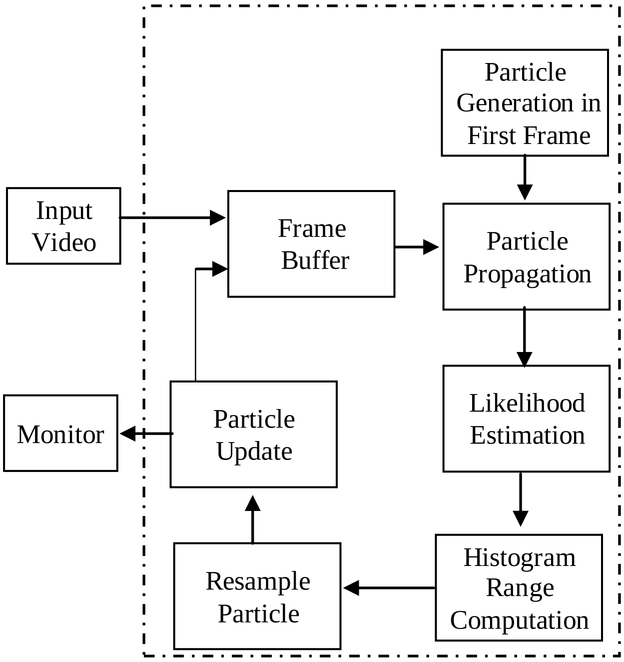

Fig. 1. System block diagram

D. Particle Resampling

The resampling step replace particles with low likelihood by particles with high likelihood. This need to be done since we have finite number of particles. Thus particles with high weights gets replicated to indicate interested area.

E. Histogram Range Computation

The cumulative sum of normalized weights is used to define the range of histogram. The histogram indexing is performed to update the state vector.

F. Particle Update

The particles gets updated through the filtering process. The particles which gets accumulated at the location of object are considered for further computation discarding other particles. Thus the particles with low weight filtered out retaining only significant particles.

V. FPGA IMPLEMENTATION AND SIMULATION RESULTS PARTICLE FILTER ALGORITHM

The state diagram of particle filter is shown in Figure 2.The *state 0* of finite state machine (FSM) corresponds to video frame buffer. The Particle generation create state space matrix representation with respect to 16 randomly generated particles.

$$S = \begin{bmatrix} X_1\, X_2\, X_3 \ldots X_{14}\, X_{15}\, X_{16} \\ Y_1\, Y_2\, Y_3 \ldots Y_{14}\, Y_{15}\, Y_{16} \\ h_1\, h_2\, h_3 \ldots h_{14}\, h_{15}\, h_{16} \\ w_1\, w_2\, w_3 \ldots w_{14}\, w_{15}\, w_{16} \end{bmatrix}$$

The first and second row of matrix corresponds to location of particles i.e. (x, y) co-ordinates. The third and fourth row of matrix indicates the motion. These particles are initialized for first video frame. In this matrix third and fourth row contains all zero elements that implies steady position of object. This corresponds to *State 1* of FSM.

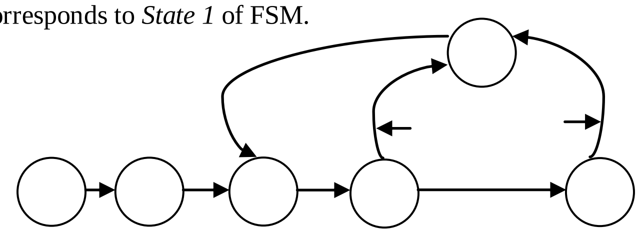

Fig. 2. Finite state Machine for particle filter.

These particles are propagated using random generation algorithm. The co-ordinates of particles are generated as new center around the old center. In this case the third and fourth row contains nonzero elements which indicates dynamic model of the object to be tracked. This process is *state 2* of FSM.

$$X_{new} = X_{old} + w_{old} \pm random$$
$$Y_{new} = Y_{old} + h_{old} \pm random$$
$$h_{new} = h_{old} \pm random$$
$$w_{new} = w_{old} \pm random$$

The random numbers are generated by concatenating the bits of eight bit counter in random manner. The likelihood computation is based on logarithmic function and it is *state 3* of FSM. Here *D* is the difference between the pixel value i.e. red, green and blue components and the reference initialized value.

$$Dist = D' * D$$
$$Dist = R^2 + G^2 + B^2$$

The likelihood function is

$$L(k) = A + B * Dist$$

Where A and B are constant and represented as:

$$A = \log\left(\frac{1}{\sqrt{2\pi\Sigma u}}\right)$$
$$B = -\frac{1}{2 * (\Sigma u)^2}$$

Here Σu indicate the standard deviation. The above computation is performed on all particles and the likelihood value is stored in the form of array in vector *L(K).*

Particle resampling is performed using cumulative sum algorithm and it is the *state 4* of FSM.

$$w = e^{(L - \max(L))}$$
$$w_{normal} = \frac{w}{\sum_{i=1}^{16} w_i}$$

Perform cumulative sum of vector w_{normal} that corresponds to the range of histogram. The histogram is generated as *state 5*of the FSM. Here the histogram is indexed to create the range vector.

$$T = \left[0, \left(\frac{1}{16}\right), \left(\frac{2}{16}\right), \left(\frac{3}{16}\right), \dots\dots, 1\right]$$
$$= [0, 0.0625, 0.125, 0.1875, \dots, 1]$$

Just for illustration consider the vector of normalized weights as:

$$w_{normal} = [0.05, 0.06, 0.11, 0.19, 0.4, 0.55, \dots\dots]$$

Histogram indexing corresponds to above values is represented by *Histo* vector as:

$$Histo = [1,1,2,4,7,9, \dots\dots]$$

From above histogram the state vector gets updated as given below:

$$S = \begin{bmatrix} X_1\, X_1\, X_2\, X_4\, X_7\, X_9 \dots \\ Y_1\, Y_1\, Y_2\, Y_4\, Y_7\, Y_9 \dots \\ h_1\, h_1\, h_2\, h_4\, h_7\, h_9 \dots \\ w_1\, w_1\, w_2\, w_4\, w_7\, w_9 \end{bmatrix}$$

From the above updated state space matrix it is inferred that the object is tracked at location of particles present in matrix. Thus the particles not present in matrix are discarded. This matrix corresponds to location of object and in subsequent frame this matrix is used as reference. The process repeats for all the video frames by retaining the particles with more likelihood. The simulation results of few model are shown below:

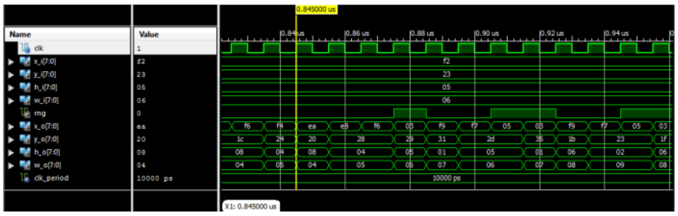

Fig. 3. Experimental results showing result of particle update in modelsim

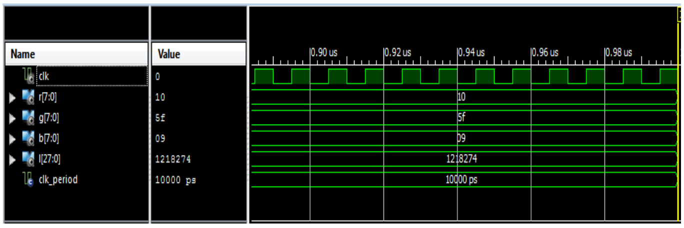

Fig. 4. Experimental results showing result of log likelihood in modelsim

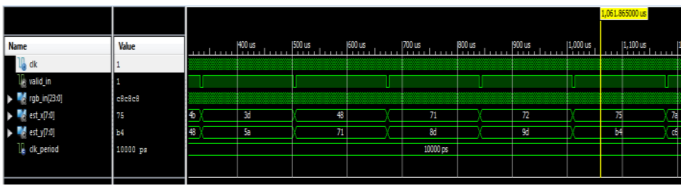

Fig. 5. Experimental results showing result of estimated center position of object in modelsim

The experimental results of particle update and log likelihood module is shown in Fig 3 and Fig 4 respectively. The result is obtained considering some arbitrary values of R, G and B components. The estimated position of object as shown in Fig. 5 is computed for the target object. The estimation is done on the video frames stored as text file.

VI. CONCLUSION

This paper deals with the object tracking method based on particle filter algorithm. For FPGA implementation the algorithm is optimized for utilization of limited hardware resources. This architecture is designed for parallel implementation using controller for executing finite state machine. This framework can be extended for multiple object tracking.

ACKNOWLEDGMENT

I would like to express sincere thanks to my beloved and respected Guide Prof. Rohita P. Patil for her encouragement and support. I gratefully acknowledge her for imparting me valuable basic knowledge of Digital Image Processing.

ISSN: 2278 – 909X
International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)
Volume 5, Issue 6, June 2016

REFERENCES

[1] L. Tiancheng, M. Bolicand P. M. Djuric, "Resampling methods for particle filtering: Classification, Implementation and Strategies," *IEEE Signal Processing Magazine*, vol. 32,no.3,pp. 70-86,May 2015

[2] *A.* Rodriguez and F. Moreno, "Evolutionary Computing and Particle Filtering :A Hardware-Based Motion Estimation System," *IEEE Transactions On Computers*, vol.64, no.11, pp. 3140 – 3152, Feb. 2015.

[3] S. Agrawal, P. Engineer, R. Velmurugan, and S. Patkar, "Parameterizable FPGA framework for particle filter based object tracking in video," *28th International Conference on VLSI Design and 14th International Conference on Embedded Systems*, pp. 35–40, Jan 2015.

[4] Wang Huan, Wang Qinglin, Wang Meng and Dai Yaping "A Particle Filter Algorithm for Real-Time Multiple Objects Tracking Based on Color Local Entropy," *IEEE International Conference On Instrumentation, Measurement, Computer, Communication and Control*, pp. 114 – 119, Sept. 2013.

[5] 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,March 2013.

[6] S. Agrawal, P. Engineer, R. Velmurugan, and S. Patkar, "FPGA implementation of particle filter based object tracking in video," *Proceedings of the 2012 International Symposium on Electronic System Design*, pp. 82–86, Dec 2012.

[7] S.-A. Li, C.-C. Hsu, W.-L. Lin, and J.-P. Wang, "Hardware/software codesign of particle filter and its application in object tracking," *International Conference on System Science and Engineering (ICSSE)*,pp. 87-91, June 2011.

[8] D.C. Cherng, S.Y. Yang, C.F. Shen, and Y.-C. Lu, "Real time color based particle filtering for object tracking with dual cache architecture," IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS), pp. 148–153, Sept 2011

[9] Q. P. H. Ying-Hao Yu and N. M. Kwok, "FPGA-based real-time color tracking for robotic formation control," *Proceedings of the 26th International Symposium on Automation and Robotics in Construction(ISARC)*, pp. 252–258, 2009.

Swati S. Jadhav,PGStudent,Elect. & Telecom Department Smt. KashibaiNavale College of Engineering,Vadgaon(BK) Pune, India

Rohita P. Patil, Asst. Professor Elect.& Telecom Department Smt. Kashibai Navale College of Engineering,Vadgaon(BK),Pune, India

.