*IOSR Journal of Computer Engineering (IOSR-JCE)*
*e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 12, Issue 6 (Jul. - Aug. 2013), PP 36-42*
*www.iosrjournals.org*

# A Novel Approach for Tracking with Implicit Video Shot Detection

Kiran S[1], Amith Kamath B[2]
[1,2]*(Master of Technology, Department of Computer Science, NMIT, Bangalore, India)*

***Abstract:*** *Video shot detection – Shot change detection is an essential step in video content analysis. The field of Video Shot Detection (VSD) is a well exploited area. In the past, there have been numerous approaches designed to successfully detect shot boundaries for temporal segmentation. Robust Pixel Based Method is used to detect shot changes in a video sequence. Tracking algorithm is a time consuming process due to the large amount of data contained in video using the video shot detection the computational cost can be reduced to a great extent by the discarding the frames which are not of any interest for the tracking algorithm. In this paper we present a novel approach of combining the concepts of Video shot detection and Object tracking using particle filter to give us a efficient Tracking algorithm with implicit shot detection.*
***Keywords –*** *Bhattacharyya distance, Local adaptive threshold, Particle filter, Robust Pixel Difference method, Residual Re-sampling, Shot detection.*

## I. INTRODUCTION

The rapid development of storage and multimedia technologies has made the retrieval and processing of videos relatively easy. Temporal segmentation is a fundamental step in video processing, and shot change detection is the most basic way to achieve it. However, while hard cuts (abrupt transitions) can be easily detected by finding changes in a color histogram, gradual transitions such as dissolves, fades, and wipes are hard to locate.

In practice, however, 99% of all edits fall into one of the following four categories hard cuts, fades, wipes and dissolves. Many shot change detection studies focus on finding low-level visual features, e.g., color histograms and edges, and then locate the spots of changes in those features. We focus on using Robust Pixel Method for shot detection. The conventional shot detection method using pixel wise comparison is not very efficient since it doesn't provide noise tolerance and because of its global thresholding nature. Many scenes involving sudden illumination changes such as lighting etc false trigger a shot change in the conventional method. The robust pixel method used in this paper provides threshold for noise and also locally adaptive threshold which makes it effective in situations where the conventional method fails.

Object tracking is an important task in the field of computer vision. It generates the path traced by a specified object by locating its position in each frame of the video sequence. The use of Object tracking is pertinent in many vision applications such as automated surveillance, video indexing, vehicle navigation, motion based recognition, security and defence areas. Occlusion and noise are generally the biggest problems in any target tracking implementation. Tracking algorithms robustness is a measure of how well it continues to track and when it loses its target. Tracking is the observation of person(s) or object(s) on the move and supplying a timely ordered sequence of respective location data to a model under consideration. It is the process of locating a moving human or object over time using a camera. It is based on computer vision. Image registration is the basic step used in tracking application. It is a process that finds the location where a good matching is obtained by matching the template over the searching area of an input image. Registration algorithm fails in complex situations and loses the target in presence of noise, scaling and transformation changes. To address the above mentioned problems, we use a particle filter based tracking method for efficient tracking. Video shot detection and tracking algorithms have both been extensively researched and have been used in real world applications individually. Very less effort has been made to combine the concepts of video shot detection and tracking which can be of great help in real-world as both the technologies complement each other. Combining the two concepts guarantees a computationally quicker, cost effective solution for tracking on large video database with minimal pre-processing. In this paper Section II covers the concepts of Video shot detection using robust pixel difference method and also demonstrates it effectiveness with results. Section III focuses on concepts of tracking algorithm with results. Section IV elaborates the method of combining the two approaches where tracking algorithm is initiated after every shot change hence serving its final purpose of computationally efficient shot detection cum tracking system.

## II. VIDEO SHOT DETECTION

### 1.1 Robust Pixel Method(Rpm)

In Robust Pixel Method, we consider a Metric M, computed for each Video Frame.
Defined as:

$$M(I^K | I^{K-1}) = \frac{1}{HW} \sum_{i,j} \rho_{i,j}^K$$

where Ik and Ik-1 are each pair of consecutive images to be compared. H and W are the image height and width.(i; j) are the coordinates of each one of the pixels in the image.
And ρ is:

$$\rho_{i,j}^K = \begin{cases} 1 & if\ |I_{i,j}^K - \mu^K| > T_n\ \&\ sign(I_{i,j}^K - \mu^K) = sign(I_{i,j}^{K-1} - \mu^{K-1}) \\ -1 & f\ |I_{i,j}^K - \mu^K| > T_n\ \&\ sign(I_{i,j}^K - \mu^K) \neq sign(I_{i,j}^{K-1} - \mu^{K-1}) \\ 0 & otherwise \end{cases}$$

where µk is the mean value of the image Ik, and Tn is a noise threshold (Heuristically set to Tn = 2). Between two consecutive images belonging to the same shot, a large amount of pixels change their values when there is a sudden illumination change. Conventional pixel-based method computes a large difference between and falsely classify the images as a transition or shot change. This limitation is addressed by computing the above mentioned metric M, as all the pixels in an image change the intensity evenly in case of sudden illumination change. Therefore, if the difference between pixel values and the mean of the image is computed, for each one of the two consecutive images, no significant variations occur for pixels. Robustness to sudden illumination changes is achieved by using Metric M for purpose of Shot Change detection. The Noise threshold Tn takes care of any noise distortions in images.

### 1.2 Locally ADAPTIVE THRESHOLDING

Conventional Pixel difference method uses a Global Threshold value which is computed using the mean and standard deviation of the Pixel Difference values for the entire video, this threshold is often too High ( does not detect many shot changes that are actually present in the video) or too Low (Falsely Detects shot changes).In Robust Pixel Method it is necessary to use a locally adaptive threshold value, to overcome false detections, or missed shot changes [5]. Therefore we can consider a window of 20 frames ( as not more than 1 shot change occurs in a window of 20 frames) as Tw, for which we adaptively compute a threshold, and detect if there has been a shot change or not. We consider the Value of Metric M for every frame between Ia and Ia+Tw to set the threshold. The two local minimum values of M between Ia and Ia+Tw are identified as M min1 and Mmin2. The shot is classified as a shot Change only if:

$$| M_{\min 1} - M_{\min 2} | > \quad T_{LocalAdaptve}$$

Where T Local Adaptive is heuristically set to 0.3.

### 1.3 RESULTS


Fig 1. Test Data set to demonstrate the effectiveness of robust pixel method.

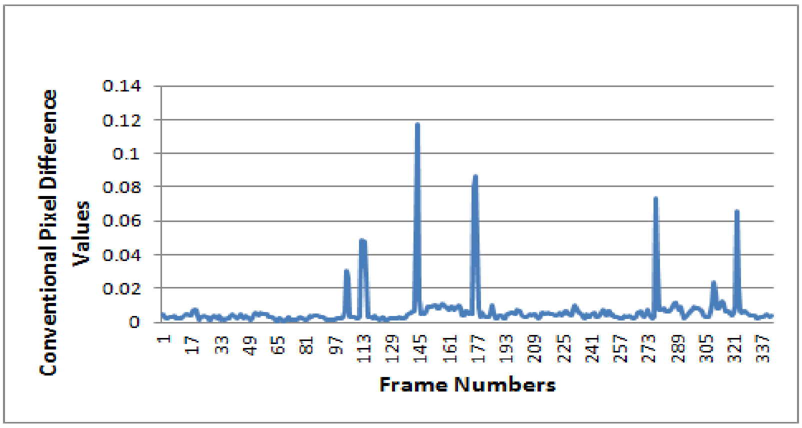


Fig 2. Graph of conventional pixel difference for above test video

Conventional Pixel Difference Method Falsely Detects 7 shots, due to the sudden change of intensity in the video.

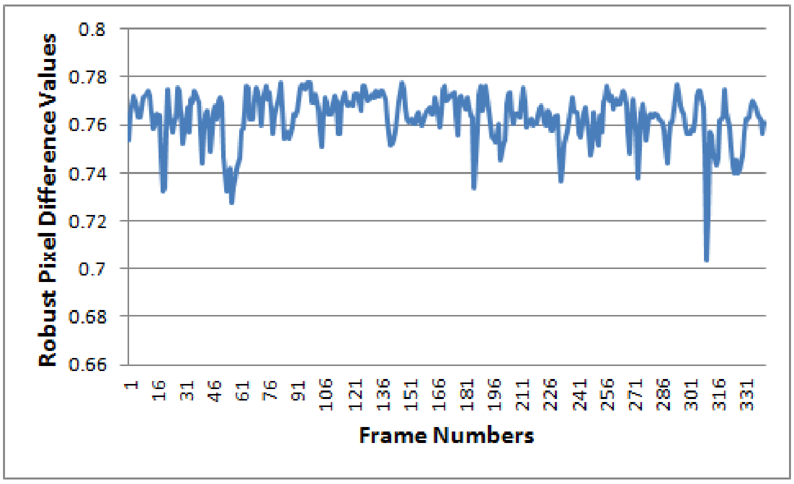


Fig 3. Graph of Robust pixel difference for above test video sequence

Robust Pixel Difference Method does not detect any false shot changes which occur due to sudden change in global illumination. Hence this method is more robust and resistant to sudden intensity changes [6].

## III. PARTICLE FILTER BASED TRACKING

Particle filtering has emerged recently in the domain of computer vision. Particle Filter is concerned with the problem of tracking single and multiple objects. It is a hypothesis tracker that is intended to explain certain observations, that approximates the filtered posterior distribution by a set of weighted particles. It weights the particles based on the likelihood score and propagates them according to the motion model used The advantage of particle filter over other types of filters (Kalman, Extended Kalman, etc.) is that it allows for a state space representation of any distribution. It also allows for non-linear, non-Gaussian models and processes.

Particle Filter is concerned with the problem of tracking single or multiple objects. It is a hypothesis tracker that is intended to explain certain observations, that approximates the filtered posterior distribution by a set of weighted particles. It weights the particles based on the likelihood score and propagates them according to the motion model used. Particle filter algorithm is a popular substitute for the Kalman filter in presence of non-Gaussianity of the noise statistics and non-linearity of the relationships between consecutive state. Particle Filter is a promising technique because of its inherent property that cope up with data association problems, account for certain uncertainties, also allows fusion of other algorithms and data.
The algorithm for Particle filter based tracking is shown below.

### 1.4 STEPS

i. Select the number of samples(particles).
ii. Select the target to track and initiate the start point.
iii. Generate a randomly distributed uniform number of particles around the start point.
iv. Observe the color distribution.

v. Calculate the RGB color distribution from each sample of the set.
vi. Then calculate the Bhattacharyya coefficient for each sample of the set 's'.
vii. Weight each sample.
viii. Resample these samples using residual Re-Sampling.
ix. Iterate these steps.

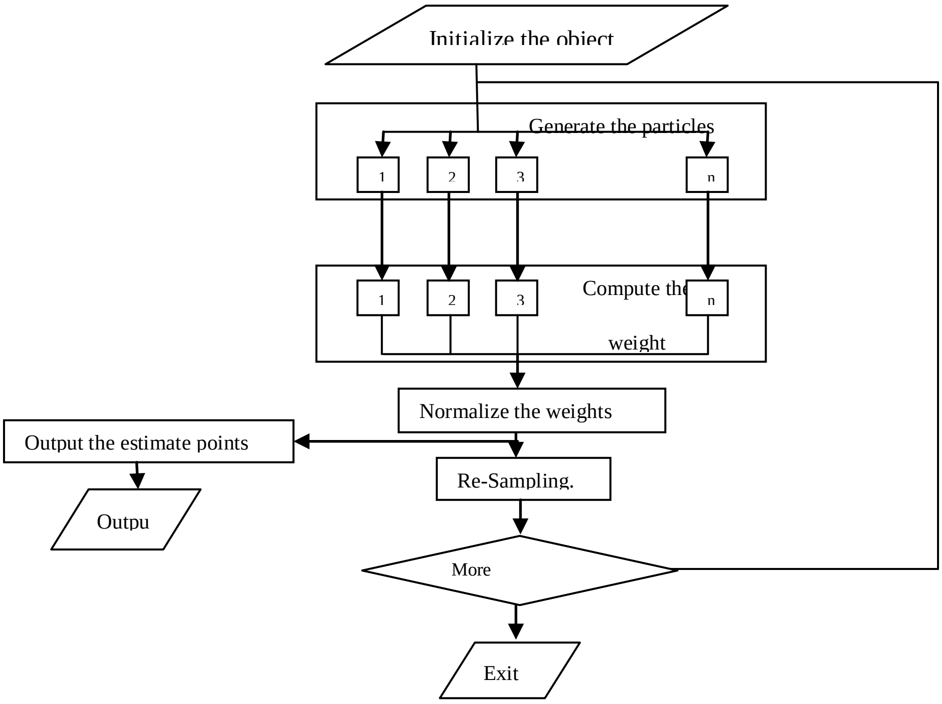


Fig 4. Flowchart of Particle filter based tracking algorithm

The tracking algorithm employs color model based particle filter. It integrates the Color distribution into particle filtering. [3] They are applied as they are robust to partial occlusion, are rotation and scale invariant and computationally efficient. The observation model of particle filter is defined by the Color information of the tracked object. This model is compared to hypotheses of the grey model particle filter using the Bhattachayya coefficient. Color histograms in particular have many advantages for tracking non-rigid objects as they are robust to partial occlusion, are rotation and scale invariant and are calculated efficiently. A target model is tracked using particle filter by comparing its histogram with the histograms of the sample positions using the Bhattacharyya distance and further Re-Sampling them. A complete segmentation of the image is not required as the image content only needs to be evaluated at the sample positions. We apply such a particle filter in a color-based context. To achieve robustness against non-rigidity, rotation and partial occlusion we focus on color distributions as target models. These are represented by histograms which are produced with the function h(xi), that assigns one of the m-bins to a given color at location (xi). The histograms are typically calculated in the RGB space using 8x8x8 bins. To weight the sample set, the Bhattacharyya coefficient has to be computed between the target distribution and distribution of hypotheses in RGB color space. Bhattacharyya coefficient or the Bhattacharyya distance [1], measure the similarity of two continuous probability distributions. The coefficient can be used to determine the relative closeness of the two samples being considered.

$$d(H_1, H_2) = \sqrt{1 - \frac{1}{\sqrt{\bar{H}_1 \bar{H}_2 N^2}} \sum_I \sqrt{H_1(I) \cdot H_2(I)}}$$

**Eqn 1: Bhattacharyya distance measure**

The measurement process is based on histogram similarity: the target histogram is compared with that of other candidate patches extracted from last captured frame and the most similar one is chosen. The similarity measure is directly computed over the entire sample point sets. The affinities between all pairs of sample points are considered based on their distances. We can compute the similarity measure between target model and candidate model by applying distance measure methods. The value obtained from the distance measure for every particle set is called the score. They are used to represent and predict the next match point. The particles

set are to be re-sampled to select and re-generate the particles around the target location to keep the tracking process intact. This avoids particle degeneracy. Residual re-sampling method is used in our algorithm [2].
The algorithm for Residual Re-Sampling is shown below
Input: Match Values, Score points
Output: Re-sampled particles

**1.5 STEPS FOR RESIDUAL RE-SAMPLING**

i. Calculate the sum of scores and select the maximum score.
ii. Select the index of the particle with maximum score.
iii. Normalize all the weights of the particles to the sum of scores.
iv. Initialize the Index with Integer part.
v. Select only the particles with particles greater than 1.
vi. Generate random particles around the particle with maximum score with existing particles.

The above steps[2] give only those selected particles that assist to predict the object in further frames. The particles will be re-generated at each frame for prediction. After re-sampling step the algorithm again runs observation part calculates scores and re-samples them. These steps run iteratively. Even when re-sampling steps is in progress at some time few particles having low weights tends to be discarded hence to maintain the number of particles, the remaining number of particles that are discarded has to be randomly re-generated.

**1.6 RESULTS**

Below shown are the results of particle filter based tracking algorithm that uses HSV color model to show is insensitivity to light changes taken in an indoor environment with illumination changes, it uses remainder re-sampling for regeneration of particles. The algorithm provides satisfactory results and is test for videos that has various light changes, scaling and rotation.


Fig 5. Output of Particle Filter based tracking algorithm on the sequence of video frames with illumination changes.

## IV. COMBINING OF VIDEO SHOT DETECTION AND TRACKING ALGORITHM

This section elaborates combining methodology of shot detection along with tracking. The proposed model acts at two levels. The first level is Video Shot Detection, where a huge video is temporally segmented into individual shots based on the algorithm described in Section II. The second level is particle filter based tracking in the video sequence as described in Section III. Upon each occurrence of a shot change the Video shot detection algorithm is interrupted and the tracking algorithm is triggered for initiation. On initiation, the object to be tracked has to be selected manually by a mouse where the position of the object is fed to the algorithm. If the tracking algorithm is not initiated then shot detection algorithm continues until it finds the next shot change. The tracking algorithm, whenever active, runs in phase with video shot detection algorithm, the detection of next shot change terminates the tracking algorithm. The algorithm again prompts the user for a valid input either to initiate the tracking algorithm or continue the shot detection algorithm at the occurrence of next shot change. Tracking algorithm will start from the video frame that is fed by shot detection algorithm at a shot change. It will continue tracking the selected object until the algorithm is interrupted for termination. In a large video data such as unedited footage of CCTV surveillance etc, this approach guarantees reduction of pre-processing time by directly presenting the tracking algorithm, the initial frame from which tracking can be initiated. As tracking algorithm has high computational demand, this approach greatly reduces the computational complexity by discarding unwanted frames and triggering tracking algorithm only when necessary.

**4.1 Proposed Method**

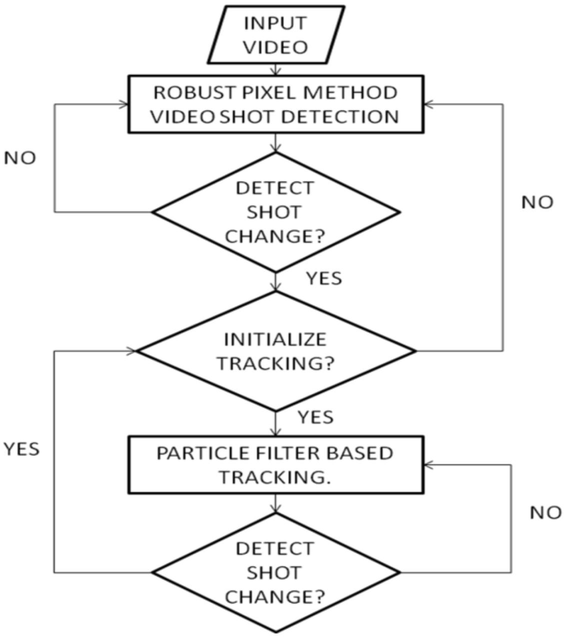


Fig 6. Flowchart showing the methodology for combining Video shot detection and tracking.

## V. EXPERIMENTAL RESULTS

Fig 7.

In fig 7 in a transition from frame 316 to frame 317 a shot change has been detected and the user is prompted for input to initiate the tracking algorithm when the user initializes the object, the particle filter algorithm tracks the object continuously until the next shot change is detected. In this phase the Video shot detection algorithm will work in phase with the tracking to detect every possible shot change. It is also observed that tracking and video shot detection continues positively without losing the target and false shot detection respectively even in presence of illumination changes. When the algorithm encounters frame 523 a new shot has been detected and the tracking algorithm is asked for a decision whether to re-initiate the tracking or to terminate it. On termination the video continues with shot detection algorithm to find the next change in shot.

## VI. CONCLUSION

Tracking with implicit shot detection algorithm provides an efficient framework for temporal segmentation and tracking in huge videos. It reduces data computational complexity to a great extent. In our methods we have observed the merits of video shot detection using robust pixel method that is insensitive to light sources or sudden illumination changes and its locally adaptive thresholding adds additional adaptability to the approach. Particle filter based tracking is an promising technique in various situation and also is insensitive to illumination changes and scaling. The advantages of both the methods makes it possible to integrate them together to provide real-time applicable system which automatically detect shot change and also gives a chance to track the objects of interest.

## REFERENCES

[1] The computation of the bhattacharyya distance between histograms without histograms S´everine Dubuisson Laboratoire d'Informatique de Paris 6, Universit´e Pierre et Marie Curie. *Presented at: TIPR'97, Prague 9-11 June, 1997.and publiseed in Kybernetika, 34, 4, 363-368, 1997.*

[2] An Efficient Fixed-Point Implementation of Residual Resampling Scheme for High-Speed Particle Filters Sangjin Hong, Member, IEEE, *Miodrag Bolic´, Student Member, IEEE, and Petar M. Djuric´, Senior Member, IEEE IEEE SIGNAL PROCESSING LETTERS, VOL. 11, NO. 5, MAY 2004.*

[3] An adaptive color-based particle filter Katja Nummiaroa,*, Esther Koller-Meierb, Luc Van Gool Katholieke Universiteit Leuven, *ESAT/PSI-VISICS, Kasteelpark Arenberg 10, 3001 Heverlee, Belgium* Swiss Federal Institute of Technology (ETH), D-ITET/BIWI, Gloriastrasse 35, 8092 Zurich, Switzerland..

[4] A Shot Boundary Detection Technique Based On Local Color Moments In Ycbcr Color Space S.A.Angadi and Vilas Naik Natarajan Meghanathan, et al. (Eds): *SIPM, FCST, ITCA, WSE, ACSIT, CS & IT (2012)*

[5] Keyframe Based Video Summarization Using Automatic Threshold & Edge Matching Rate. Mr. Sandip T. Dhagdi, Dr. P.R. Deshmukh *,International Journal of Scientific and Research Publications, July (2012)*

[6] Real-time shot detection based on motion analysis and multiple low-level techniques.Carlos Cuevas and Narciso Garcia,Grupo de Tratamiento de Imagenes - *E.T.S. Ing. Telecomunicacion (2010)*