IIT KHARAGPUR
2015

Optimization of MOSSE Tracker

Project Report

Ankit Deora
12EE10004

Introduction

Object Tracking is an important task in the field of computer vision. This involves estimating the location of a object in each frame of an image sequence or a camera feed. Developing an algorithm for accurate and precise tracking is a challenging problem. Object tracking has a large number of applications like gesture recognition, traffic monitoring, eye gaze tracking, security surveillance systems, etc. In this report we discuss the implementation of a hybrid tracker which incorporates Lucas-Kanade or LK features in to the original MOSSE tracker. The tracker outperforms most of the trackers which fail to handle large scale variations. Implementation of LK features has reduced the sudden motion failure to a great extent. The performance of the tracker is found to be good for object models with good texture. The algorithm is implemented in python using the OpenCV library.

Later in the report we discuss the performance of the proposed tracker and MOSSE tracker on 25 benchmark sequences from VOT (visual object tracking) challenge 2014 dataset. Our results show that the proposed algorithm improved the performance by 48.5 % of the original MOSSE tracker.

Tracking of a ball

Literature Review

A Lot of research has been done on object tracking algorithms and is still going on. Trackers generally differ from each other based on their approach to object detection, tracking features and object representation.

In the literature, the Kalman filter has been extensively used in the vision community for tracking. Broida and Chellappa [4] used the Kalman filter to track points in noisy images. Tanizaki [19] used Particle filter to overcome the limitations of Kalman filter.

Commonly used interest point detectors include Moravec's interest operator[14], Harris interest point detector [8], KLT detector [18], and SIFT detector [13]. Haritaoglu et al. [7] and Collins et al. [5] use background subtraction methods to detect regions of interest. Rasmussen and Hager [17] use a constrained JPDAF(*Joint Probability Data Association Filter*) filter to track regions.

Cox and Hingorani [6] use Murty's [15] algorithm to determine k-best hypotheses in polynomial time for tracking interest points. Isard and MacCormick [10] propose joint modelling of the background and foreground regions for tracking. Avidan [1] used a Support Vector Machine (SVM) classifier for tracking. In 2000, MacCormick and Blake extended the particle filter-based object tracker in Isard and Blake [9] to track multiple objects by including the exclusion principle for handling occlusion.

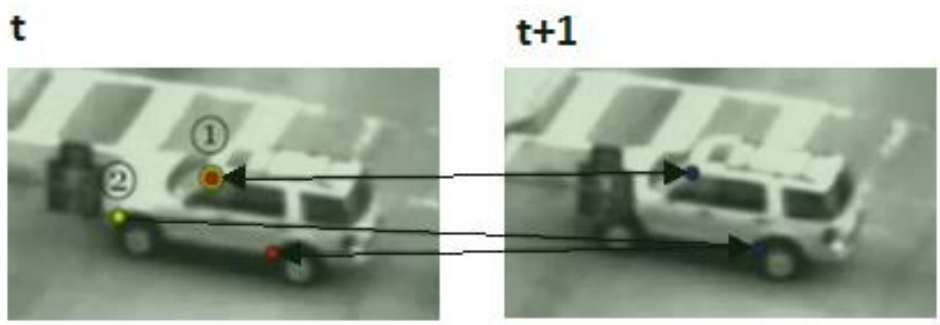
Probabilistic state-space methods including Kalman Filters by Bar-Shalom and Foreman [2], JPDAFs by Cox [6], HMMs by Rabiner [16], and Dynamic Bayesian Networks (DBNs) by Jensen [11] have been extensively used to estimate object motion parameters.

The above mentioned tracking techniques utilizes one particular feature for tracking, thus limiting their scope and performance to a particular type of object models.

MOSSE tracker proposed by David S. Bolme outperforms most of the trackers because of its online updation of the object model and good performance. The object model in the first frame is selected by the user which makes it model free tracker and the implementation in Fourier Domain makes the computations quite fast. In spite of being a robust tracker, MOSSE tracker has limitations like scaling, redetection after failure, drifting of object model and sudden motion failure. Our hybrid tracker integrates LK features within the original MOSSE tracker. Use of LK features makes our object model scalable and also overcomes the limitation of sudden motion failure.

Proposed Algorithm

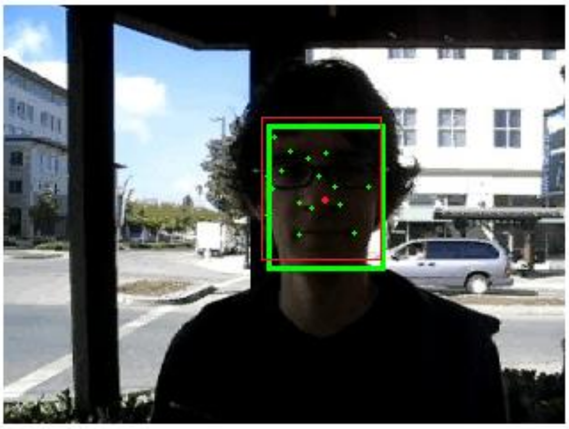
MOSSE tracker implements a Filter which is convolved with the object model to get the desired output. This output gives the updated centre of the bounding box in every frame and the bounding box size is same throughout. For the purpose of rescaling the bounding box, Filter is needed to be rescaled first. To avoid sudden motion failure a new centre of the bounding box is needed. LK tracker provides both the parameters – rescaling ratio as well as the new centre .In our approach we took some points p0 on the object to track in a particular t^{th} frame and the corresponding p1 points were found in the next $(t+1)^{th}$ frame using back and forth method of LK tracker. Using the information provided by these points both the parameters were calculated.



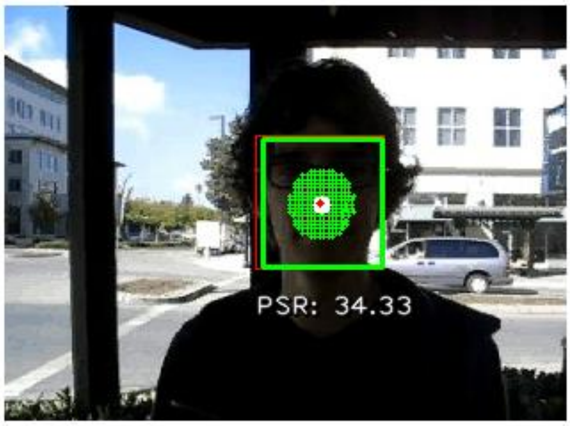
p0 points P1 points

back and forth method

The points were earlier selected using Shi-tomashi good points to track function but the number of points obtained from this method was quite less resulting in lower performance of LK tracker. So to get more number of points on the object, points were selected from uniform distribution by dividing the object in equal number of portions. To keep the central portion of the object in focus and to avoid tracking of points in background, points were selected in an elliptical window. This way of selecting the points ensured that number of points to track is large and always in the central portion of the object.



shi-tomashi points



uniform points in elliptical window

Rescaling Implementation

We used the basic concept of zoom-in and zoom-out for rescaling purpose. Whenever an image is zoomed in or an object comes closer in the video frame then the distance between any two points on the object increases. Similarly whenever an image is zoomed out or an object goes farther in the video frame then distance between any two points on the object decreases. In our approach median of the distances were found in every single frame and ratio was calculated by dividing it with the value obtained in previous frame. Now the rescaling ratio so obtained was passed to the Filter in every frame and the filter was updated with new scaling.

Object far from camera

Object close to camera

Pseudo code to obtain rescaling ratio and new centre :

For every p0 point in t image:
  For every pt. point in t+1 image:
    If distance between pt. and p0 < d_offset:
      p1 = pt.

$$\text{rescaling ratio} = \frac{\text{median}(p0) - \text{pt. in } t \text{ image}}{\text{median}(p1) - \text{pt. in } (t+1) \text{ image}}$$

new centre = median(p1)

Finding new centre

New centre was selected from the p1 points in $(t+1)^{th}$ image. Instead of taking average of all the trackable points on the object, median was calculated. Median was used for finding centre to nullify the effect of faraway points or points on the background,. This was used as the new centre of the bounding box whenever there was sudden motion failure.

Reinitialization of LK tracker

Occlusion is a major problem while working with LK tracker. Whenever there is a partial or full occlusion , number of points tracked by LK technique tends to reduce. If the occlusion continues for long time, number of points might reduce to zero in some cases. To make sure that points tracked by LK technique do not become too less, points were to be reproduced by some technique. For this purpose, points were reinitialized after a particular number of frames (5 in our case). This ensured the smooth working of the LK tracker.

No Occlusion

Occlusion due to trees

Software Architecture

The algorithm was implemented in python. Modular implementation of the algorithm made it possible to track more than one object in a video feed. The algorithm has three major parts :

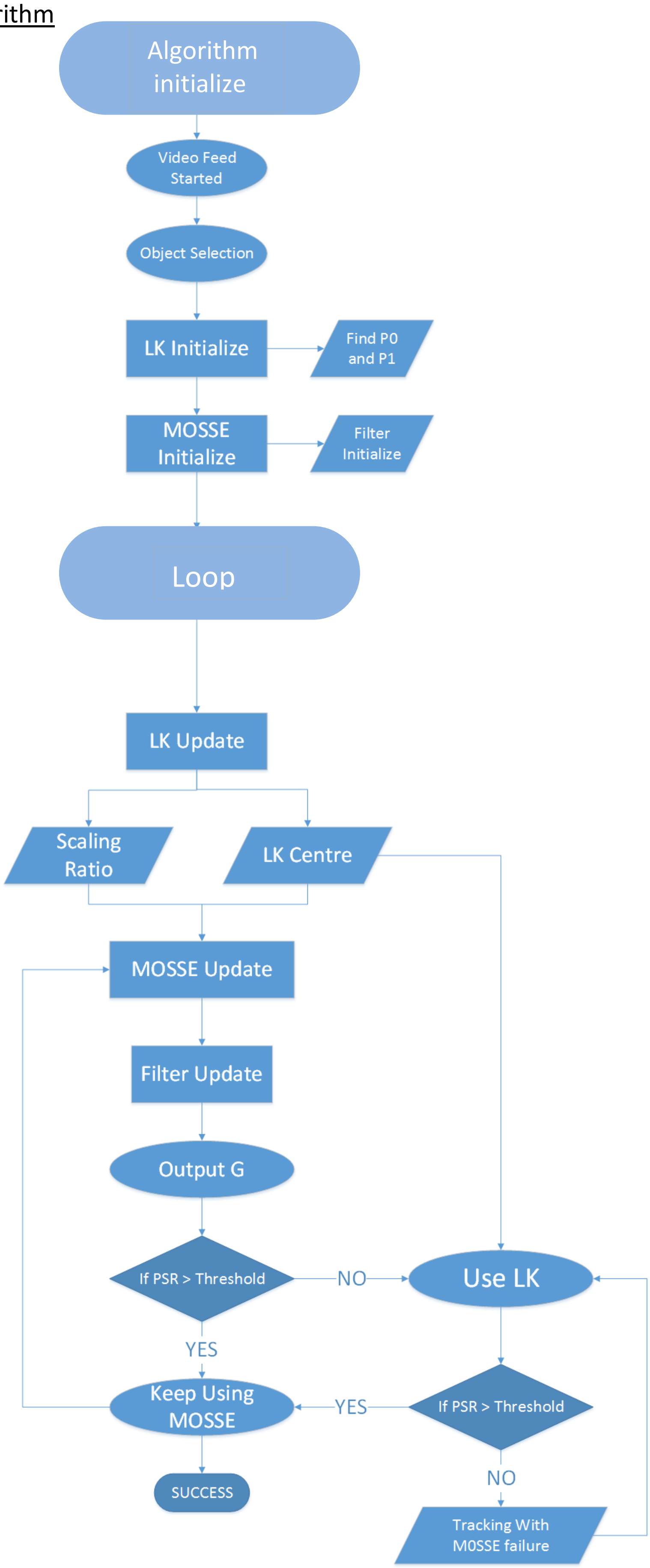
(a) **Initialization** –
- in this part, video feed, LK tracker and MOSSE tracker were initialized.
- p0 points were obtained in the first frame using the uniform distribution in elliptical window.
- In MOSSE tracker the filter was initialized from the first frame. Affine variations were implemented on the filter to make it more robust for PSR calculation.
- PSR is obtained from the output obtained by convolving 2d filter matrix with the 2d object matrix. In the output so obtained PSR is the ratio of peak value and standard deviation.

(b) **Updation** –
- in this part, points to track were updated using LK tracker's back and forth technique and rescaling ratio and new centre were calculated. These parameters were passed to the MOSSE update function to update the filter.
- The filter was rescaled to the scaling ratio by using linear interpolation. To avoid sudden changes of rescaling ratio, ratio was calculated using the present ratio as well as past values.
- A parameter was defined to change the weightage give to the present and past values. For gradual change in the ratio value an optimum value of the parameter was chosen.

(c) **Decision** –
- in this part, tracker's success was calculated using PSR values.
- This PSR value was compared with a particular threshold value, for larger values of PSR tracking was considered as successful and smaller values than the threshold were considered as failure.
- Whenever the tracker failed due to decrease in PSR value then new centre from LK tracker was used for new Bounding Box location and again the PSR value was calculated. In this way the LK tracker worked as a backup tracker.

After implementing the above algorithm, hybrid tracker was tested on camera feed. Overall performance of the hybrid tracker was observed to be better than the original MOSSE tracker. Bounding box rescaled as expected and the sudden motion failure also reduced to a great extent.

Block diagram of Algorithm



Experiments and Results

Both the hybrid tracker and the MOSSE tracker were tested on VOT 2014 and 2015 dataset videos. Their performance was observed by finding two parameters :-

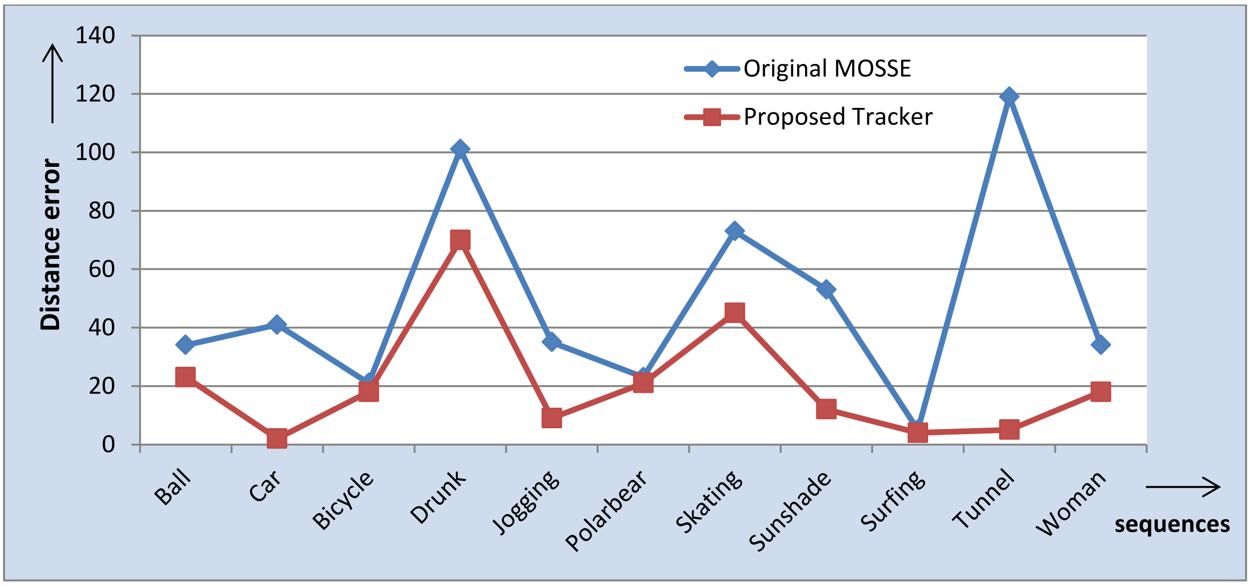
1. Overlapping area between bounding box and ground truth
2. Distance between centres of bounding box and ground truth

Lesser the distance parameter and more the Area parameter, better the performance of the tracker. Overlapping area was found to be more in case of the hybrid tracker as compared to the MOSSE tracker in the maximum videos and also the distance between centres was less in hybrid tracker which shows the better performance of hybrid tracker over the MOSSE tracker. Following snapshots show the performance of the optimized tracker and the original MOSSE tracker on a video sequence "jogging" from the 2014 dataset videos :

Optimized MOSSE tracker

Original MOSSE tracker

Following chart shows the performance of both trackers on some video sequences from VOT 2014 dataset based on the distance parameter. Overall calculations showed an improvement of **48.5 %** in the performance of the proposed tracker over the MOSSE tracker.



Results based on Distance Parameter

Conclusion

Implementation of the hybrid tracker shows how the performance of an object tracker can be improved by incorporating different features (LK features in our case). The hybrid tracker is computationally slower as compared to the MOSSE tracker but the performance has greatly improved. This report shows how limitations of non-rescaling and sudden motion failure of MOSSE tracker can be eliminated using LK technique. The hybrid tracker can also be extended to eliminate the other limitations of MOSSE tracker i.e. redetection after complete failure, rotation of bounding box etc. Possible future works include :

- Rotation of Bounding box by finding angle from LK points
- Implementing Particle filter for redetection after occlusion
- Implementing computations on coloured images instead of grayscale images

References

[1] AVIDAN, S. 2001. Support vector tracking. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 184–191.

[2] BAR-SHALOM, Y. AND FOREMAN, T. 1988. Tracking and Data Association. Academic Press Inc.

[3] Bolme, David S., J. Ross Beveridge, Bruce A. Draper, and Yui Man Lui. "Visual object tracking using adaptive correlation filters." In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pp. 2544-2550. IEEE, 2010.

[4] BROIDA, T. AND CHELLAPPA, R. 1986. Estimation of object motion parameters from noisy images. IEEE Trans. Patt. Analy. Mach. Intell. 8, 1, 90–99.

[5] COLLINS, R., LIPTON, A., FUJIYOSHI, H., AND KANADE, T. 2001. Algorithms for cooperative multisensory surveillance. Proceedings of IEEE 89, 10, 1456–1477.

[6] COX, I. AND HINGORANI, S. 1996. An efficient implementation of Reid’s multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. IEEE Trans. Patt. Analy. Mach. Intell. 18, 2, 138–150.

[7] HARITAOGLU, I., HARWOOD, D., AND DAVIS, L. 2000. W4: real-time surveillance of people and their activities. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, 809–830.

[8] HARRIS, C. AND STEPHENS, M. 1988. A combined corner and edge detector. In 4th Alvey Vision Conference. 147–151.

[9] ISARD, M. AND BLAKE, A. 1998. Condensation - conditional density propagation for visual tracking. Int. J. Computer. Vision 29, 1, 5–28.

[10] ISARD, M. AND MACCORMICK, J. 2001. Bramble: A Bayesian multiple-blob tracker. In IEEE International Conference on Computer Vision (ICCV). 34–41.

[11] JENSEN, F. V. 2001. Bayesian Networks and Decision Graphs. Springer.

[12] Kalal, Zdenek, Krystian Mikolajczyk, and Jiri Matas. "Tracking-learning-detection." Pattern Analysis and Machine Intelligence, IEEE Transactions on 34.7 (2012): 1409-1422.

[13] LOWE, D. 2004. Distinctive image features from scale-invariant key points. Int. J. computer. Vision 60, 2, 91–110.

[14] MORAVEC, H. 1979. Visual mapping by a robot rover. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). 598–600

[15] MURTY, K. 1968. An algorithm for ranking all the assignments in order of increasing cost. Operations Research. 16, 682–686.

[16] RABINER, L. R. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77, 2, 257–286.

[17] RASMUSSEN, C. AND HAGER, G. 2001. Probabilistic data association methods for tracking complex visual objects. IEEE Trans. Patt. Analy. Mach. Intell. 23, 6, 560–576.

[18] SHI, J. AND TOMASI, C. 1994. Good features to track. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 593–600.

[19] TANIZAKI, H. 1987. Non-Gaussian state-space modelling of nonstationary time series. J. Amer. Statist. Assoc. 82, 1032–1063.

[20] Yilmaz, Alper, Omar Javed, and Mubarak Shah. "Object tracking: A survey." Acm computing surveys (CSUR) 38, no. 4 (2006): 13.