Mean-Shift Tracking

Jae Lee

December 16, 2015

Abstract

Object tracking using mean shift algorithm is presented in this paper. The implementation presented here uses only color histogram of the object to be tracked. Then, mean shift algorithm is applied to conduct similarity search using the backprojection image of the object's color histogram.

1 Introduction

Object tracking is an interesting topic that has been attracting many researchers in the field of computer vision. It has a variety of applications such as human-computer interaction, surveillance, augmented reality, and traffic control etc. The goal of object tracking is to detect the object of interest and find its location continuously and robustly in a sequence of images that is either recorded or displayed in real-time. As one of the applications of object tracking, researchers in MAGICC lab directed by Dr. Randy Beard has been developing an algorithm called Recursive RANSAC to track multiple-objects simultaneously which is mainly intended to be used on unmanned aerial vehicle(UAV) platform.

The basic idea of R-RANSAC algorithm is first to identify any moving objects in a video sequence and store the trajectories of these objects. Second, it keeps managing a list of tracks using R-RANSAC algorithm. For example, if a track has existed for sometime above threshold and has a high enough inlier ratio which means previous measurements support the current measurement as a part of the track, the track is classified as a good track. Otherwise, it is classified as a bad track in a list that won't be displayed in the tracking result.

However, one thing that can be tried to make the R-RANSAC tracking more robust is to add a feature based tracker because R-RANSAC currently only cares about trajectories of the objects. In this paper, a tracking using mean shift algorithm on color histogram is examined as one possible option to enhance the robustness of R-RANSAC tracking.

As the name of the algorithm implies, mean shift is an algorithm that shifts a window (for example, a circle) so that the window includes the most possible points of interest. As the window continues to shift to include more points, eventually it is expected to arrive to a point that includes the maximum number of points. Because of this behavior, it is a hill climbing algorithm. Figure 1 below describes this process.

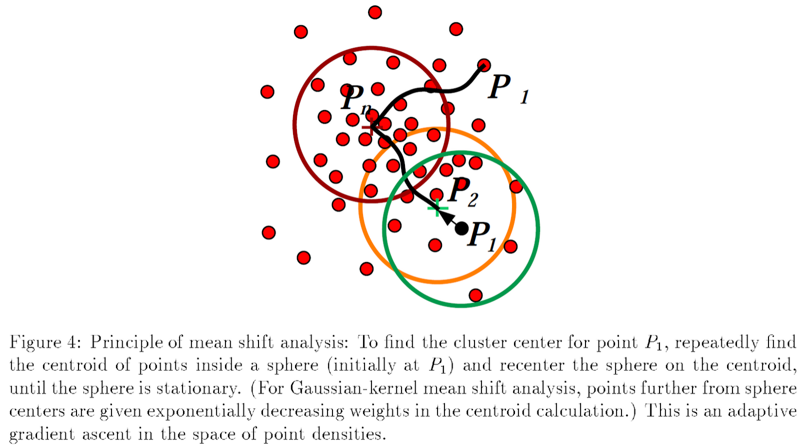

Figure 1: Mean shift Algorithm[1]

2 Procedure

Mean shift is just a fundamental algorithm that can be applied to solve any other problems, not just tracking problem. Itself doesn't imply any tracking capability. Thus, it seems appropriate to explain what other things need to happen before actually using the mean shift. The basic idea of mean shift tracking is to find a window that has the most similar color histogram in the current image to the color histogram of the object to be tracked as shown in Figure 2.

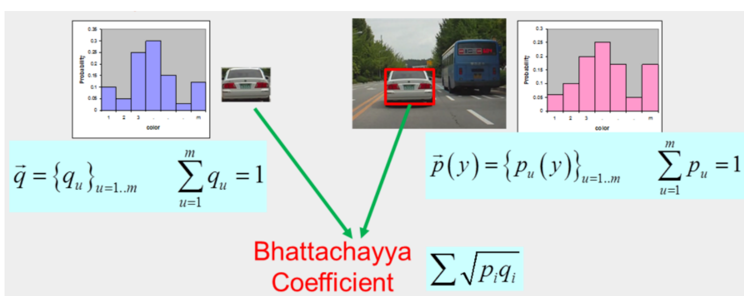

Figure 2: The color histogram of the object to be tracked is searched for in input image. The metric to measure the similarity of two histograms is Bhattachayya Coefficient.

However, it is computationally rather inefficient if the histogram is calculated for all possible windows in every frame of a video stream. For this reason, it is common to use histogram backprojection to reduce the computation[2]. When an image is backprojected with a model histogram, the resulting image will show the probability distribution of how similar to a model histogram. Then, mean shift is applied on the backprojected image to find where the track is likely to be. Figure 3 summarizes the process of mean shift tracking.

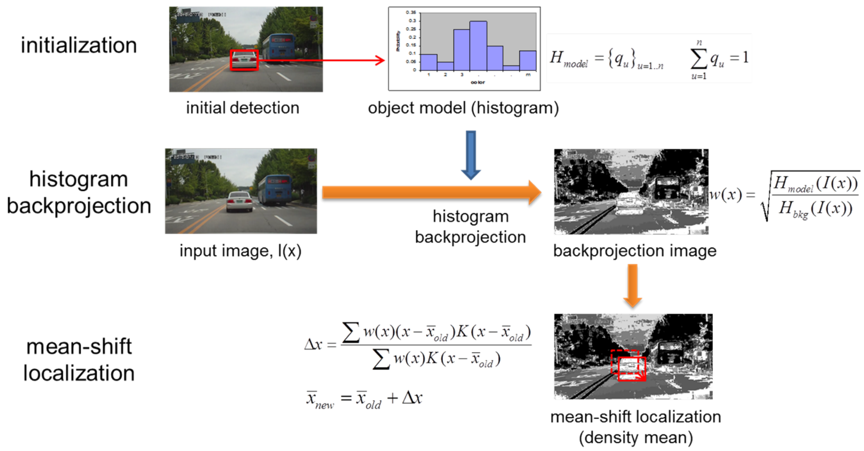

Figure 3: Mean shift tracking using backprojection method

3 Implementation

This version of mean shift tracking implementation takes advantage of Open Source Computer Vision library(OpenCV) since OpenCV provided with several proper functions for the implementation such as calcHist(), normalize(), calcBackproject(), and meanshift() as well as user interaction.

The implementation of mean shift tracking can be summarized as the following:

1. Pick a point on the object of interest that describes the color information of the track.
2. Use connected component algorithm to find more part of the track. Then, create a mask that only contains the part of the track. The connected component looks for every connected pixels within certain color similaries to a point picked in step 1.
3. Using the mask created in step 2, calculate and store the color histogram of the track. Create the initial search window of the track.
4. Backproject the incoming image with the histogram stored in step 3.
5. Apply mean shift starting at the position of the search window in the previous frame.

The Figure 4 - 8 show an example of each step.

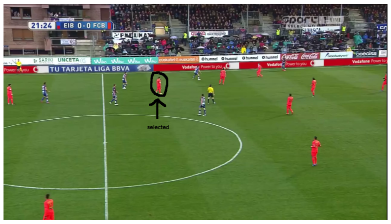

Figure 4: Step 1: Track selection

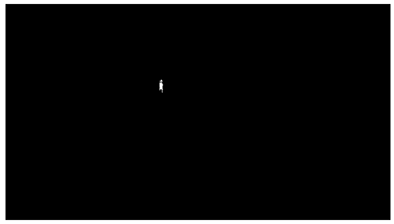

Figure 5: Step 2: Create a mask

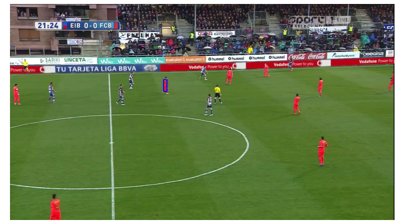

Figure 6: Step 3: Calculate the histogram and create the initial search window

Figure 7: Step 4: Backproject the incoming image with the histogram in step 3

Figure 8: Step 5: Apply mean shift

4 Experimental work

The mean shift tracking was experimented on three different video stream: messi.mp4, byu.mp4 and seoul.mp4. Messi data illustrates an ideal setting that mean shift tracking might work quite robustly because camera doesn't move abrubtly and the size of the track doesn't change dramatically throughout the video. Also, the color of the track and background are noticably different. BYU data has a similar setting to Messi data in a way that camera is almost static, but it is also different in a way that some objects can have similar color to the background color. Seoul data was filmed in the most complex and crowded environment and the result of the mean shift tracking shows that it is not robust in dealing with occlusions and in crowded setting.

4.1 Result on Messi data

The performance on Messi data yielded the most satisfactory result of the mean shift tracking. Once the track is selected, it is being tracked to the end most of times. Figure 9-11 shows a successful scenario of the tracking.

Figure 9: The track is selected at frame 2

Figure 10: The track is being tracked at frame 338

Figure 11: Tracking was successful up to the end of the video

However, when two players in the same team intersect, sometimes the track switches to the other player. Figure 12-13 shows this scenario.This shows that mean shift tracking can be enhanced when motion based tracker or other feature such as SIFT is used together[3].

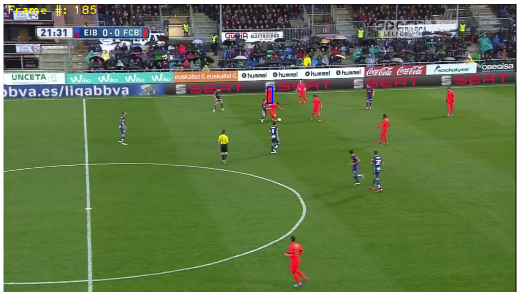

Figure 12: The track switched to another player

Figure 13: The track switched because another player had higher probability of matching to the histogram of the original track

4.2 Result on BYU data

The result on the BYU video sequence showed clearly the upside and downside of mean shift tracking. Figure 14-16 shows when the mean shift tracking can work well. Since an object with distinct color was selcted, there is only one distinct white clutter in the backprojection image which the tracker doesn't have any problem following the track.

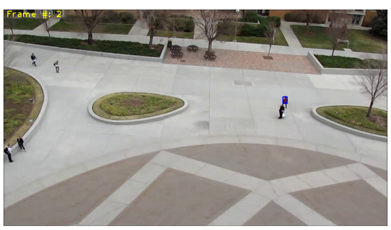

Figure 14: A person with red shirts selected to be tracked

Figure 15: A unique color makes it easy to track this person

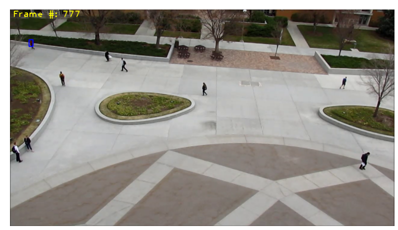

Figure 16: Tracking was successful up to this person going out of the camera view

However, if an object with similar color to the background is to be tracked, the mean shift tracker is not robust and reliable because the backprojection would have multiple regions with high probability of matching to the histogram of the actual track. Subsequently, mean shift is confused in which direction the real track is moving. Figure 17-19 illustrates this scenario.

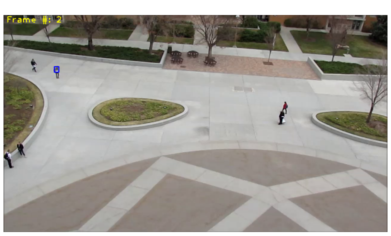

Figure 17: A person with gray shirts selected to be tracked

Figure 18: The track is not distinct in the backprojection

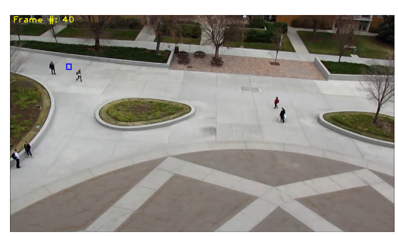

Figure 19: Tracker loses the track soon

4.3 Result on Seoul data

Mean shift tracker can not be considered as a robust tracker on an environment like in Seoul data. Many things can go wrong in many ways. However, this data is used to illustrate what happens on the track when being occluded by other objets. Figure 20-22 shows this situation. When the track(an orange color taxi) is occluded by the utility pole in the middle of the image, the mean shift tracker gets stuck at the right side of the pole even if the track is moving to the left side of the pole.

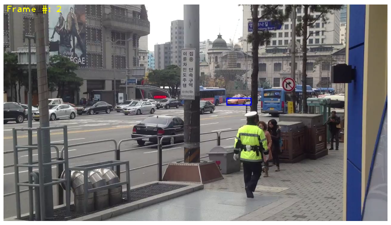

Figure 20: Initial object is selected

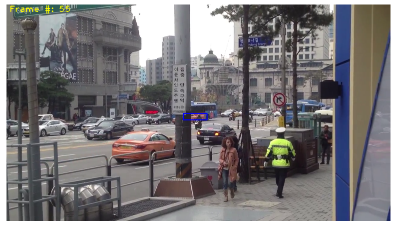

Figure 21: Right before occlusion

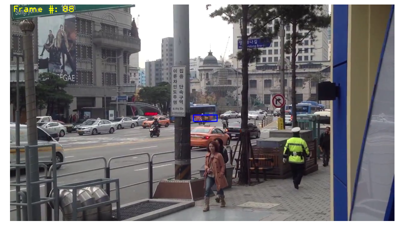

Figure 22: Right after occlusion

5 Conclusion

Mean shift with the backprojection method can be used to track an object in a video stream. The ideal settings for mean shift tracking include the static or slowly moving camera and a distinct color of the object to be tracked. Other than in the ideal settings, mean shift tracking cannot be a robust tracking method. There are more ways that can go wrong than right in unidealized environment such as camera that moves too quick, objects intersecting across each other, selected track changing its size in video sequence, and the occlusion of the track. However, mean shift can be robust if used with other tracking schemes.

References

[1] D. Dementhon. Spatio-Temporal Segmentation of Video by Hierarchical Mean Shift Analysis, 2002. Center for Automat. Res., U of Md, College Park.

[2] D. Comaniciu and P. Meer. Real-Time Tracking of Non-Rigid Objects using Mean Shift, 2000. IEEE CVPR.

[3] H. Zhou, Y. Yuan, and C. Shi. Object tracking using SIFT features and mean shift, 2009. Computer vision and image understanding 113 (2009) 345-352