

# Homework 5

---

Nathan Conroy

April 27, 2017

## 1 OVERVIEW

In this assignment, we were required to test and compare the performance of Tracking-Learning-Detection (TLD) and tracking using a Kalman filter tracker. In order to test this, I took a video of my face and used it as the input to two implementations that I found online. Both implementations created a bounding box around my face (after I specified it as the tracking object) and attempted to stay locked in as I moved around throughout the video.



Figure 1.1: screenshot from original video

## 2 KALMAN FILTER

Kalman filtering is an algorithm which takes a collection of inputs for known measurements or variables that have been observed over time, and uses them to estimate unknown states or variables as time moves on. It is used in the field of machine vision to track the movement of targeted objects in videos from frame to frame as they move within the field of view. The algorithm involves a two step process which includes a prediction step and an update step. During the prediction step the filter creates estimates of the current state of an object given its known variables as well as their uncertainties. During the update step, the next measurement is used to update the previous estimates by calculating a weighted average, with more weight being given to measurements with a higher degree of certainty.

For this task, I used an implementation that I found on GitHub which uses three features to model the tracking object: hue, saturation, and rotation invariant local binary pattern. Because the tracking heavily relies on color attributes, the bounding box stays focused on not only my face but my neck throughout the video.

This implementation seemed to perform a bit less smooth as the TLD overall. The bounding box wasn't as tight to my face, as it expanded to parts of my neck (see figure 2.1). However, the algorithm did seem to handle the rotation of my face better than the TLD, primarily because it was focusing on the hue and saturation of my skin tone and wasn't very disrupted by the event of some of my facial features leaving the frame (see figure 2.2).



Figure 2.1: Kalman filter tracking face and neck

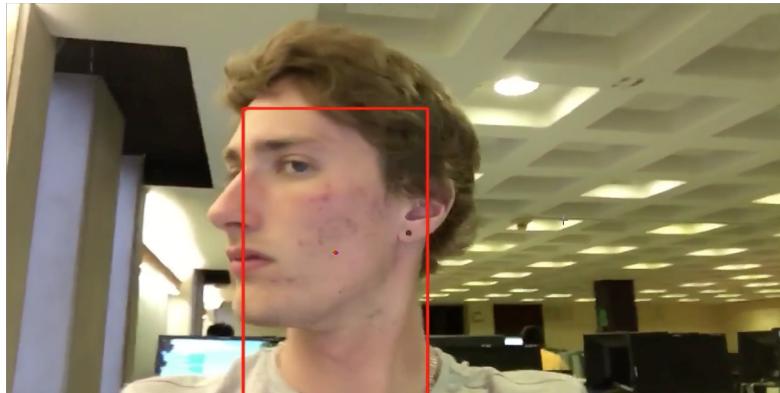


Figure 2.2: tracking my face turned to the side

### 3 TLD

TLD tracking breaks up the process of tracking an object into 3 subtasks which work concurrently. The subtasks are tracking, learning, and detection. Tracking is the process of following an object from frame to frame, the detecting process involves looking at all of the appearances of the targeted object that have happened up to the current point in time and adjusts the tracker if it strays, and the learning process detects errors that have been made by the tracker and uses estimations to avoid these errors going forward.

For this task, I used an implementation that I found on GitHub which uses a recursive tracking algorithm that estimates the location of an array of points within a bounding box that is specified in the first frame from their position in the last frame. The implementation uses an error threshold to decide when a bounding box should be abandoned and reinitialized, such as when there is an occlusion blocking the target object. The detection is implemented by splitting up the video area into many small subwindows which are evaluated using a sliding window approach to check if the subwindow contains the target object. The subwindows are

first discriminated by foreground and background, so that the background area is not within the search space and therefore not considered for evaluation.

The implementation seemed to perform very smoothly as seen in the video. I gave the initial boundaries for my face in the first frame as I launched the program. As my full face stays within the video frame, the boundary box stays locked in with a high level of accuracy as seen in figure 3.1. The only time that the algorithm really "lost" my face was when I turned to the side, and half of my face left the frame (figures 3.2 and 3.3). Although, as my full face re-entered the frame, the bounding box reinitialized and found my face again (figure 3.4).

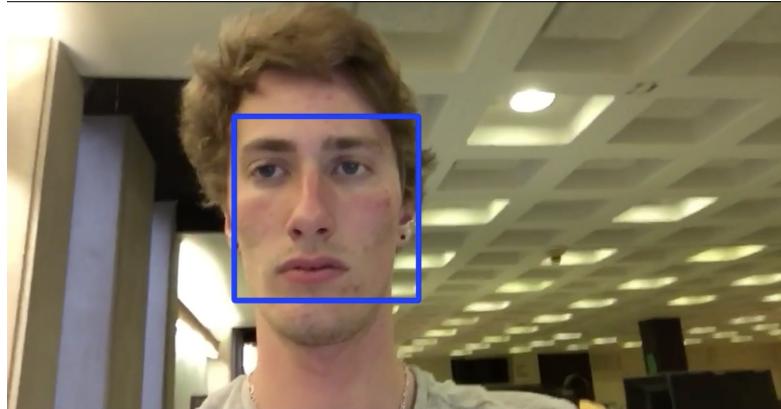


Figure 3.1: TLD tracking my face

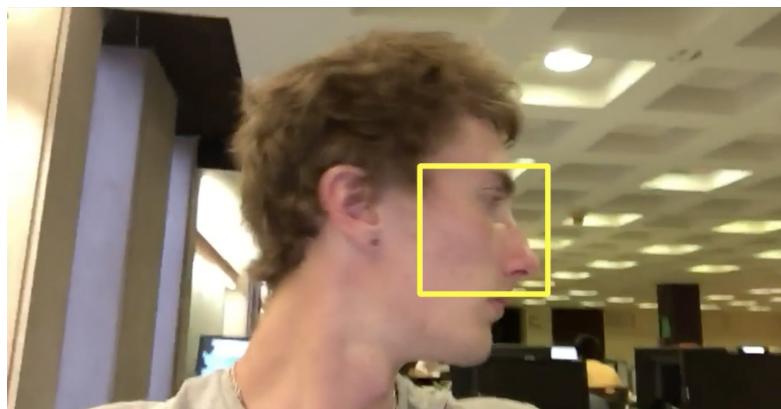


Figure 3.2: tracking is disrupted as I turn my head

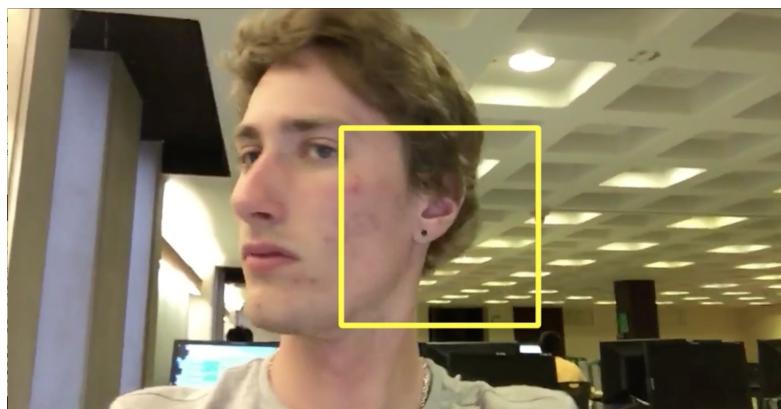


Figure 3.3: tracking is disrupted as I turn my head

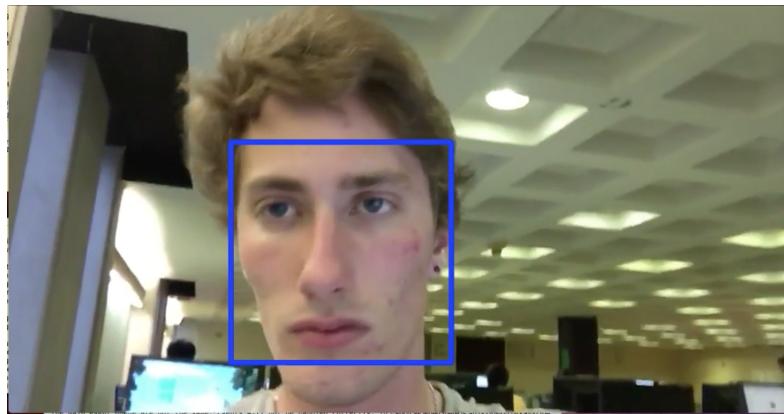


Figure 3.4: TLD reinitializes bounding box, finds face again

## 4 VIDEOS

My recorded videos can be found in the "Videos" directory.

- original.mov – the original video
- kalman.mov – a video of the Kalman filter tracking my face
- tld.mov – a video of the TLD tracking my face

## 5 SOURCES

Cheng, Shaoguang, camshiftKalman, (2015), GitHub repository, <https://github.com/shaoguangcheng/-camshiftKalman>

Nebehay, Georg, open\_tld, (2015), GitHub repository, [https://github.com/pandora-auth-ros-pkg/open\\_tld](https://github.com/pandora-auth-ros-pkg/open_tld)