Tracking with Normalized Correlation

Erik G. Learned-Miller

April 23, 2014

In this assignment you will build a simple tracking algorithm using normalized correlation. You will use a grayscale image sequence found on the web. It has 462 images, but you can just use the first 10 to get your tracker working. It's up to you if you want to try to get your tracker working on the rest of the images.

Do the following steps.

1. Download the images from here `http://vision.ucsd.edu/~bbabenko/project_miltrack.shtml`. Use the "David Indoor" sequence.

2. Use matlab to load in the first image and plot it on the screen. Use the command `impixelinfo` to allow you to figure out the coordinates of a box around the face that you want to track.

3. You can hand code the coordinates of the box around the part of the image you want to track. I recommend using a patch whose dimensions are odd (e.g. 21x35) so that it is easier to interpret the output of the filter2 command (see below).

4. By using this rectangular patch as a filter in the `filter2` command, you can obtain the *dot product* of the target with each patch of the image. Use the method I described in class to convert this output to a *normalized correlation output*.

5. Find the coordinates of the largest value in the normalized correlation. This is your new estimate of the position of the target.

6. Once you have found the face in the next frame, reset the "tracked object" to be the patch from the position you found in the second frame, and then repeat this process. This will be called your basic tracker.

7. Show the results of your basic tracker by showing an image of each frame with a rectangle around the current position from your tracker. Obviously, if your tracker is working, the rectangle should stay on the face in each frame. *It is not essential for your tracker to work throughout the entire video, but it should work for a least a few frames.* Make a .pdf document showing the results of your tracker.

8. Once you have finished your basic tracker, try modifying your tracker in the following way. Instead of replacing the tracked object in each frame with a patch from the current image, take a weighted average of the old tracked object and the new tracked object. That is:

$$t_{new} = \alpha \times t_{old} + (1 - \alpha) \times t_{currentFrame}.$$

Experiment with different values of α.

9. Show results for at least 2 different values of α. Of course, if you set α to 0, you will recover your basic tracker, so don't show results for $\alpha=0$ in this portion of the assignment.

Turn in your code, and your results for the basic tracker and the modified tracker in a single .pdf file.