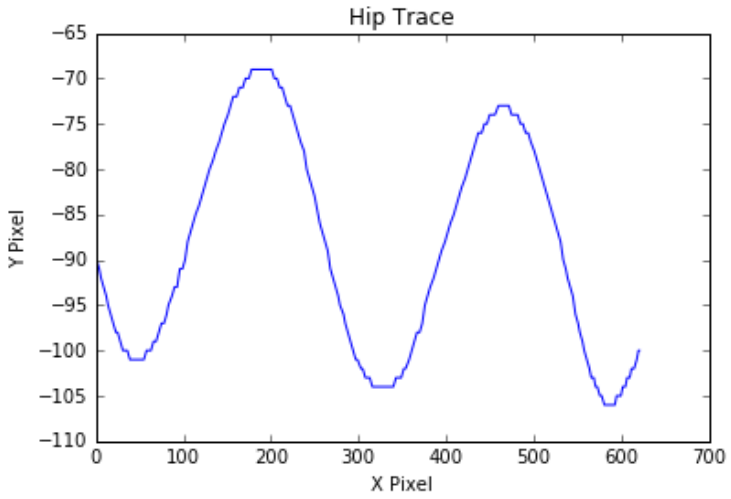
PIC 16, Winter 2018 – Assignment 9F

Assigned 3/9/2018. Code (a single `.py` file) due by the end of class 3/14/2018 on CCLE. Hand in a printout of this document with the self-assessment portion completed by the end of class on 3/14/2018.

In this assignment, you will track the location of a human hip during running using OpenCV template matching.

Task

1. Try the Template Matching tutorial using the provided `frame.png` and `hip.png`. For the next part, choose a single template matching "method" that finds the template `hip.png`.
2. Create a `VideoCapture` object for the video `RyanRun.mp4` and in a while loop, read each frame, convert it to greyscale, find the hip, and draw a rectangle around it as for the single frame in part 1. In other words, play the video and track the location of the hip throughout. As in the previous video playback examples, you'll want to use `imshow` instead of `plot` now, you'll have to use `waitKey`, etc…. Try other template matching methods if the one you chose in the last part is not very reliable (but a few glitches is OK).
3. The hip doesn't actually appear in the first ~870 frames of the video, so these frames aren't of interest. Instead of `read`ing every frame, it is possible to separate the actions of `grab`bing the next frame from actually `retrieve`ing the information in it (see VideoCapture documentation). `grab` the first 870 frames without actually `retrieve`ing or displaying the frame, and then start actually `retrieve`ing and displaying the frames. Likewise, you can break the loop after frame 1082 to avoid showing more video than necessary.
4. Store the *x* and *y* (pixel) locations of the hip (the top left of the location of the template is fine) in a list, and plot the data (the "trace" of the hip) after closing the movie. Reverse the *y*-axis so that up in the plot corresponds with up in the video. You should get something like:



5. Depending on which method you used, there may be a few frames in which the template matching thinks the hip is in the wrong place; i.e. the tracked feature appears to jump. A simple way to avoid this is to look for the template only near the location of the template in the previous frame. Incidentally, this reduces the template matching time substantially. Implement this idea.

Self-Assessment

Print the assignment document and check off the steps you completed successfully. (20pts each)