Homework 4: Object Tracking

18799-K Cognitive Video

03/27/14

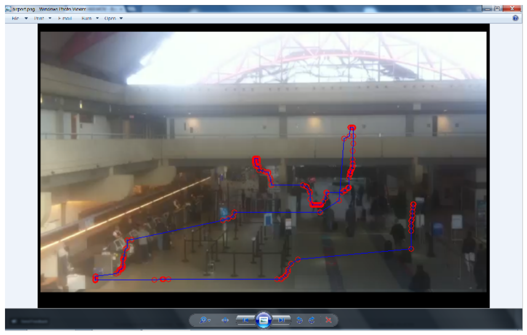
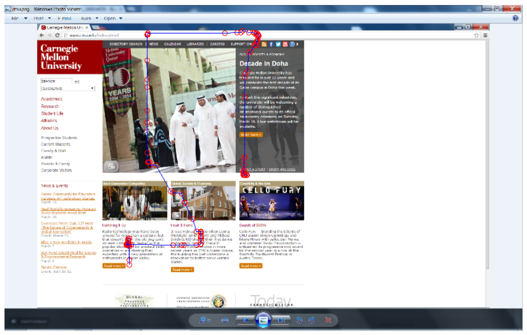
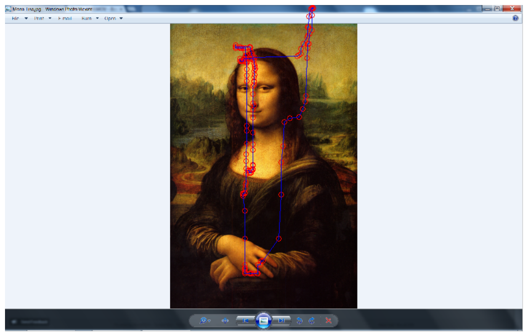
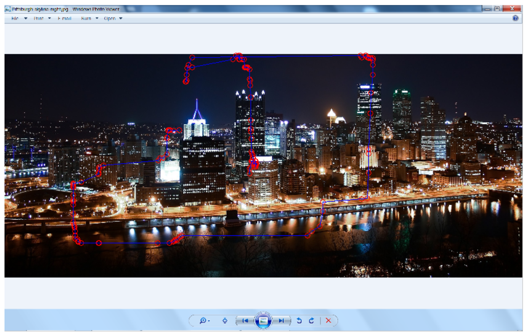
Ranika Kejriwal (rkejriwa)
Jessica Lo (jlo1)
Preeti Singh (preetisi)

*See Note at bottom of page to run the code

A. Download the images and eye gaze datasets. Plot the saccadic gaze points on each image and link them with lines.

Initially, we upload all the the csv files and image files. We simply have a for loop that goes through each image and its corresponding csv files. We show the image and hold it. We then have another for loop that goes through each loop in the csv file. We first check to make sure that left found and left calibrated are both TRUE. If they are not, then we discard that row. If the row is valid, then we obtain the X & Y points from the CSV file. We convert them to doubles and then plot them onto the image. Additionally, we also store the X & Y value points, so then we can plot the lines with the useful points. Finally we just write out this frame that contains all of the points that are also linked with lines.

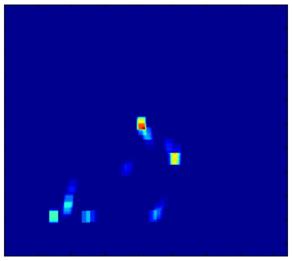
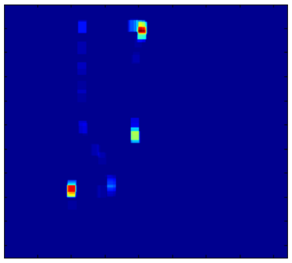
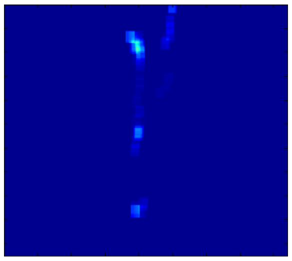
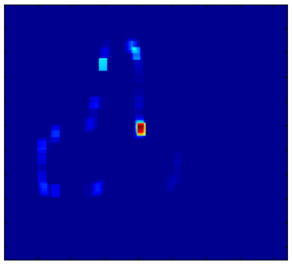
Below are the images we created. We also attached them in the zip file.

B. Based on above, plot the heat map based on gazing durations (the longer gaze, the 'hot' the color).

Using a similar approach as in part A, we go through each point in the csv file. However, instead of plotting the point, we add to an empty matrix of the same size as the image. The longer the duration of the gaze, then the larger the value of the area in the matrix. For this problem, we used a 50 by 50 box, hence our results looked more box-like.

Below are the heat maps we created in this section. We also attached them in the zip file. They are listed in the order of the image number from left to right.

C. Animate the four saccadic eye movement sequences and save each animated sequence in a video file.
Again, we worked on this part based on the results from part A. This time, we took into account the time difference and wrote extra frames to the video for points where the gaze was held longer. Furthermore, we couldn't simply draw all the lines afterwards like we did in part A. Hence, we did things in the following order for each point: find the point, draw the point, check if there was a prior point to draw a line to current point. By the end of the program, we get a separate video file for each of the input csv file. You will notice that points with longer gaze will be reflected in the video.

The video files of the results of part C are included in the zip folder. We included the video files that were resized by a factor of 10. If you'd like more clear video files, email us. We resized this since the video file was too large.

D. Based on above, animate the Left Pupil Diameter (D) and save each sequence to corresponding video file. Discuss the findings.

For this part, we once again iterate through all the CSV files. We also do the set up for all the output video files within this for loop. Within the loop, we first check to make sure that left found and left calibrated are both TRUE. If they are not, then we discard that row. We then extract the value for the left pupil diameter. We ensure that the value is not "-1". If it is not, then we store that value in a vector. For that left pupil diameter we also extract the time that is it held for. Using these two values we plot the size of the pupil diameter (increased by a factor of 10) and we leave it on the screen for the time specified. We are consistently writing these frames to the video and we output 4 different video files at the end of the run.

The video files of the results of part D are included in the zip folder.

*Note regarding running the source code for all parts:
No argument is needed for any of the methods. Instead, be sure the data of the csv files and the image files are in a folder "Assignment4-data" within the working directory. The scripts pull the files from that data. Also, ensure there exists a folder called "hw4results" for the output images and video files.