Curve Fitting on Wheel/Mouse Sequences

Ömer Sümer, Heidelberg University (HCI/IWR). **08 June 2015**

Methodology and Experimental Setup:

We are given a set of video sequences which are recorded on the wheel walk of a mouse, and the objective is to find the outer curve of wheel. Here, the methodology which is used will be described briefly. (We do not give the details of methods here, because it is straightforward and nothing new in this scope.) The test videos are of Mouse ID 1, 4 and 5. The names of trials and filenames are below:

- Mouse1/Trial_201411111614571
- Mouse1/Trial_20141020191115117
- Mouse4/Trial_201411122034231
- Mouse5/Trial_2014102103300061

All video files are extracted at 25 *fps* and each consist roughly 500 frames. In each frame, we first applied adaptive histogram equalization and gaussian smoothing with $\sigma = 2$.

There are small circles near to the surface of wheel. Thus, it is comparatively easier to find the surface that mouse walks back and forth. We used hough transform-based circle fitting with the radius range 10 to 15.

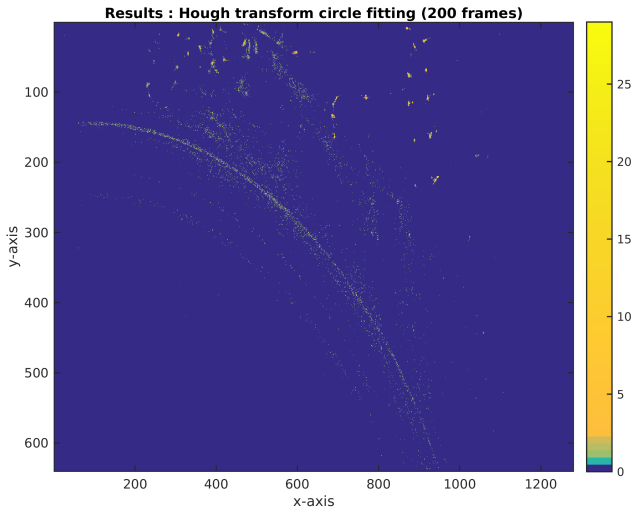


Figure 1: Hough transform circle detections ID:Maus1 Trial:201411111614571

For a single frame, Hough transform is not enough to detect this circles and fit a curve, however it can give related information when it is applied on large number of similar frames. We could say that other reasons of detections come from mouse parts or random noises in specific coordinates, but the ones from the wheel must be consistent, because the appearance of wheel (little circles) are same over all the video. Thus, it could be interesting to see the distribution of these detections on frame. Figure 1 shows the circle detections over 200 frames, and it is observable that we can fit a large circle on this points.

We used all frames of each video (nearly 500 frames) because our aim is to find the curvature of wheel more accurately for future works on these sequences, instead of finding with a few frames or faster. In this step, there are various approaches to find the curve of wheel, such as least squares, hough voting, or RANSAC. We prefer using RANSAC fitting on a circle equation (We did not use LS, it sometimes fits on a very far region from the wheel even median LS fitting is used). Here, the important point is to give initial/possible ranges of center point (x_0, y_0) and radius (r).

The range of position does not change too much so we gave the same interval. On the other hand, the radius could change due to the camera distance. RANSAC fitting takes only 1-2 seconds, and it is not difficult to change upper bound of radius manually. It fits accurately after a few trials if it does not work in first one. [1]

Results:

In following pages, it could be seen the results of each sequence.[2] Because we extracted Hough transform detections in each frames in order to get more accurate results, it takes a few minutes for each video sequences. Total number of detected circles are between 20,000 and 50,000. RANSAC fitting with these points takes only 2-3 seconds.

Initial values and variables:

Image/frame size : 680×1280

(Initial) center of the wheel = [-500:+500, 1100:2500]

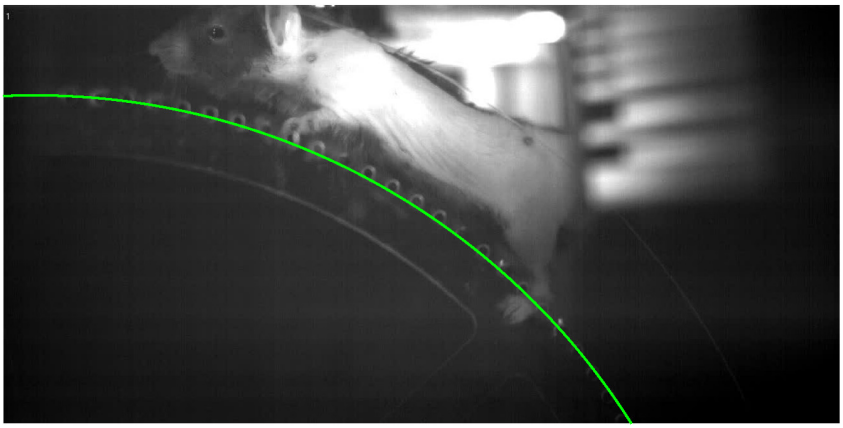
The radius of the wheel = [500,1000][3]

RANSAC threshold = 10 and # of iteration = 1000 or 3000

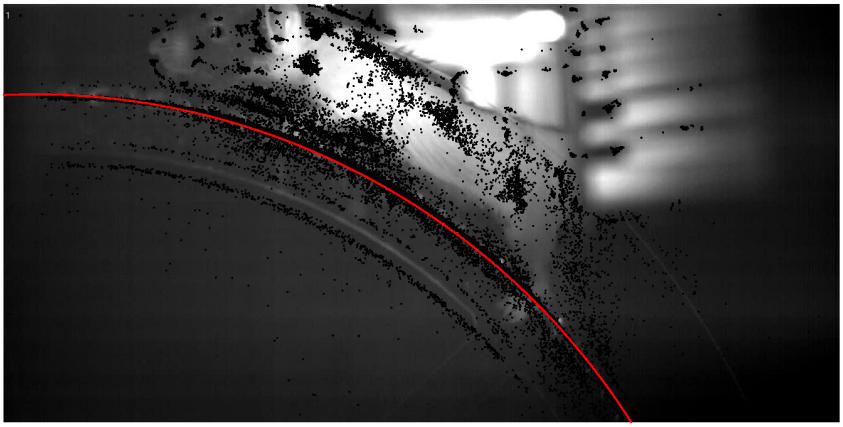
[1] Here, I did not increase the sweep interval in radius, because it gives more variant and bad results in fitting if the given range is too wide.

[2] Figures on the following pages are plotted on the first frame on each sequence to see whether the predictions are accurate or not.

[3] If it does not converge to real/observable position of the wheel curve, it is enough to try less/greater outer value depending on the situation.

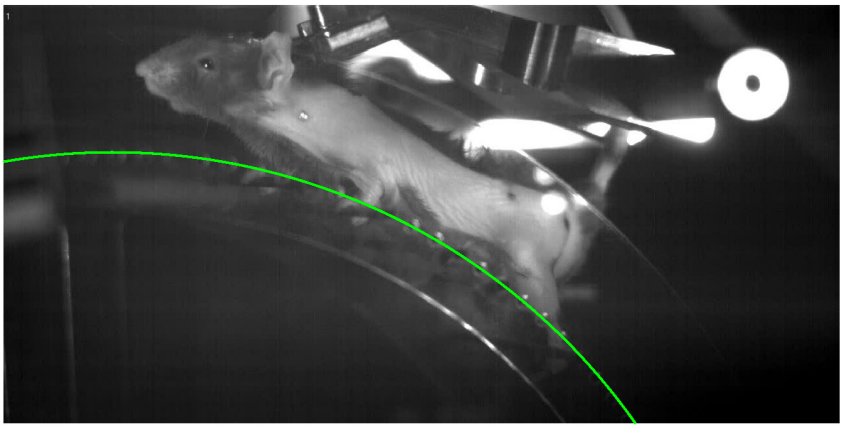


(a) Fitted line

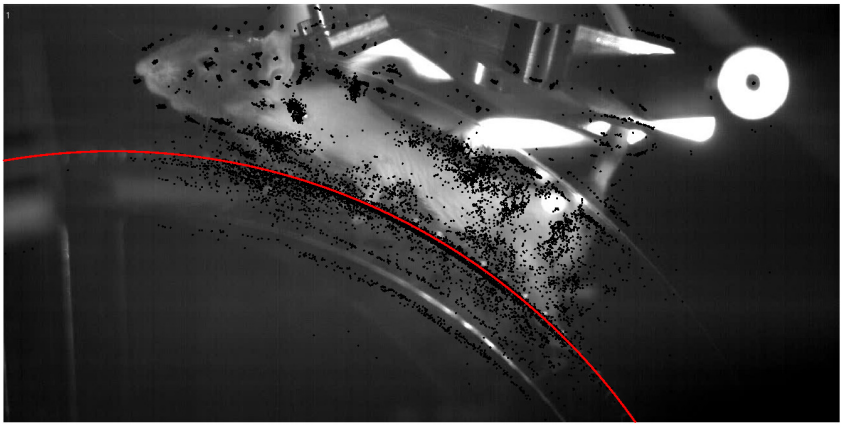


(b) Fitted line with circle detections

Figure 2: Mouse1/Trial_201411111614571

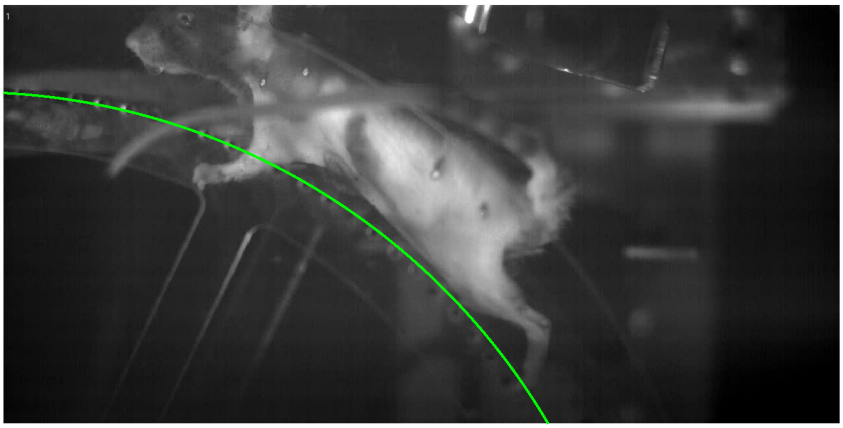


(a) Fitted line

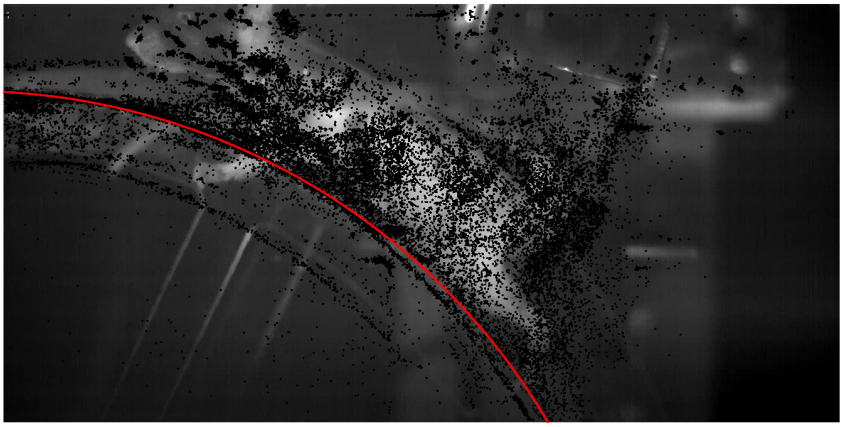


(b) Fitted line with circle detections

Figure 3: Mouse1/Trial_20141020191115117

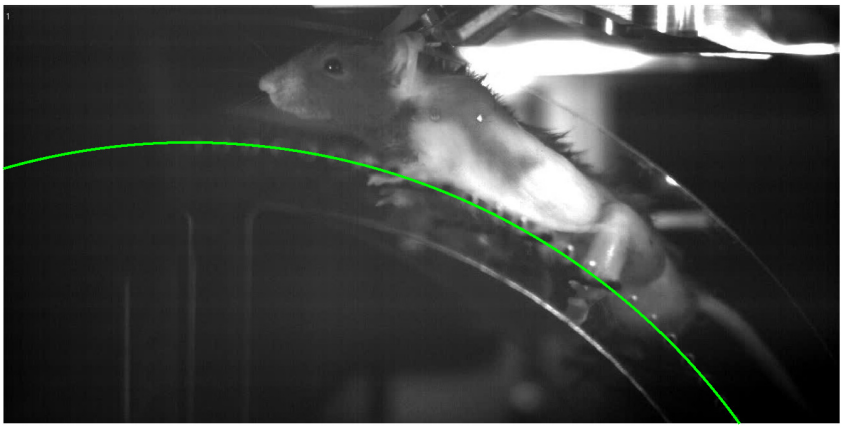


(a) Fitted line

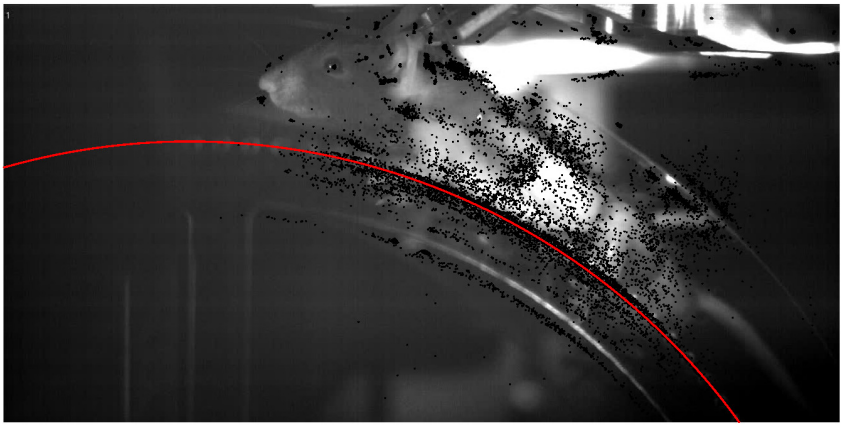


(b) Fitted line with circle detections

Figure 4: Mouse4/Trial_201411122034231



(a) Fitted line



(b) Fitted line with circle detections

Figure 5: Mouse5/Trial_20141021033000061

Notes:

- I tried only on these four sequence. among them, three sequences are blurry and hough detection works fairly good even under these circumstances.
- Alternatively, I investigated the maximum deviation from optical flow in opticla flow direction as scanning a thin (a few pixels width) horizontal patch on very left side of image, and vertical thin patch on the lower part of image. So, finding only two points in enough to predict a circle of wheel. Left point crossing the vertical axis can be detected with very good accuracy, however the other (lower right) point is noisy due to distortion of the parts of mouse in this region. I did not send the figures of this method, because of poor results.
- Sometimes, the initial ranges of center point and radius could be tricky. But, I think that increasing the number of iteration and try narrow intervals of radius is helpful, if it did not converge in the first try.