Mika Thormann 5465950

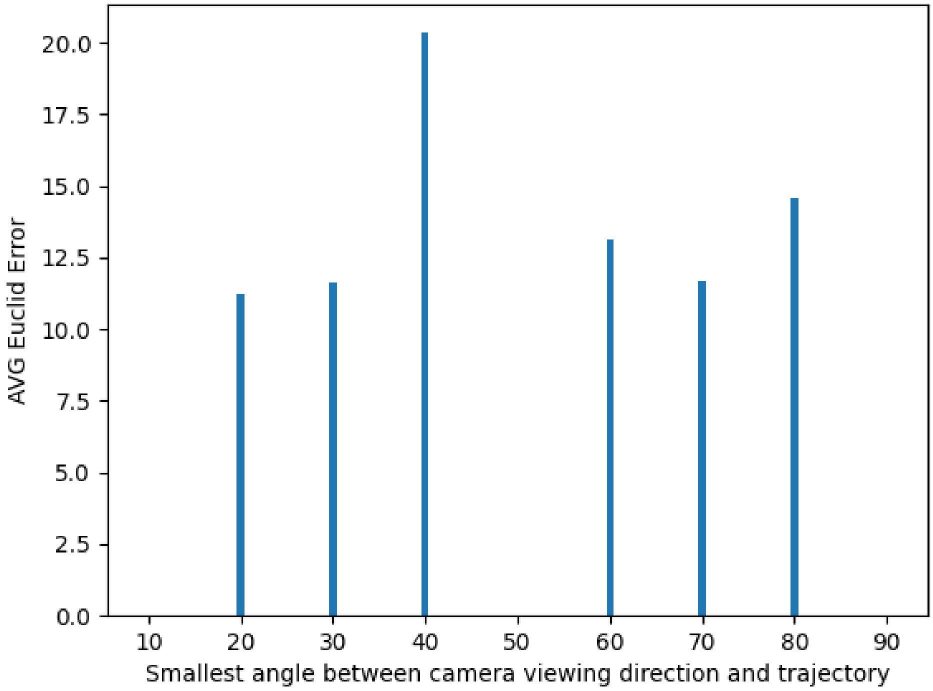
Progress report for week 7-10

Task:

Design a benchmarking tool for evalution

Implementation:

As there still is no ball tracking tool that I can use, I had to generate trajectories with the simulation I made. These 3D coordinates are projected on the image as 2D coordinates. Using the estimated camera poses. I generated multiple simple projectiles with different angles, to cover a fairly wide variety of shots. I can extract the euclidean error between the original simulation data and the estimations made by my program. These are evaluated in the evaluation.py script, generating plots, describing how well the estimations are from different camera angles in relation to the table and the ball trajectory. This is done by simple bar plots indicating the error of 10° angle regions from 0 to 90 degrees(horizontal and vertical camera angle / camera to trajectory angle). Following you can see an example of my plots. Regions that were not present in the evaluation have no bar.

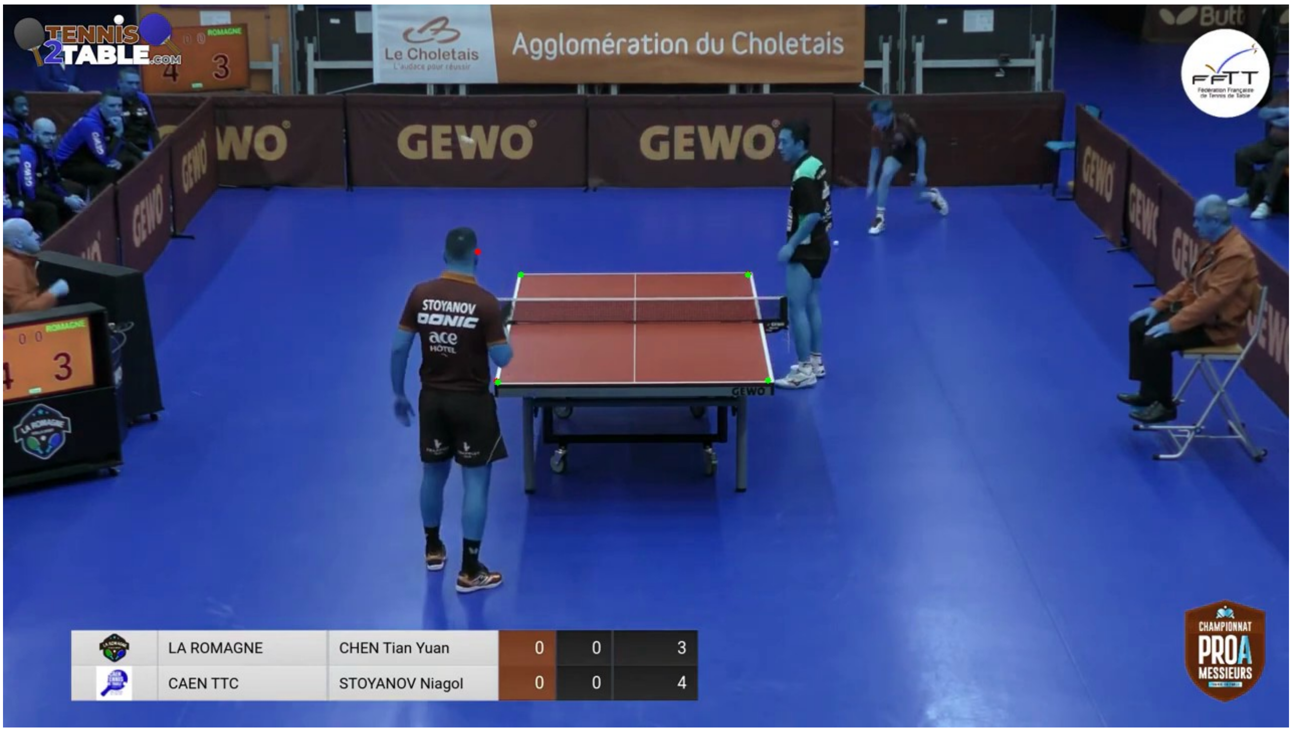


Other things I did:

- fixed the problem of big blue areas in the background being problematic
- fixed the input order for corners in the pose estimation for some cases
- bugfixes

Problems:

- Images made in a close to 90° angle to a tables edge result in bad approximations:
  - How to proceed with these? Keep or discard?
  - If I should keep them I need to change the way I sort out bad approximations



Missing things:

For now the script only works in cases, where the ball hits the table 2 times(serve). To make it work in an ongoing match with racket hits and only one bounce per hit, I have to detect those racket hits and approximate their position with the current trajectory plane. Then i can approximate their x and y positions on the table plane and create a new trajectory plane. To make this work I have to detect every bounce and hit(event). If there is an interruption in pose estimation or an event is not detected for some reason, the estimation has to be stopped, as everything builds up on the previously generated trajectory plane. Another problem with this is that estimations can in theory get worse and worse as a set goes on, as the planes are not estimated perfectly and used to generate further planes.