Advanced tracking

Ivan Nikolov

I. Introduction

This assignment is consisted of two parts. In the first part we implemented the Kalman filter with three different motion models (NCV, NCA and RW) and tested the Kalman smoothing on synthetically generated trajectories.

For the second part, I implemented a Particle filter which uses the motion models from above as a dynamic model and a color histogram as a visual model. The different tracker parameters were evaluated on the VOT2014 dataset.

II. Experiments

A. Motion models and Kalman filter

First, we defined the state and system matrices and derived them using the `sympy` package. The results are shown in the Appendix section.

The Kalman filter smoothing was then applied to the spiral and two more trajectories.

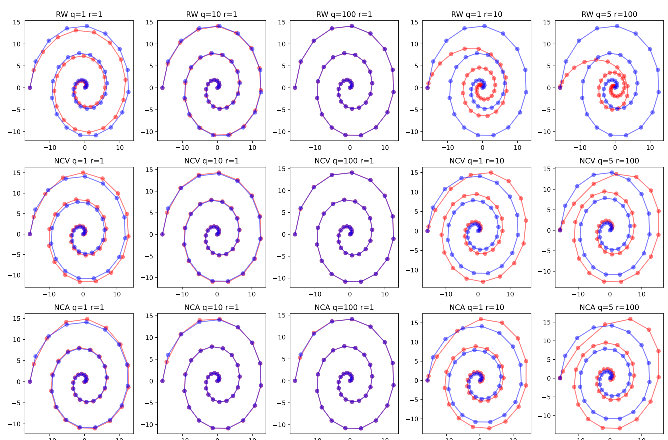

Figure 1: Kalman filter with different motion models and q and r parameters. Blue color denotes the GT, while the red color denotes the approximated trajectory.

On Figure 1 we can see filter responses for the spiral trajectory. Each row denotes a different motion model (RW, NCV, NCA) accordingly and different q and r values for the uncertainty in the motion and observation models. Based on the results, the filter performs best with $q = 100$ and $r = 1$. This means that there is a large uncertainty in the motion model and small in the observation model. With an increase of the uncertainty of the observation model, there is a degradation in the performance of the filter, which can be noticed visually. This is logical, because our observation model is a direct copy of the positions of the points.

We did further evaluation on two more jagged trajectories. On Figure 2, a trajectory having sudden changes and a saw-tooth like shape is shown. And a rectangle path shown on Figure 3. The results we got on these two paths further confirm the initial conclusions. In the scenarios where there is large noise in the observation model, when the points follow some smooth trajectory, the NCV and NCA models perform better than the RW model (last column in Figure 1). In addition, on Figure 2 we can see that NCA responds better to suddent changes than NCV.

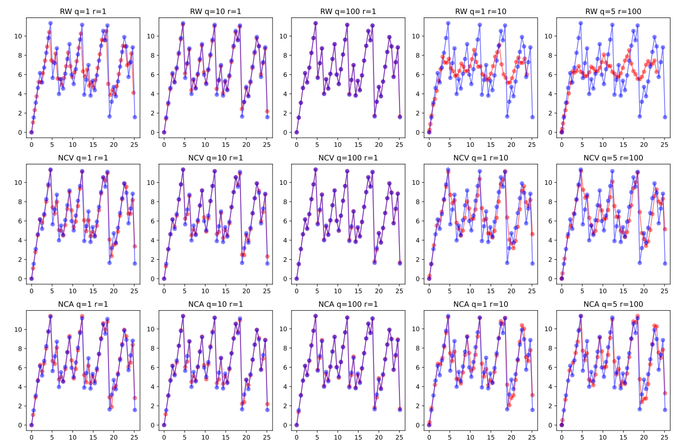

Figure 2: Kalman filter with different motion models and q and r parameters. Blue color denotes the GT, while the red color denotes the approximated trajectory.

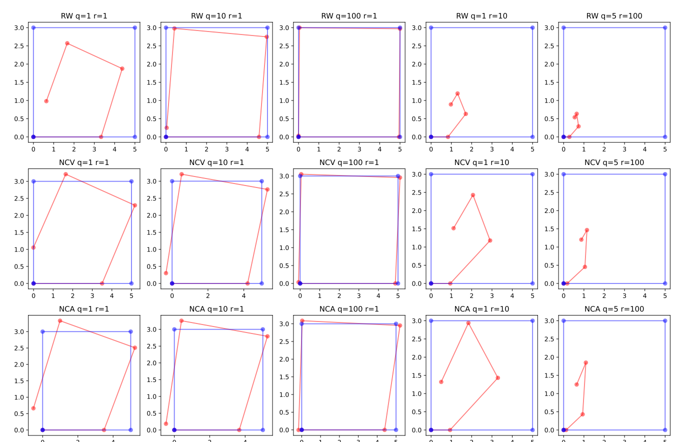

Figure 3: Kalman filter with different motion models and q and r parameters. Blue color denotes the GT, while the red color denotes the approximated trajectory.

B. Particle filters

Here, we did the tuning of the parameters by manually testing multiple parameters and trying to achieve as high as possible robustness and overlap, while still maintaining a solid FPS (>50).

First we tested the performance on sequences from VOT2014 where the previous trackers performed worse. Based on the results on Table I, the Particle filter tracker outperforms the simpler trackers (especially in the number of failures). Here, we set the number of particles to 100, used a NCV dynamic model, set q to 0.1, r to 1, σ for transforming the distances to probabilities to 1. For the visual model, we used RGB histogram with 16 bins and scaled the patches to 0.8 of their original size.

Sequence	AO	Failures	FPS
fish1	0.30	4	90
bolt	0.55	0	76
david	0.52	0	81
ball	0.43	0	95
diving	0.33	2	86

Table I: AO, Faliures and FPS on five sequences from VOT2014

We can see that NVC fails to model the fast changes of direction of movement of the fish sequence, and has low AO and a higher number of failures.

N	AO	Failures	FPS
50	0.40	36	147
100	0.39	28	85
150	0.39	23	56

Table II: Comparison of particle filter tracker with different number of particles (N).

We also compared the performance of the tracker under different number of particles. An interesting observation is that despite chaining N, the AO had no significant changes. The parameter specification is the same as the one described for the video sequences above. Noticeable change is observed in the FPS performance, as for more particles we need more computations. This can be optimized if we parallelize this part. With more particles, it seems that the tracker becomes more robust, as the number of failures drops.

III. Conclusion

In this assignment, we implemented the three dynamic models and tested them with the Kalman filter. We also implemented the Particle filter tracker and evaluated it on the VOC2014 dataset. This filter outperforms the other filters implemented so far for this course. This is because it uses additional information from the dynamic models and does not rely only on the visual model. Additional improvements can be made in implementing additional visual models or testing out different color spaces for the histograms.

IV. Appendix

A. Nearly constant velocity

$$X = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{bmatrix} F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Phi = \begin{bmatrix} 1 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} \frac{\Delta T^3 q}{3} & \frac{\Delta T^2 q}{2} & 0 & 0 \\ \frac{\Delta T^2 q}{2} & \Delta T q & 0 & 0 \\ 0 & 0 & \frac{\Delta T^3 q}{3} & \frac{\Delta T^2 q}{2} \\ 0 & 0 & \frac{\Delta T^2 q}{2} & \Delta T q \end{bmatrix} L = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

B. Random walk

$$X = \begin{bmatrix} x \\ y \end{bmatrix} F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \Phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} L = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} \Delta T q & 0 \\ 0 & \Delta T q \end{bmatrix} H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

C. Nearly constant acceleration

$$X = \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ y \\ \dot{y} \\ \ddot{y} \end{bmatrix} F = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\Phi = \begin{bmatrix} 1 & \Delta T & \frac{\Delta T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & \Delta T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{\Delta T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & \Delta T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} \frac{\Delta T^5 q}{20} & \frac{\Delta T^4 q}{8} & \frac{\Delta T^3 q}{6} & 0 & 0 & 0 \\ \frac{\Delta T^4 q}{8} & \frac{\Delta T^3 q}{3} & \frac{\Delta T^2 q}{2} & 0 & 0 & 0 \\ \frac{\Delta T^3 q}{6} & \frac{\Delta T^2 q}{2} & \Delta T q & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\Delta T^5 q}{20} & \frac{\Delta T^4 q}{8} & \frac{T^3 q}{6} \\ 0 & 0 & 0 & \frac{\Delta T^4 q}{8} & \frac{\Delta T^3 q}{3} & \frac{\Delta T^2 q}{2} \\ 0 & 0 & 0 & \frac{\Delta T^3 q}{6} & \frac{\Delta T^2 q}{2} & \Delta T q \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$