Exercise 4: Advanced tracking

Veljko Dudić

I. Introduction

The assignment is divided into two main sections. In the first section, a Kalman filter was implemented using three distinct motion models (NCV, NCA and RW) and Kalman smoothing was applied to synthetically generated trajectories. In the second section, a particle filter was constructed that employs the aforementioned motion models as its dynamic component and utilizes a color histogram as its observation model. Various tracker parameter configurations were then evaluated on the VOT2014 dataset.

II. Experiments

A. Motion models and Kalman filter

The state and system matrices were first defined and symbolically derived using the SymPy package, with the detailed derivations presented in the Appendix. Kalman filter smoothing was then applied to the spiral trajectory as well as two additional trajectories.

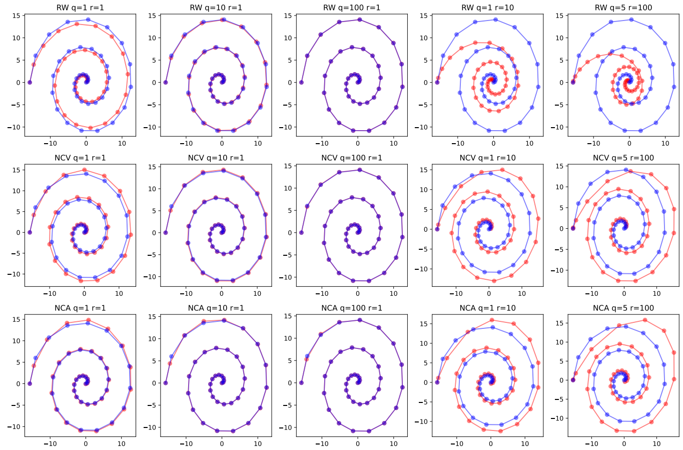

Figure 1: Kalman filter applied using different motion models with specified process (Q) and measurement (R) noise parameters. The ground-truth trajectory is shown in blue, and the estimated trajectory in red.

Figure 1 displays the filter's output for the spiral trajectory. Each row corresponds to one of the motion models (RW, NCV, NCA), with varying process noise (q) and measurement noise (r) settings. The best performance is achieved with q = 100 and r = 1, indicating high uncertainty in the motion model and low uncertainty in the observation model. As r increases, the filter's accuracy visibly deteriorates, which is expected since the observation model directly uses the point positions.

Further evaluation was conducted on two additional jagged trajectories. Figure 2 depicts a path with abrupt changes and a sawtooth pattern, while Figure 3 shows a rectangular trajectory. The outcomes for these two paths reinforce the earlier findings. Under conditions of high observation noise on an otherwise smooth trajectory, the NCV and NCA models outperformed the RW model (as seen in the last column of Figure 1). Moreover, Figure 2 illustrates that the NCA model adapts more rapidly to sudden changes than the NCV model.

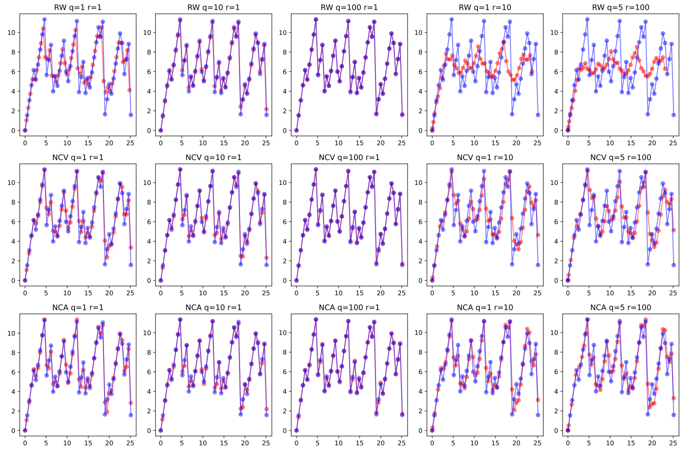

Figure 2: Kalman filter applied using different motion models with specified process (Q) and measurement (R) noise parameters. The ground-truth trajectory is shown in blue, and the estimated trajectory in red.

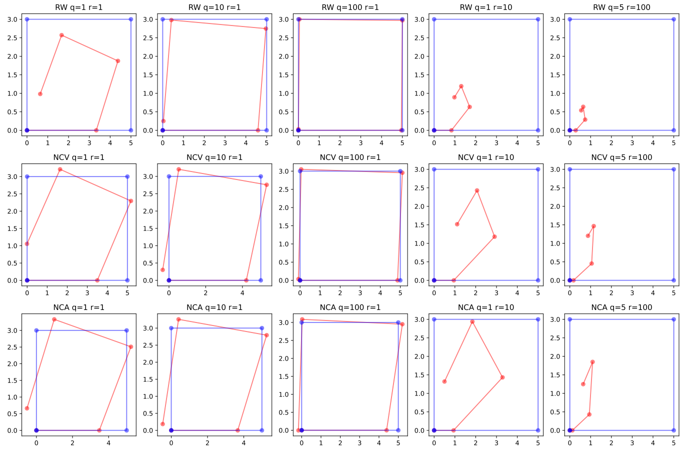

Figure 3: Kalman filter applied using different motion models with specified process (Q) and measurement (R) noise parameters. The ground-truth trajectory is shown in blue, and the estimated trajectory in red.

B. Particle Filters

Parameter tuning was carried out through manual experimentation with various settings to maximize robustness and overlap, while ensuring a frame rate exceeding 45 FPS.

Performance was evaluated on VOT2014 sequences in which the simpler trackers showed poor performance. As reported in Table , the particle filter tracker achieves superior results—most notably in reduced failure count. The tracker was configured with $N = 100$ particles, an NCV dynamic model, process noise $q = 0.1$, measurement noise $r = 1$, and a distance-to-probability scaling factor $\sigma = 1$. The visual model employed a 16-bin RGB histogram, and image patches were resized to 80% of their original dimensions.

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

Sequence	AO	Failures	FPS
fish1	0.30	4	52
bolt	0.55	0	38
david	0.52	0	41
ball	0.43	0	55
diving	0.33	2	43

The NCV model is unable to capture the rapid direction changes in the fish1 sequence, resulting in a lower AO and a higher number of failures.

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

N	AO	Failures	FPS
50	0.41	35	89
100	0.42	27	47
150	0.42	24	34

Tracker performance was evaluated across different particle counts. Interestingly, the AO remained essentially constant as N increased. All other parameters matched those used in the earlier video-sequence experiments. In contrast, FPS dropped noticeably with larger particle sets, due to the extra computation—a cost that could be mitigated by parallelizing this stage. Nevertheless, robustness improved with more particles, as indicated by a lower failure rate.

C. Particle filter tracker using different motion models

Table III: AO, Failures and FPS for different motion models and process noise values (q)

Model	q	AO	Failures	FPS
NCV	0.1	0.42	27	47
NCV	0.5	0.44	29	54
NCV	1	0.44	30	48
NCV	5	0.45	29	54
NCV	100	0.46	179	52
NCA	0.1	0.44	146	51
NCA	0.5	0.45	209	44
NCA	1	0.46	232	41
NCA	5	0.47	375	43
NCA	100	0.44	930	34
RW	0.1	0.39	73	55
RW	0.5	0.43	42	47
RW	1	0.43	33	49
RW	5	0.45	28	49
RW	100	0.44	126	53

For the NCV model, the smallest process-noise scale $q = 0.1$ yields the best performance—larger values of q lead to more frequent failures, even though they can marginally improve FPS. In the NCA model, increasing q consistently degrades both robustness (failures), and at $q = 100$ the added computational burden causes FPS to drop. The RW model benefits from increasing q up to about $q = 5$, beyond which failures rise.

For all models "generally", the average overlap (AO) gets bigger with higher q-s.

III. Conclusion

In this assignment, three dynamic motion models were implemented and evaluated using a Kalman filter. A particle filter tracker was then developed and tested on the VOT2014 dataset. By combining information from the dynamic models with a visual model, the particle filter outperforms the previously implemented filters in this course. Future enhancements could include integrating additional visual models or experimenting with alternative color spaces for the histograms.

IV. Appendix

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

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

Figure 4: Nearly constant velocity.

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

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

Figure 5: Random walk.

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

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

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

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

Figure 6: Nearly constant acceleration.

References