Exercise 4: Advanced Tracking

Sara Bizjak

I. Introduction

Tracking various objects can play a huge role in many different fields. Tipically, CNN based trackers have very good performance and are very popular due to their simplicity. One such tracker is particle filter tracker, which can be a powerful technique for tracking deformable objects in image sequences with complex backgrounds. In this report, we first present results after implementing motion models, more specifically – random walk, nearly-constant velocity and nealy-constant acceleration – all using Kalman filter. Then we present how particle filter performs on sequences from VOT 2014 using `pytracking-toolkit-lite`.

II. Experiments

A. Motion models and Kalman filters.

We first implement motion models using Kalman filter. We test the implementation for different parameters q and p on few different curves. The result can be seen on Figures 1, 2, 3, 4.

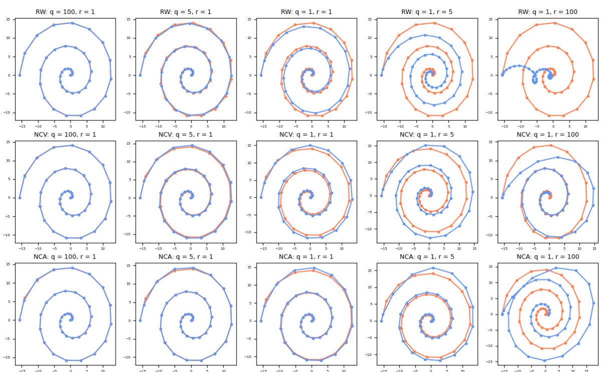

Figure 1. Kalman filter, example 1 – reference figure.

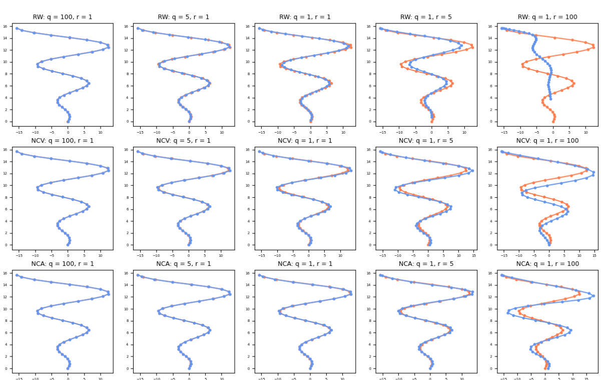

Figure 2. Kalman filter, example 2.

We can quickly see that with increasing r the results get worse. On the other hand, increasing q bring better results, as both curves (red and blue) are almost completely overlaping. Thus, in all cases, the choice of $q = 100$ and $r = 1$ is the suitable one.

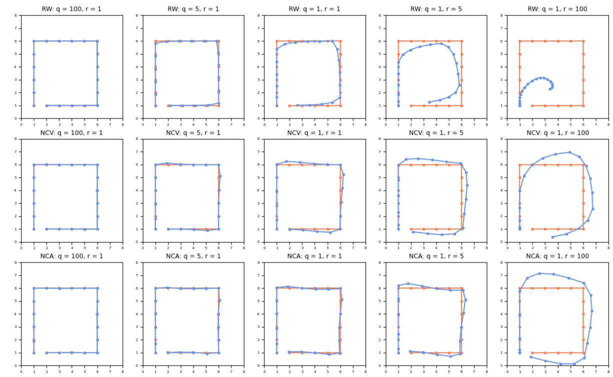

Figure 3. Kalman filter, example 3.

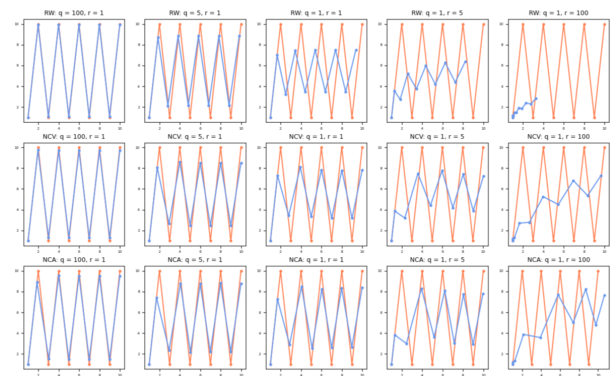

Figure 4. Kalman filter, example 4.

B. Particle filter tracker.

After implementing particle filter tracker, we evaluate it on some sequences from VOT2014. Before reporting the result, we first calibrate parameters, which were for further analysis set as

- $num\_particles = 100$
- $n\_bins = 8$
- $sigma = 10$
- $prob\_sigma = 2$
- $alpha = 0.05$

The parameters can be also seen in the attached file `particle_filters.py` under class `PFParams`.
The averaging result are shown in the Table I and results for few selected sequences are shown in the Table II.

Results for tracker:	
Total failures	119
Average failures	4.76
Average overlap	0.42
Average speed [FPS]	70.3

Table I
Obtained total/averaged results after integration of tracker on VOT2014 sequences, together with chosen parameters

VOT2014 sequence	Failures	Overlap	Average speed [FPS]
basketball	12	0.49	59.4
bolt	5	0.51	66.3
drunk	3	0.42	38.6
gymnastics	0	0.51	65.9
hand1	9	0.33	81.4
hand2	11	0.42	85.7
jogging	1	0.56	72.2

Table II
Some individual results after integration of tracker on some of the VOT2014 sequences.

We first notice that this tracker is quite slower then the one we implemented in the last exercise. Also, for fixed parameters, this tracker makes more mistakes for some (not all!) of the sequences. Thus, it is worth mentioning that for better results with this tracker, we should calibrate parameters manually for every sequence separately.

After testing particle filter tracker with motion model of NCV, we also test the tracker using RW and NCA for parameters $q = 100$ and $r = 1$. The results are shown in the Table III.

Motion model	Total failures	Average overlap	Average speed [FPS]
RW	103	0.44	68.3
NCV	119	0.42	70.3
NCA	136	0.38	70.8

Table III
Affection of choosing different motion models on tracker performance.

We can see that RW outperforms the two other motion models for this parameter setup. Additionally, in the Table IV we observe how calibrating parameter q with fixed $r = 1$ affects the performance of tracker (using NCV notion model).

Parameters (q, r)	Total failures	Average overlap	Average speed [FPS]
(1, 1)	154	0.37	71.6
(5, 1)	144	0.37	70.0
(50, 1)	133	0.41	70.42
(100, 1)	119	0.42	70.3

Table IV
Affection of calibrating parameter q with fixed r.

We can see that with increasing q the results are getting better. That was also noticed when testing different curves with NCV motion model (Figures 1, 2, 3, 4). However, we must be careful with parameter q – when too little, the tracker may fail when the target object is making large movements, and when too large, the particles are spread around the object too much. We then must find a siutable trade-off observing both accuracy and robustness. For even better performance, we could determine the parameter q considering the size of tracked object, so that there would be some relation between them.

Lastly, we observe how the number of particles impact on tracking performance. The results can be seen in the Table V. Results obtained with q and r set to 100 and 1 again, with NCV motion model.

Number of particles	Total failures	Average overlap	Average speed [FPS]
10	144	0.4	408.3
20	131	0.4	269.4
50	125	0.42	129.22
80	127	0.43	86.68
100	119	0.42	70.3
120	118	0.42	61.11

Table V
Affection of choosing different number of particles on tracker performance.

The number of particles is indeed the parameter who affects the tracking performance the most. We see that with increasing the number of particles, the tracker gets significantly slower, however, the number of failures gets smaller, while average overlap stays practically the same. Due to speed limitation for higher numbers of particles, we conclude that it is not worth increasing the number over 100, since the tracker would become almost useless then (i. e. too slow).

III. Conclusion

As a part of the exercise, we implemented three motion models and particle filter tracker. The tracker was tested on several sequences, on which a set of parametrs were set, so that they are overall a good choice. However, to achieve even better performance, it is important to set parameters separately for each sequence. While calibrating with parameters, we noticed that for parameters q and r, the most appropriate choice is $q = 100$ and $r = 1$. We saw that changing the number of particles have the biggest impact on performance, however, we must be careful not to increase it too much, as the tracker may become unsustainable slow. To perform even better performance in terms of accuracy and robustness, we could try using different colorspaces to create color histogram, and then choose the best option. All in all, to achieve the best possible performance, we need to calibrate parameters for sequences separately and and consider whether they are appropriate choice also in terms of tracking speed.

Appendix

- Matrices for Random Walk (RW):

$$\text{state} = \begin{bmatrix} x & y \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \ \Phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

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

- Matrices for Near-Constant Velocity (NCV):

$$\text{state} = \begin{bmatrix} x & y & \dot{x} & \dot{y} \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \ \Phi = \begin{bmatrix} 1 & 0 & t & 0 \\ 0 & 1 & 0 & t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathrm{L} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \ \mathrm{Q} = \begin{bmatrix} \frac{qt^3}{3} & 0 & \frac{qt^2}{2} & 0 \\ 0 & \frac{qt^3}{3} & 0 & \frac{qt^2}{2} \\ \frac{qt^2}{2} & 0 & qt & 0 \\ 0 & \frac{qt^2}{2} & 0 & qt \end{bmatrix}$$

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

- Matrices for Near-Constant Acceleration (NCA):

$$\text{state} = \begin{bmatrix} x & y & \dot{x} & \dot{y} & \ddot{x} & \ddot{y} \end{bmatrix}$$

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

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

$$\mathrm{L} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \ \mathrm{Q} = \begin{bmatrix} \frac{qt^5}{20} & 0 & \frac{qt^4}{8} & 0 & \frac{qt^3}{6} & 0 \\ 0 & \frac{qt^5}{20} & 0 & \frac{qt^4}{8} & 0 & \frac{qt^3}{6} \\ \frac{qt^4}{8} & 0 & \frac{qt^3}{3} & 0 & \frac{qt^2}{2} & 0 \\ 0 & \frac{qt^4}{8} & 0 & \frac{qt^3}{3} & 0 & \frac{qt^2}{2} \\ \frac{qt^3}{6} & 0 & \frac{qt^2}{2} & 0 & qt & 0 \\ 0 & \frac{qt^3}{6} & 0 & \frac{qt^2}{2} & 0 & qt \end{bmatrix}$$

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