Advanced tracking

Radoslav Atanasoski

I. Introduction

This assignment introduces Motion models, the Kalman filter and Particle filters. The assignment is made up of two subtasks, where for the first subtask, the motion models are implemented for Nearly Constant Velocity (NCV), Nearly Constant Acceleration (NCA) and Random Walk (RW). The motion models are tested on three different trajectories using a variety of parameters. For the second subtask, the particle filter is implemented, where the tracker uses the implemented motion models and particles of different weights to track an object. The tracker is further integrated with a provided toolkit for evaluation of the entire VOT14 dataset, using average overlap, number of fails and frames per second (FPS).

II. Experiments

Motion models and Kalman filter

For this subtask three motion models are implemented: RW, NCV and NCA. They can be seen in appendix A.

Each motion model is tested on three different trajectories: spiral, flower-like shape and a sun-like shape. We can see an example of this in the following figures. In figure 1, we can see

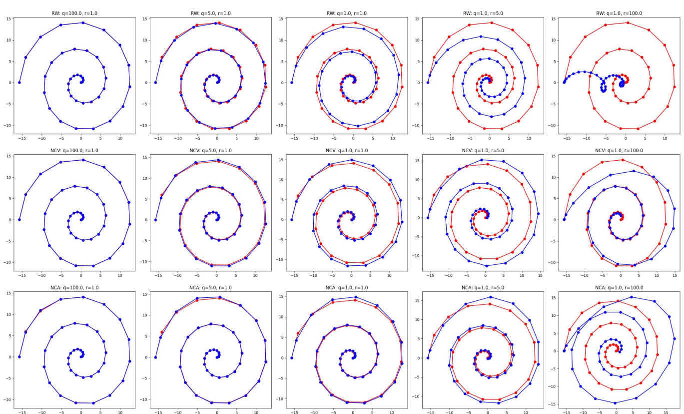

Figure 1: Kalman filter on spiral trajectory.

an example of the Kalman filter on a spiral trajectory, where the first row represents the RW motion model, the second row NCV motion model and the third row NCA motion model over different q and r parameters. The parameter q represents the noise in the motion model or uncertainty, and the parameter r represents the uncertainty of the observation model. We can see two trajectories, the red one represents the actual trajectory, and the blue one represents the prediction. We see that the best prediction is when q = 100 and r = 1, meaning that the best results are obtained when the observation model is certain and we have random noise in the motion model. When decreasing q and increasing r, we can see a slight and gradual decline in the prediction. To further check if the observations are accurate, the models are tested on tow additional trajectories. In figure 2, we see the observation models tested on a flower-like shaped trajectory, and in figure 3, we see a sun-line shaped trajectory. Both examples confirm the observations, and an additional observation can be made that depending on the situation, a specific motion model will outperform the other ones. If we

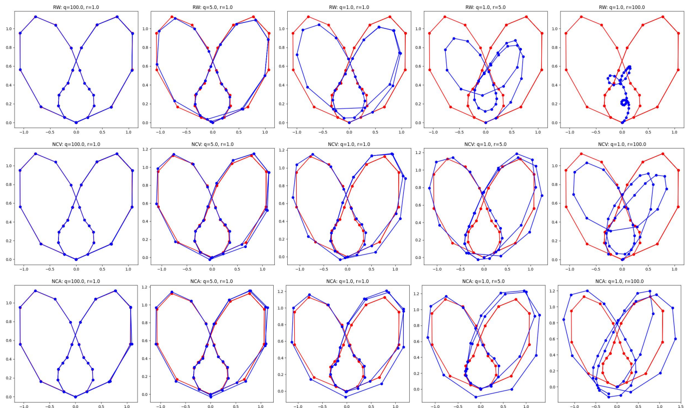

Figure 2: Kalman filter on flower-like trajectory.

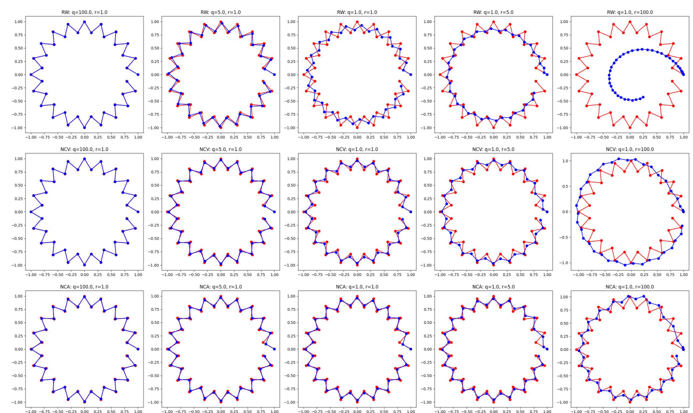

Figure 3: Kalman filter on sun-like trajectory.

have a continuous similar trajectory, NCV preforms really well. If we have sudden changes in the trajectory, the NCA model preforms well.

Particle filters

For this subtask, the particle tracker is implemented using the previously defined motion models. The tracker is tested on a variety of sequences, with different parameters, showing their performances on specific situations, and on the whole VOT14 dataset, showing the overall performance for each motion model with different parameters. The performance of the tracker is evaluated using: number of failures (robustness), average overlap (accuracy) and speed (measured in FPS). We can see an example of the particle tracker using NCV on the whole VOT14 dataset in figure I.

N	σ	d_σ	α	SF	q_{rate}	**Fails**	**AO**	**FPS**
100	0.5	0.15	0.05	0.7	0.1	32	0.45	61.69
100	0.5	0.15	0.05	1.0	0.1	42	0.46	54.28

Table I: Particle tracker using NCV on the whole VOT14 dataset.

We can see that the scale factor plays an important role in this implementation of the tracker. Because we are using color histograms, by decreasing the scale factor, we exclude the majority of the background and focus mostly on the center of the object. We further inspect the performance of the tracker on specific sequences, for different environments and trajectory types.

Sequence	Parameters					
	$\sigma = 0.5$, $\alpha = 0.05$			$\sigma = 1.0$, $\alpha = 0.1$		
	Fails	AO	FPS	Fails	AO	FPS
basketball	1	0.61	34.06	2	0.62	34.67
bicycle	0	0.48	41.82	0	0.44	43.92
torus	0	0.52	44.05	2	0.54	47.80
woman	1	0.48	40.75	5	0.42	38.99
hand2	4	0.42	47.26	7	0.43	47.67

Table II: Example of using NCV motion model of 5 different sequences with 2 sets of parameters.

The parameters are mostly set constant except for α, to see how the learning rate affects the tracker in different situations. N set to 100, the number of bins is set to 16, the scale factor is set to 1.0, and the distance sigma which is used in the recalculation of the weights is set to 0.15. Finally, the q_{rate} which is used to dynamically estimate the value for q for the motion model based on the size of the patch is set to 0.1.

In figure II, we see examples of NCV motion model on 5 different diverse sequences. The sequences were chosen specifically for showing the advantages and disadvantages of using NCV motion model. First, we can see that by using a lower α value we get better performance. In woman we get 4 more fails then by using a lower α. It makes sense, as more then half of the time, the woman in the sequence is occluded by a car. Also, we can see that the performance of NCV motion model is not very good for the hand2 sequence. The poor performance in that sequence is expected as the motion in that sequence is not constant, it has a lot of sudden changes. For the rest of the sequences, we get very good predictions, as the sequences have mostly nearly constant trajectories, especially the bicycle sequence.

Sequence		q = 0.01 r = 1		q = 0.1 r = 1		q = 0.5 r = 1	
		Fails	AO	Fails	AO	Fails	AO
basketball	RW	9	0.37	6	0.41	2	0.42
	NCV	0	0.55	2	0.61	2	0.62
	NCA	1	0.49	4	0.56	5	0.58
bolt	RW	6	0.45	4	0.43	2	0.54
	NCV	3	0.48	3	0.51	3	0.53
	NCA	7	0.48	8	0.53	11	0.43
polarbear	RW	0	0.44	0	0.45	0	0.51
	NCV	0	0.49	0	0.49	0	0.62
	NCA	1	0.47	0	0.57	0	0.56
sunshade	RW	12	0.41	9	0.41	5	0.51
	NCV	7	0.42	0	0.49	0	0.55
	NCA	4	0.39	1	0.55	2	0.52
gymnastics	RW	0	0.51	0	0.54	1	0.49
	NCV	1	0.49	1	0.49	1	0.50
	NCA	1	0.48	3	0.51	2	0.46

Table III: Comparing all three motion models with different model parameters.

In figure III, we can see the performance of all three motion models using different model parameters (q), on 5 different sequences. The parameter q in this case means the q_{rate}, which dynamically calculates the parameter q for the motion model based on the patch size using the formula $q = q_{rate} * min(patch.shape[0], patch.shape[1])$. We can see that in most cases, using a higher q_{rate} gives better performance for both RW and NCV models, and using smaller q_{rate} gives better performance for NCA. However, that is dependent of the situation in which the model is tracking the object. We can see that in sequences like basketball with nearly constant trajectory, with a few acceleration frames, we get better results with a smaller q_{rate}. However, on sequences like sunshade where there are quite a few rapid movements, using a larger q_{rate}, gives better results. Another observation is that out of the three models, based on the shown 5 sequences, on average NCV performed best, closely followed by NCA, and RW gave worse performance out of the three models.

Finally, we compare optimal parameters for NCV and see how the tracker preforms under different number of particles and different color spaces. The same parameters are used as the first example in figure I, under different N and color spaces (CS).

N	CS	Fails	AO	FPS
10	RGB	80	0.42	410.48
10	HSV	83	0.40	473.06
10	Lab	79	0.42	388.79
50	RGB	34	0.44	104.94
50	HSV	35	0.45	129.69
50	Lab	37	0.46	113.30
150	RGB	34	0.46	45.64
150	HSV	35	0.46	44.41
150	Lab	33	0.46	42.57

Table IV: Comparing particle tracker with NCV for different N and CS.

We can see from figure IV that increasing the number of particles helps improve the performance of the tracker, however it comes with the cost of speed. Furthermore, we can see that using different color spaces in this situation with the set parameters does sometimes improve performance, but not drastically. However, when tested on the sequence bolt, Lab gave 0 fails, outperforming RGB. Thus, on sequences that we have multiple similar objects near each other, using a different color space helps keep the tracker on the right object.

III. Conclusion

In this assignment, three motion models were implemented along with the Kalman filter. Also, the particle tracker was implemented and tested on a variety of situationsS with a variety of parameters using different motion models. The particle tracker outperforms the previously implemented trackers in this course by a fair margin. It is a strong tracker, that if used under the right circumstances and implemented the right way, will give great results.

Appendix

A. Random Walk (RW)

$$X_{state} = \begin{bmatrix} x \\ y \end{bmatrix} F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} L = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

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

B. Nearly Constant Velocity (NCV)

$$X_{state} = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} F = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \phi = \begin{bmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} Q = \begin{bmatrix} \Delta Tq & 0 & \Delta Tq & 0 \\ 0 & \Delta Tq & 0 & \Delta Tq \\ \Delta Tq & 0 & \Delta Tq & 0 \\ 0 & \Delta Tq & 0 & \Delta Tq \end{bmatrix}$$

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

C. Nearly Constant Acceleration (NCA)

$$X_{state} = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ \ddot{x} \\ \ddot{y} \end{bmatrix} H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} L = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

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

$$Q = \begin{bmatrix} \frac{\Delta Tq}{4} & 0 & \frac{\Delta Tq}{2} & 0 & \frac{\Delta Tq}{2} & 0 \\ 0 & \frac{\Delta Tq}{4} & 0 & \frac{\Delta Tq}{2} & 0 & \frac{\Delta Tq}{2} \\ \frac{\Delta Tq}{2} & 0 & \Delta Tq & 0 & \Delta Tq & 0 \\ 0 & \frac{\Delta Tq}{2} & 0 & \Delta Tq & 0 & \Delta Tq \\ \frac{\Delta Tq}{2} & 0 & \Delta Tq & 0 & \Delta Tq & 0 \\ 0 & \frac{\Delta Tq}{2} & 0 & \Delta Tq & 0 & \Delta Tq \end{bmatrix}$$