Human-Oriented Robotics | Winter term 2013/2014
Prof. Dr. Kai Arras, Social Robotics Lab | University of Freiburg
Lab instructors: Timm Linder, Luigi Palmieri, Billy Okal | Department of Computer Science

Exercise 9: Kalman Filter

Submission: Send your solution to `palmieri@informatik.uni-freiburg.de` until January 27, 2014 with subject line "[`exercises`] `Sheet 9`". All files (Matlab scripts, exported figures, handwritten notes in pdf/jpg format) should be put into a single zip file named `lastname_sheet9.zip`.

For this exercise, you will need to download the Matlab frames `updatestate.m`, `predictstate.m`, `PersonTrackingFrame.m` and dataset `datatracks.mat` from the course website.

Exercise 9.1: Getting Started, Read From Data Set

a) Start by loading the file `datatracks.mat` and check the variables that you have read in. The file contains one $2 \times n$-matrix `Z` with a sequence of (x, y)-observations of the position of a moving person observed by a simulated sensor and one Boolean $1 \times n$-vector `zvalid` whose entries indicate if the corresponding measurement is valid. We assume the target has been observed at a constant frequency f and let $\Delta t = 1/f$. What is the meaning of matrix `R` ?

b) Plot the observation sequence into a new figure. Use the `zvalid` variable to only plot the valid observations. Note that observation sequence may have gaps and outliers. Explain possible causes of such events.

We will now implement a Kalman filter the person and start with the observation sequence `Z`.

Exercise 9.2: State Representation and Initialization. We define the state representation of the filter to be $\mathbf{x} = (x, y, \dot{x}, \dot{y})^T$. It contains the estimated position of the person and its velocities in x and y directions. The associated covariance is thus a 4×4-matrix whose entries describe the uncertainties of the state components and covariances between them.

a) Initialize the state at time 0 from the first measurement. As our sensor can only observe the position of the target and not its velocities, we set the velocity component to zero and define a large initial covariance P_0 that reflects our lack of knowledge with suitably large number on its diagonal (`P0` is defined in the frame).

Exercise 9.3: Motion Model. The role of the motion model is to project the state into the future within Δt. Here we use the transition matrix F from the ball tracking example discussed in class. This model actually implements a constant velocity motion model that assumes constant velocities perturbed by a normally distributed zero-mean acceleration.

a) Complete the m-file for the function `predictstate` to implement the motion model. The function takes Δt, Q, the last posterior state estimates $\mathbf{x}(k|k), P(k|k)$ as input and returns the predicted state and predicted state covariance $\mathbf{x}(k+1|k), P(k+1|k)$. Q is given in the frame.

Exercise 9.4: Measurement Model.

a) In general, observations of the state cannot be done directly but remotely over a (linear or non-linear) function of the state. This relationship is the measurement or observation model, $\hat{\mathbf{z}}(k+1) = H\,\mathbf{x}(k+1|k)$. The model serves to predict the next measurement based on the predicted state. The term $\hat{\mathbf{z}}(k+1)$ is the position where we expect the next measurement to occur. Since in our case we can directly observe the two state components x and y, H acts as a row selection matrix. Determine H and assign the predicted measurement to a variable, e.g. `zp`.

b) To extend the tracker with the capability to deal with outliers we use a statistical compatibility test, also known as *gating*. Gaiting is a test if $\mathbf{z}(k+1)$ is statistically compatible with the prediction $\hat{\mathbf{z}}(k+1)$. Implement the gating test

$$d^2 = \nu(k+1)^T\, S(k+1)^{-1}\, \nu(k+1)$$

using the innovation $\nu(k+1)$ and innovation matrix $S(k+1)$:

$$\begin{aligned} \nu(k+1) &= \mathbf{z}(k+1) - \hat{\mathbf{z}}(k+1) \\ S(k+1) &= H\ P(k+1|k)\ H^T + R \end{aligned}$$

The distance d^2 is the squared Mahalanobis distance, a generalized form of the Euclidian distance. An observation is an outlier if the condition $d^2 < \theta$ is not satisfied with θ being a threshold drawn from a cumulative χ^2 (chi-square) distribution. To obtain θ, use `chi2invtable(0.99,2)` from the `librobotics` library. In case the test is not satisfied, the state predictions are the best available estimates for the cycle's posteriors. Thus, close the loop by copying the state predictions into the posteriors $\mathbf{x}(k+1|k+1), P(k+1|k+1)$.

Exercise 9.5: Kalman Filter. Given the predicted state and the observation that successfully passed the gating test, we are now able to close the loop and compute the posterior state and state covariance estimates $\mathbf{x}(k+1|k+1), P(k+1|k+1)$. This is done by the Kalman filter, the optimal minimum mean-square error estimator under linear Gaussian conditions.

a) The Kalman update expressions for the mean $\mathbf{x}$ and the covariance P are

$$\begin{aligned} K(k+1) &= P(k+1|k)\ H^T\ S(k+1)^{-1} \\ \mathbf{x}(k+1|k+1) &= \mathbf{x}(k+1|k) + K(k+1)\ \nu(k+1) \\ P(k+1|k+1) &= P(k+1|k) - K(k+1)\ H\ P(k+1|k) \end{aligned}$$

Implement the update equations in the m-file for `[x P] = updatestate(xp,Pp,v,S,H)` where `xp,Pp` denote the predicted state estimates $\mathbf{x}(k+1|k), P(k+1|k)$.

b) We want to store the histories of variables `x,P,xp,Pp` over the tracking sequence. This is done by copying the estimate for each k into variables called, for instance `xhist`, `Phist` etc. Variable `xhist` has dimension $4\times n$, `Phist` has dimension $4\times 4\times n$ and is a multi-dimensional array. Concatenate the two matrices along the third dimension.

c) Track the person by running the filter over the observation sequence. Plot the estimation histories of $\mathbf{x}(k+1|k+1), P(k+1|k+1)$ and $\mathbf{x}(k+1|k), P(k+1|k)$. For the covariance matrices, use the `librobotics`-command `drawprobellipse` (use 0.95 for α). Explain the tracking behavior of the filter in particular during maneuvers.

d) Plot the state component x, y, $\dot{x}, \dot{y}$, and the four diagonal elements of the $P(k+1|k+1)$ in different subfigures (e.g. using `subplot`). **Hint:** to obtain a vector of values across a multi-dimensional array, use the `squeeze` command.