Kalman Filter

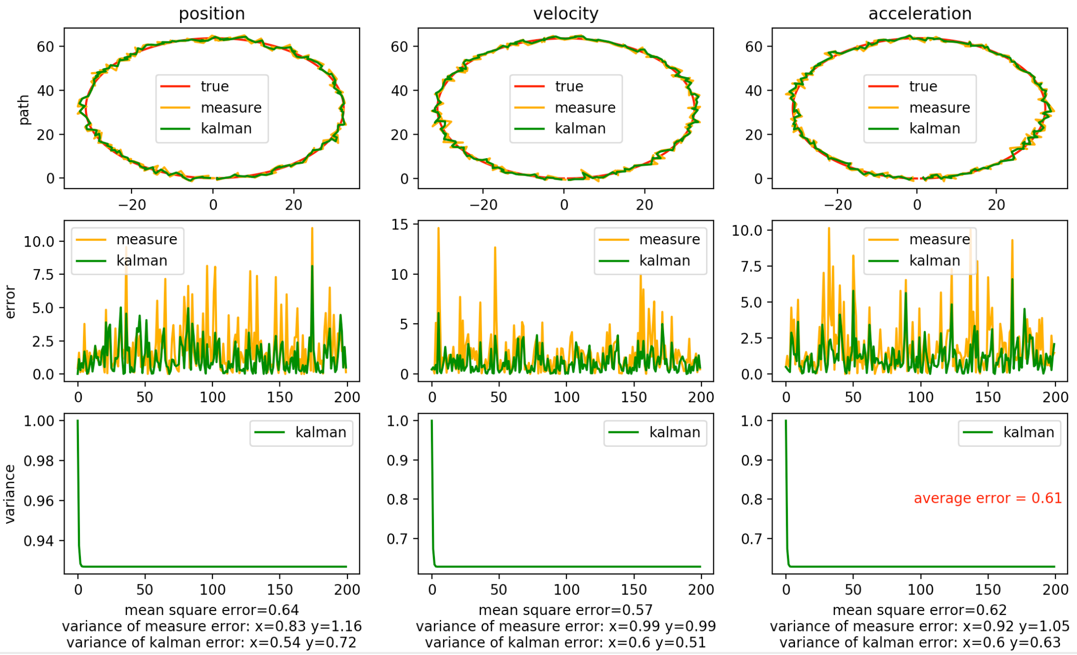
Jacky Wang 40575042H

- How do I implement Kalman filter

In order to implement three versions of the Kalman Filter, I implement each sample file which includes true path and three track data. The position in x and y is based on track1, the velocity is based on track2, while the acceleration is based on track3.

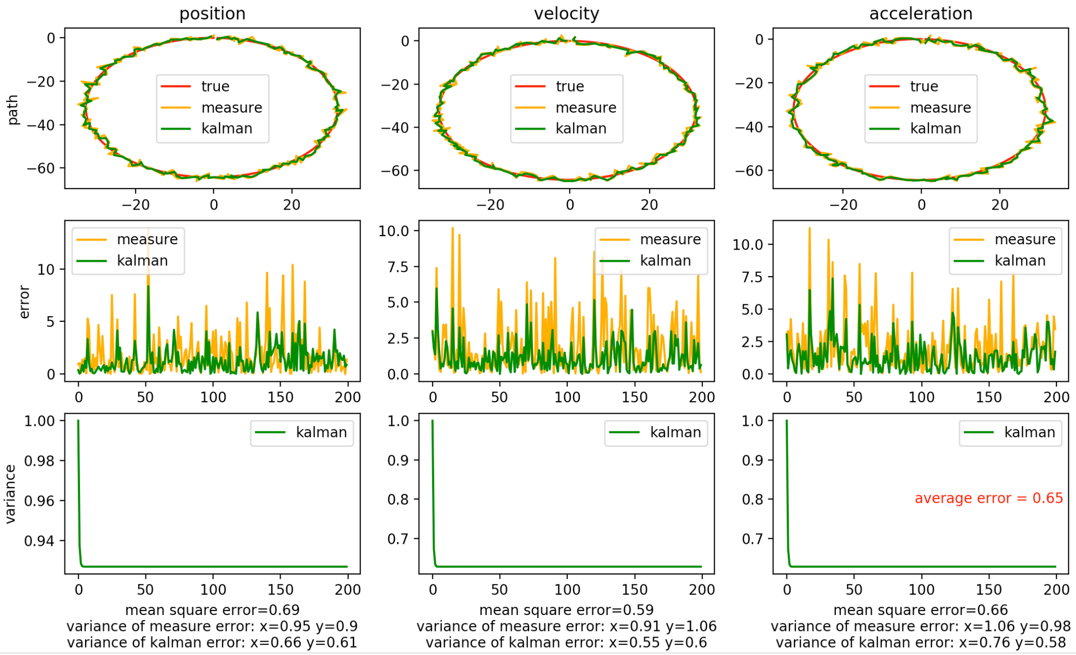
- Result (All of these information below are putted into one image)
  - The true path of the robot
  - The output of the Kalman Filter for the path of the robot
  - The variance of the Kalman Filter against time
  - The error of the Kalman Filter path against time
  - average error = (mse_position + mse_velocity + mse_acceleration) / 3

1. Data1.dat



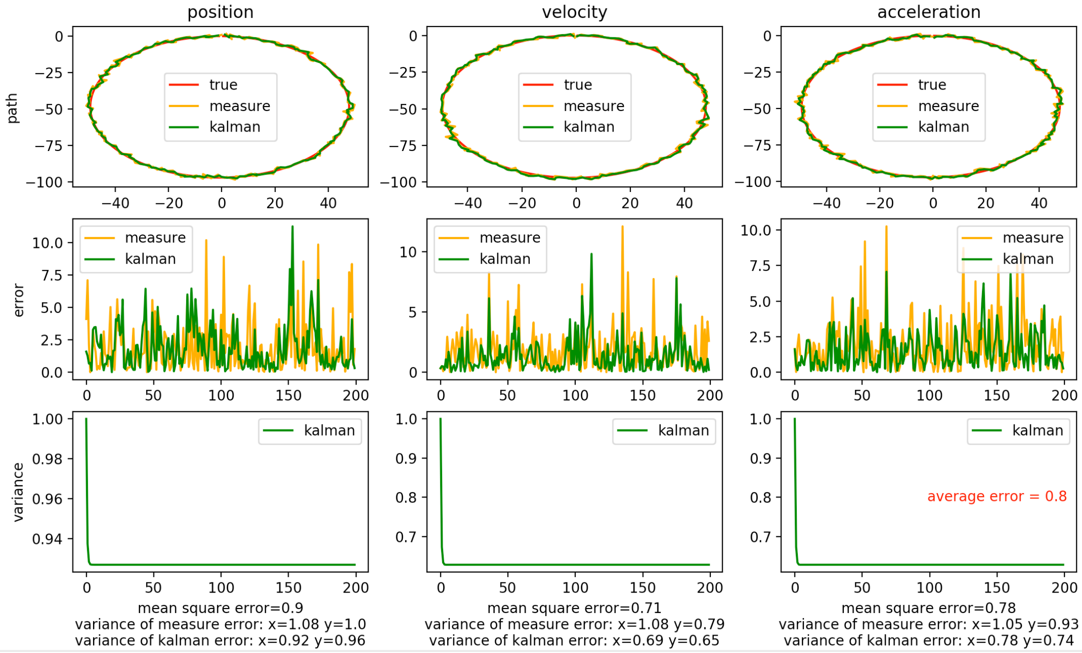
	position	velocity	acceleration
mean square error	0.64	0.57	0.62
variance of measure error	x: 0.83 y: 1.16	x: 0.99 y: 0.99	x: 0.92 y: 1.05
variance of Kalman error	x: 0.54 y: 0.72	x: 0.6 y: 0.51	x: 0.6 y: 0.63
average error	0.61		

2. Data2.dat



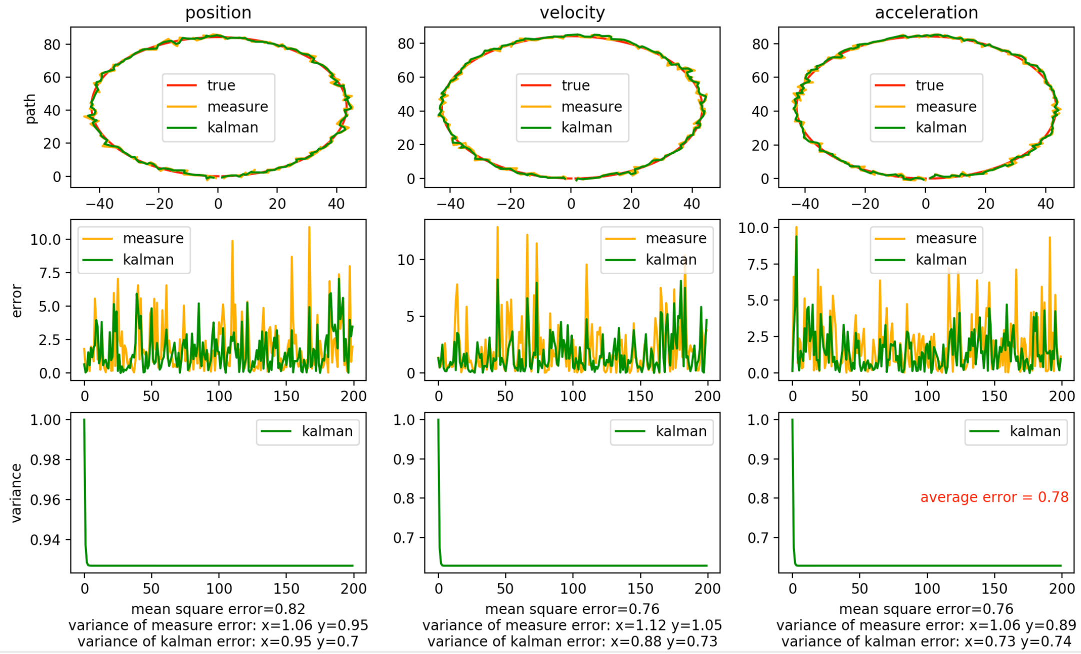
	position	velocity	acceleration
mean square error	0.69	0.59	0.66
variance of measure error	x: 0.95 y: 0.9	x: 0.91 y: 1.06	x: 1.06 y: 0.98
variance of Kalman error	x: 0.66 y: 0.61	x: 0.55 y: 0.6	x: 0.76 y: 0.58
average error	0.65		

3. Data3.dat



	position	velocity	acceleration
mean square error	0.9	0.71	0.78
variance of measure error	x: 1.08 y: 1.0	x: 1.08 y: 0.79	x: 1.05 y: 0.93
variance of Kalman error	x: 0.92 y: 0.96	x: 0.69 y: 0.65	x: 0.78 y: 0.74
average error	0.8		

4. Data4.dat



	position	velocity	acceleration
mean square error	0.82	0.76	0.76
variance of measure error	x: 1.06 y: 0.95	x: 1.12 y: 1.05	x: 1.06 y: 0.89
variance of Kalman error	x: 0.95 y: 0.7	x: 0.88 y: 0.73	x: 0.73 y: 0.74
average error	0.78		

- Evaluation

  - How do I evaluate performance of Kalman filter

In order to evaluate performance of Kalman filter, I use mean square error(mse) function to compare with each Kalman filter, because mse function can tell us the distance between true value and Kalman value, as well as distance between true value and measurement.

  - Why I use mean square error function

Furthermore, I decided to use mean square error function to evaluate because I found something interesting. When I tuned the parameter of noise matrix(R and Q), two phenomena appeared. One is Kalman filter (high R and Q) path will be very accurate but consists of gaps with no data. At first, I thought this performance of Kalman filter is very accurate, but due to the lagged gaps, Kalman filter path is actually not quite precise in the present time. In contrast, the other Kalman filter path have relatively bigger error but it is very close to the true value, and that means the distance between true value and Kalman value is small. Therefore, mean square error function not only can tell a distance but also give an example of not always trusting one’s eyes.

  - Results and Discussion section

The results show that data1 file performs the best, and velocity in the data1 file performs best of all, because it has the minimum average error and minimum mean square error.

Besides, the results indicate that all of the variances of the Kalman error are lower than variance of measurement error, that means the Kalman values are more stable than measurement, also the values are more smooth than measurement.

- Drawbacks of the Kalman filter

Since predicted values of Kalman filter will be affected by process noise covariance matrix and measurement covariance matrix (R and Q), it is very important to tune parameter of R and Q. According to this, the disadvantage of the Kalman filter is that it requires a lot of past data to determine parameter of the R and Q in order to predict the true values accurately.

- Solve the drawbacks of the Kalman filter

Since this is the initialization problem, I tend to tune parameter equally. Even though the mean square error are not quite small, it is acceptable.