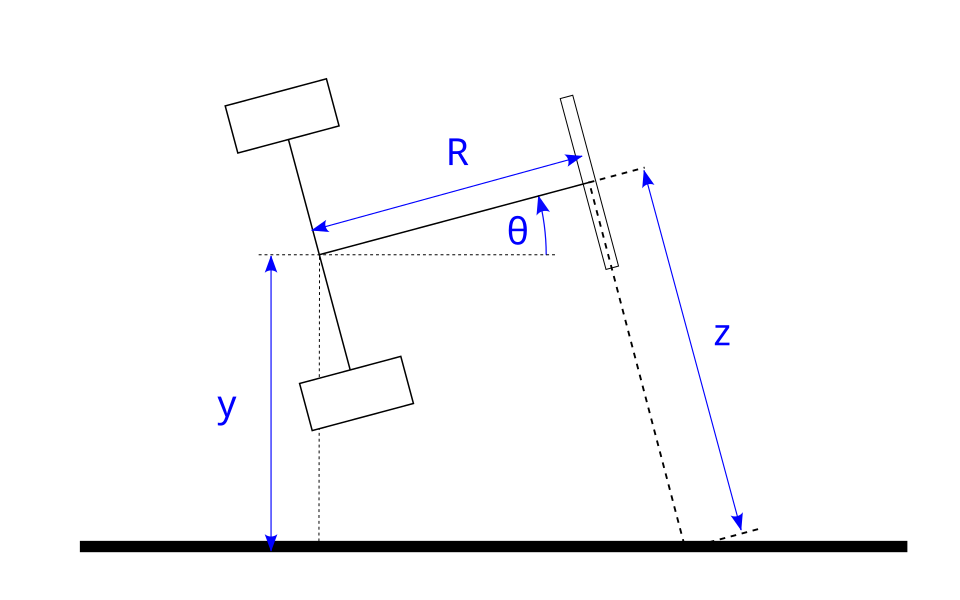
I implemented LQR method and Kalman filter on a line follower.

The robot is composed of two wheels and a row of sensors at the front. The distance between sensors and the center of two wheels is . The robot is moving forward at a constant speed . The angular velocity is the input of the system to make sure that the robot follows the line. The sensor can measure the distance away from the line, and the measured value is the output of the system. is the distance between the line and the center of two wheels. is the angle between the moving directions and the line.



It can be seen that and

I design the state to be , the state space model can be linearized at

. Therefore,

This system is controllable and observable.

Because the sensor value is discrete, the variance of output is

Based on simulation, this method works great on following a straight line.

Interestingly, it works on following a curve under certain conditions.

When calculating the LQR gain, the weighted matrix for the state which is a 2\*2 matrix. The weight matrix for input is float value. If the weight for is smaller or is smaller, the robot will rotate more aggressively to follow the line, and therefore it can follow a sharper curve. On the other hand. The optimal weight for in depends on . If decrease, the weight for needs to be increased. Also, the upper bound for is determined by the width of sensors. When increases, it required wider range of distribution of sensors so that robot can correct the direction in time. The feedback frequency could be as low as 100Hz under some conditions.