Ali Aydın YAMANDAĞ
Tel: +90 506 668 08 50 mail: aliaydinyamandageee@gmail.com
Emre YILMAZ
Tel: +90 506 888 77 42 mail: emreyilmazeee@gmail.com
Gürkan Durmuş YILMAZ
Tel: +90 554 612 27 66 mail: gurkandy@gmail.com
Hakkı GÜLCÜ
Tel: +90 534 397 29 66 mail: hakkigulcu35@gmail.com
Sonay ULUKAYA
Tel: +90 506 138 94 38 mail: sonayulukaya@gmail.com

WEEKLY REPORT VIII

This week we have done research on the types of SLAM algorithms. For our project we have chosen Kalman filter based SLAM.

Kalman filter is a variance of Bayes filter for the case of Gaussian distributions with linear functions. Because given this assumption everything stays Gaussian during the recursive steps of the filter. However, for SLAM we have nonlinear functions due to angles being involved. Therefore, the extended Kalman filter is being used which is just a generalization of the Kalman filter to nonlinear functions.

Kalman filter is a recursive filter. There are two steps of it, namely the prediction step and the correction step. The prediction step estimates how the states have changed after applying the motion command. The correction step relates the prediction with the observation and a correction is made depending on the differences between them.

The state space of the operation is given in equation 1. x_R represents the position of the robot whereas m_i terms represent the landmark positions. The algorithm can be summarized as the following part in 5 steps.

$$\mu = \begin{pmatrix} x_R \\ m_1 \\ m_2 \\ \vdots \\ m_n \end{pmatrix} \quad (1)$$

1. State Prediction

When a motion command is applied, only the position of the robot changes meaning that it does not change the positions of the landmarks given the robot does not modify the environment. Therefore, only the robots position needs to be predicted in the state space. After this operation, the uncertainty in robots position as well as the uncertainty of the landmark with respect to robot need to be updated. However, the uncertainty in the positions of the landmarks with respect to other landmarks stay the same.

2. Observation Prediction

When a motion command is applied, observation is predicted using the covariance matrix.

3. Observation

After the motion command robot will observe the environment.

4. Data association

Observation data needs to be associated with the landmarks. After that the difference between the observation data and the predicted observation data needs to be calculated.

5. Update

After completing the steps before, the states as well as the covariance matrix can be updated.

During this week, we will implement this algorithm and try to obtain a very basic map for the module demo.