Project 2: Extend Kalman Filter

CPE 416: Autonomous Mobile Robotics – Fall 2025
Team Members: Johnny Rourke, Samuel
Date: November 21, 2025

The covariance matrix produced by the Extended Kalman Filter provides a numerical measure of the filter's confidence in its current state estimate. Our state vector consisted of three state variables: $x=[x,\ y,\ \theta]$. Therefore the covariance for our Extended Kalman Filter is then a 3×3 matrix whose diagonal entries represent the uncertainty in each state variable, and the off-diagonal terms represent the correlations between the errors in each of the state variables. Throughout the experiment, the behavior of P reflects the accuracy of each state variable. We implement the extended Kalman filter in two main steps: predict and update.

For the prediction step we use the IMU (Inertial Measurement Unit) inside the OAK-D camera on the robot to measure the yaw rate for the robot directly. During the prediction step the filter integrates IMU yaw rate to estimate the heading of the robot but this model cannot predict the entire next state of the robot because we do not predict the x and y.

Next in the update state we receive a LIDAR odometry message, the update step dramatically the update step has a full estimation of the new state of the robot. Using the lidar scan and the velocities sent to the robot the odom messages predicts a new x, y, and θ for the robot. This provides a relatively accurate evaluation of the state. However this model gives a pretty inaccurate reading for the heading. In order to improve accuracy of the state estimation we then fuse these two models using some linear algebra. The linear algebra fuses both the state estimation and the covariance for the model. Using this EKF we tried to plot a teleoperated path we took using the covariance values from both the lidar and imu sensors as shown in Figure 1.

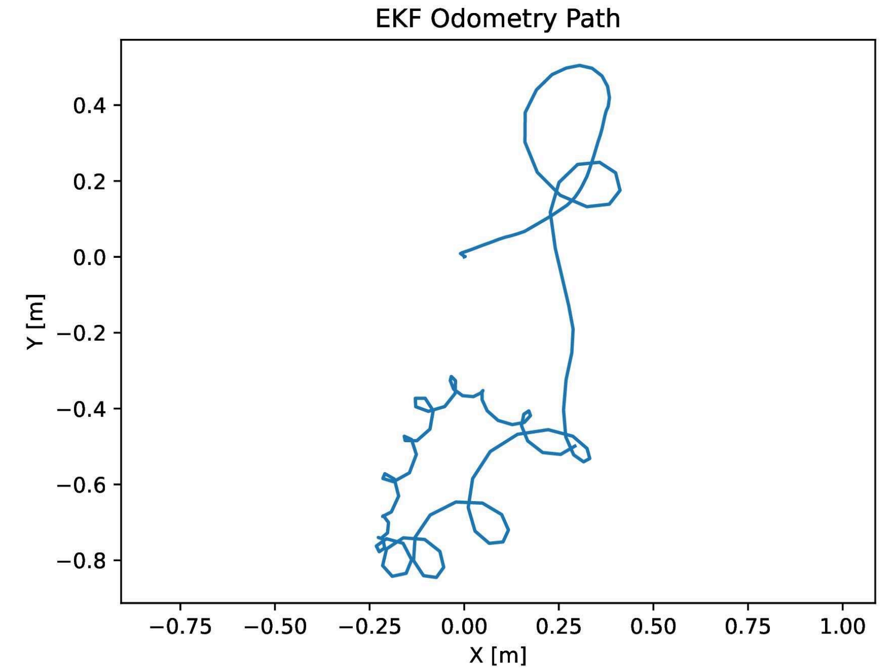

Figure 1. *The EKF Odometry Path is made by using the variable covariance from the sensors.*

This was obviously not the path we took, to try to improve this model we decided to hardcode the covariance from the sensors, we learned the LIDAR yaw input was least reliable for this reason we made the covariance from the yaw the highest so that we rely more on the IMU as seen in Figure 2.

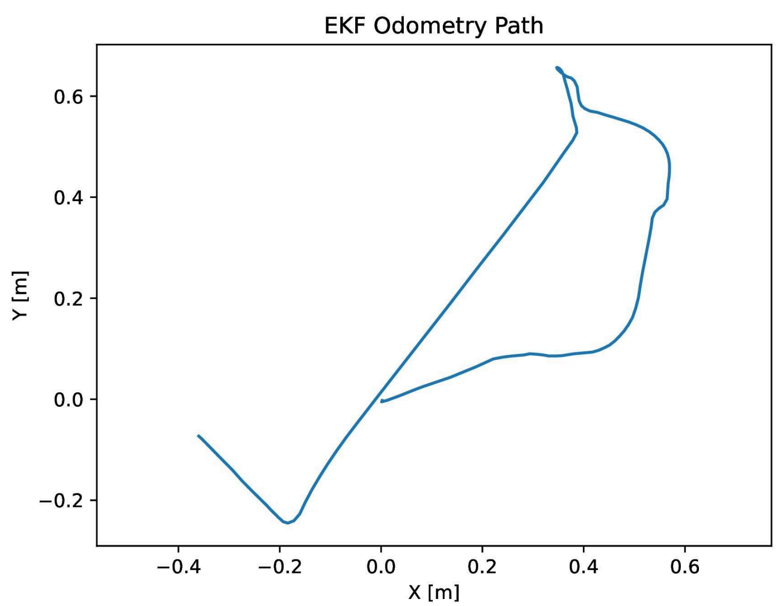

Figure 2. *The EKF Odometry Path is made by using the fixed covariance.*

After increasing the LiDAR yaw covariance, the filter relied more on the IMU's yaw rate, which noticeably stabilized the heading estimate. Reducing trust in the LiDAR's noisy orientation measurements prevented the large jumps and oscillations the EKF previously inherited from the odometry, and the yaw trajectory became smoother and more consistent. However, the improvement is still limited. The IMU-based integration continues to introduce gradual drift during longer runs, and additional noise is cause by the steady state error in the motors leading to the noise from the lidar updates of the x and y values. The adjustment helped, but the yaw estimate remains imperfect and would require further tuning to be more reliable and repeatable. In Figure 3. We show how the covariance evolves over the course of the robots movement.

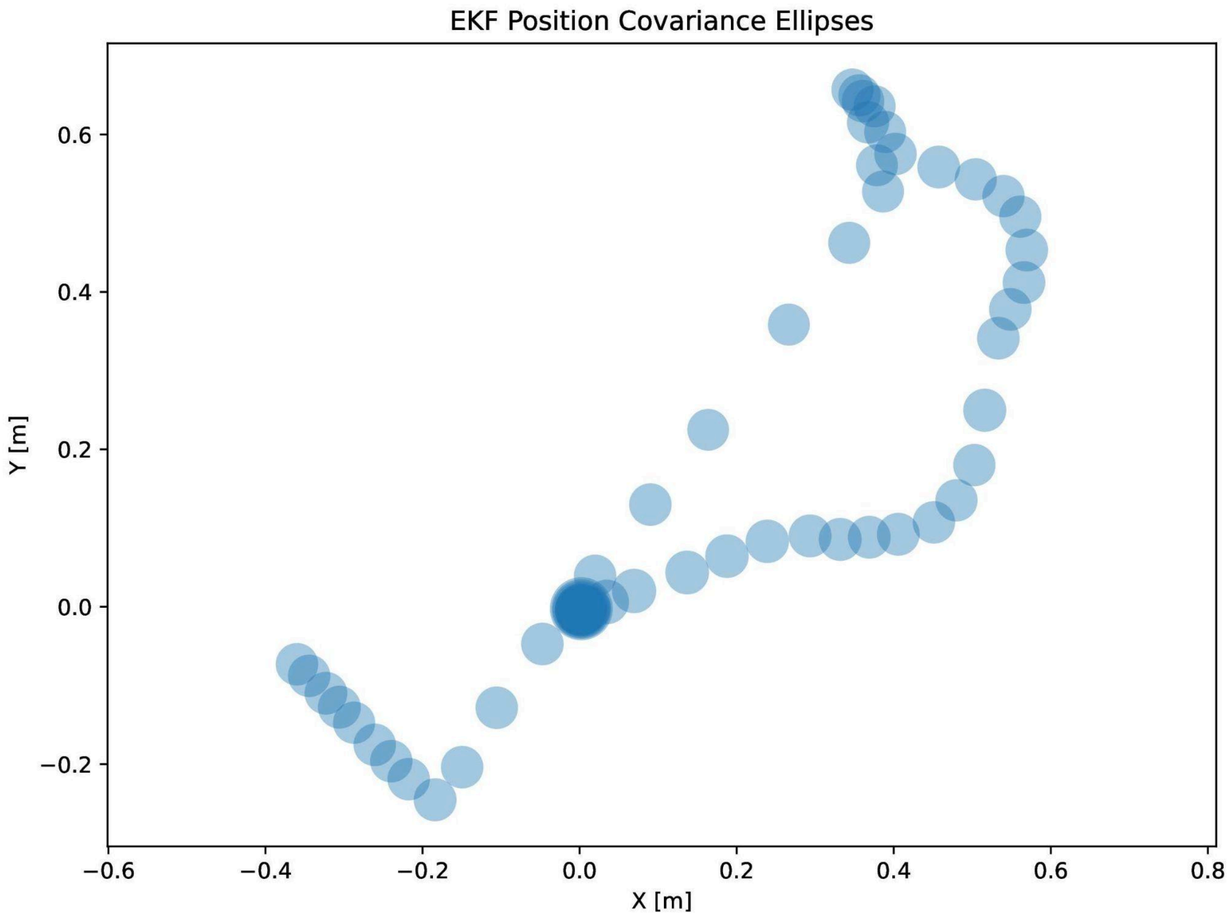

Figure 2. *The EKF Odometry Path is made by using the fixed covariance with the size of the marker being proportional to the* $\theta\theta$ covariance..