Midterm Competition Progress Report

By: John Abanes

April 1, 2024

Brief Description

The project solution consists of two primary approaches: optimization and visual odometry. For the optimization, I analyzed which values for the codebook and BallTree resulted in the fastest convergence. I sampled multiple values, using the iteration count and a stopwatch to measure performance. I then combined the optimal values across the cluster count, n_init, and leaf size. For the display_next_best_view function, I simply changed it to *index+8* for now since this was the furthest I could do while not being disoriented.

I began working on visual odometry (VO) as a way to map the maze. This is because I wanted to explore the maze algorithmically, such as through depth-first-search (DFS). Ideally, I could just manually control the camera to stick to the right/left side while using VO as a guide. However, I didn't have enough time to fully implement VO. The performance is poor and the odometry itself is still questionable. So I left it out for my current implementation, but the files for it are still inside the codebase. Below is an image of it in action.