Modeling Stairs using Stereo Vision

Yuncong Chen
Advisor: Thomas Bewley, Yoav Freund

Introduction

Modeling stairs is an important problem for autonomous robot navigation in a multilevel building. A stairs modeling algorithm provides mobile robots with information about the stairs like the width and slope so that robots can choose the optimal paths and maneuvers to climb the stairs. This project develops such an algorithm for robots with stereo cameras. Given stereo images of a staircase, step edges are first detected from RGB images and projected in 3D space. The best model is then found using a sample consensus approach.

Algorithm

0. Stereo Setup

The customized stereo rig consists of two Logitech C310 webcams. We perform standard stereo calibration and produce vertically rectified image pairs. A stereo matching algorithm estimates the disparity of each pixel, which in turn gives the depth of that pixel in the 3D space.

Since stairs steps are always horizontal, placing cameras at the same level does not provide enough disambiguation along the scanlines. A vertical configuration directly improves the result of stereo matching.

1. Computing the Disparity Map

The lack of textures in a stairs scene makes stereo matching difficult. We have compared a number of algorithms including MRF-stereo, ELAS and SGBM (Semi-global block matching). SGBM has the best result, albeit still fails to compute the disparity values for a good proportion of pixels. An ideal disparity map requires heavy tuning of the parameters, which is infeasible for a real-time robotic application. This motivates our sample consensus method that is robust against noisy disparity estimates.

2. Generate Point Cloud

3D point cloud is generated based on the disparity map. The cloud is first down-sampled into voxel grids, so that points in each 3D grid reduce to its centroid. The surface normal at each point is estimated by fitting a planar patch to all points in a small neighborhood using PCA. If the average point-to-patch distance is large, RANSAC is applied to prevent outliers from biasing the estimate. Isolated points are detected as those with a large average neighbor distance and are removed. Nearest-neighbor search is made efficient using a KD-tree.

3. Extract Step Edges

For a regular staircase, the step edges largely determines its location. It is much easier to find these edges from the 2D RGB image than from the 3D point cloud. We use Canny line detector to produce binary image with the edges, then apply weighted Hough transform to obtain parameters of the line segments. Edge points vote higher for lines that in accordance with their gradient direction. As the complexity of Hough transform is determined by the number of voters, we employ Progressive Probabilistic Hough Transform to reduce the number of points that actually vote. Finally, non-horizontal lines are discarded and nearby lines are merged. Note that our algorithm does not require all edges to be detected, and the edges do not have to cover the entire step width.

4. Project Step Edges to 3D Space

Pixels that are on the detected step edges and have valid disparity values are projected to the 3D space. PCA is then used to fit 3D lines to these projected edge points. The center points of these edge lines and the average line direction are used to produce a set of parallel lines that act as constraints for the model. Edges that were not detected in the previous step can be located and inserted by identifying unusually large distances between existing edges.

5. Sample Consensus Plane Estimation

Sample consensus method generates random models that are subject to certain constraints, and finds the one with the highest inlier support from the point cloud. In case of stairs, the model contains two sets of equally spaced planes that are orthogonal and coincide roughly on the projected edge lines. Specifically, we estimate the rise height, tread depth and step width.

The on-robot gyroscope senses the pose of the camera and therefore gives reliable initial estimate for the orientation of rise planes in the 3D point cloud. Based on this and locations of the edge lines we can estimate the height and depth of each step. A number of candidate models are sampled by allowing random parameter deviations around these estimates.

To compute the inlier support of a candidate model, we sample test points from the point cloud. Valid test point must be in the regions right next to a plane. Sampling and region constraint limit the number of test points and make this method available for real-time use. A test point is regarded an inlier if it is within a distance threshold from its closest plane, or the surface normal (estimated in step 2) is sufficiently aligned with the normal of the associated plane. The degree of alignment determines the support weight of this inlier.

6. Estimate Step Width

The dimensions of steps are estimated by projecting the inliers to their corresponding planes and finding the bounding boxes of the projections.

Future Work

- Robustness of this algorithm will be evaluated under various lighting conditions, stairs textures, and different viewpoints.
- The presented algorithm uses only one stereo frame. An extension is to use a sequence of consecutive frames. The parameters inferred from each frame can be fused (i.e. with a Kalman filter) and the model can be constantly corrected and extended by new observations. Locations of step edges and feasible plane normals can be tracked continuously using motion information that is estimated with on-board sensors or visual odometry. These give strong priors that help reduce the model search space.
- Design path planning algorithm that finds the best path and maneuvers for a robot to climb the stairs given the model. This largely depends on the dynamic and kinematic properties of the robot.