Cartographer Documentation

Release 1.0.0

The Cartographer Authors

Mar 31, 2017

Contents

1 Configuration **1**
- 1.1 cartographer.common.proto.CeresSolverOptions . . . 1
- 1.2 cartographer.kalman_filter.proto.PoseTrackerOptions . . . 1
- 1.3 cartographer.mapping.proto.MapBuilderOptions . . . 1
- 1.4 cartographer.mapping.proto.SparsePoseGraphOptions . . . 2
- 1.5 cartographer.mapping.sparse_pose_graph.proto.ConstraintBuilderOptions . . . 2
- 1.6 cartographer.mapping.sparse_pose_graph.proto.OptimizationProblemOptions . . . 2
- 1.7 cartographer.mapping_2d.proto.LocalTrajectoryBuilderOptions . . . 3
- 1.8 cartographer.mapping_2d.proto.RangeDataInserterOptions . . . 3
- 1.9 cartographer.mapping_2d.proto.SubmapsOptions . . . 4
- 1.10 cartographer.mapping_2d.scan_matching.proto.CeresScanMatcherOptions . . . 4
- 1.11 cartographer.mapping_2d.scan_matching.proto.FastCorrelativeScanMatcherOptions . . . 4
- 1.12 cartographer.mapping_2d.scan_matching.proto.RealTimeCorrelativeScanMatcherOptions . . . 4
- 1.13 cartographer.mapping_3d.proto.KalmanLocalTrajectoryBuilderOptions . . . 5
- 1.14 cartographer.mapping_3d.proto.LocalTrajectoryBuilderOptions . . . 5
- 1.15 cartographer.mapping_3d.proto.MotionFilterOptions . . . 5
- 1.16 cartographer.mapping_3d.proto.OptimizingLocalTrajectoryBuilderOptions . . . 5
- 1.17 cartographer.mapping_3d.proto.RangeDataInserterOptions . . . 5
- 1.18 cartographer.mapping_3d.proto.SubmapsOptions . . . 6
- 1.19 cartographer.mapping_3d.scan_matching.proto.CeresScanMatcherOptions . . . 6
- 1.20 cartographer.mapping_3d.scan_matching.proto.FastCorrelativeScanMatcherOptions . . . 6
- 1.21 cartographer.sensor.proto.AdaptiveVoxelFilterOptions . . . 6

2 Getting started **7**
- 2.1 Getting started with ROS . . . 7
- 2.2 Getting started without ROS . . . 7

3 System Requirements **9**
- 3.1 Known Issues . . . 9

4 How to cite us **11**

CHAPTER 1

Configuration

cartographer.common.proto.CeresSolverOptions

bool use_nonmonotonic_steps Configure the Ceres solver. See the Ceres documentation for more information: https://code.google.com/p/ceres-solver/

int32 max_num_iterations Not yet documented.

int32 num_threads Not yet documented.

cartographer.kalman_filter.proto.PoseTrackerOptions

double position_model_variance Model variances depend linearly on time.

double orientation_model_variance Not yet documented.

double velocity_model_variance Not yet documented.

double imu_gravity_time_constant Time constant for the orientation moving average based on observed gravity via linear acceleration.

double imu_gravity_variance Not yet documented.

int32 num_odometry_states Maximum number of previous odometry states to keep.

cartographer.mapping.proto.MapBuilderOptions

bool use_trajectory_builder_2d Not yet documented.

cartographer.mapping_2d.proto.LocalTrajectoryBuilderOptions trajectory_builder_2d_options Not yet documented.

bool use_trajectory_builder_3d Not yet documented.

cartographer.mapping_3d.proto.LocalTrajectoryBuilderOptions trajectory_builder_3d_options Not yet documented.

int32 num_background_threads Number of threads to use for background computations.

cartographer.mapping.proto.SparsePoseGraphOptions sparse_pose_graph_options Not yet documented.

cartographer.mapping.proto.SparsePoseGraphOptions

int32 optimize_every_n_scans Online loop closure: If positive, will run the loop closure while the map is built.

cartographer.mapping.sparse_pose_graph.proto.ConstraintBuilderOptions constraint_builder_options Options for the constraint builder.

cartographer.mapping.sparse_pose_graph.proto.OptimizationProblemOptions optimization_problem_options Options for the optimization problem.

int32 max_num_final_iterations Number of iterations to use in 'optimization_problem_options' for the final optimization.

double global_sampling_ratio Rate at which we sample a single trajectory's scans for global localization.

cartographer.mapping.sparse_pose_graph.proto.ConstraintBuilderOptions

double sampling_ratio A constraint will be added if the proportion of added constraints to potential constraints drops below this number.

double max_constraint_distance Threshold for poses to be considered near a submap.

cartographer.sensor.proto.AdaptiveVoxelFilterOptions adaptive_voxel_filter_options Voxel filter used to compute a sparser point cloud for matching.

double min_score Threshold for the scan match score below which a match is not considered. Low scores indicate that the scan and map do not look similar.

double global_localization_min_score Threshold below which global localizations are not trusted.

double lower_covariance_eigenvalue_bound Lower bound for covariance eigenvalues to limit the weight of matches.

bool log_matches If enabled, logs information of loop-closing constraints for debugging.

cartographer.mapping_2d.scan_matching.proto.FastCorrelativeScanMatcherOptions fast_correlative_scan_matcher_options Options for the internally used scan matchers.

cartographer.mapping_2d.scan_matching.proto.CeresScanMatcherOptions ceres_scan_matcher_options Not yet documented.

cartographer.mapping_3d.scan_matching.proto.FastCorrelativeScanMatcherOptions fast_correlative_scan_matcher_options_3 Not yet documented.

cartographer.mapping_3d.scan_matching.proto.CeresScanMatcherOptions ceres_scan_matcher_options_3d Not yet documented.

cartographer.mapping.sparse_pose_graph.proto.OptimizationProblemOptions

double huber_scale Scaling parameter for Huber loss function.

double acceleration_weight Scaling parameter for the IMU acceleration term.

double rotation_weight Scaling parameter for the IMU rotation term.

double consecutive_scan_translation_penalty_factor Penalty factors for changes to the relative pose between consecutive scans.

double consecutive_scan_rotation_penalty_factor Not yet documented.

bool log_solver_summary If true, the Ceres solver summary will be logged for every optimization.

cartographer.common.proto.CeresSolverOptions ceres_solver_options Not yet documented.

cartographer.mapping_2d.proto.LocalTrajectoryBuilderOptions

float laser_min_range Laser returns outside these ranges will be dropped.

float laser_max_range Not yet documented.

float laser_min_z Not yet documented.

float laser_max_z Not yet documented.

float laser_missing_echo_ray_length Laser returns beyond 'laser_max_range' will be inserted with this length as empty space.

float laser_voxel_filter_size Voxel filter that gets applied to the range data immediately after cropping.

bool use_online_correlative_scan_matching Whether to solve the online scan matching first using the correlative scan matcher to generate a good starting point for Ceres.

cartographer.sensor.proto.AdaptiveVoxelFilterOptions adaptive_voxel_filter_options Voxel filter used to compute a sparser point cloud for matching.

cartographer.mapping_2d.scan_matching.proto.RealTimeCorrelativeScanMatcherOptions real_time_correlative_scan_matcher Not yet documented.

cartographer.mapping_2d.scan_matching.proto.CeresScanMatcherOptions ceres_scan_matcher_options Not yet documented.

cartographer.mapping_3d.proto.MotionFilterOptions motion_filter_options Not yet documented.

double imu_gravity_time_constant Time constant in seconds for the orientation moving average based on observed gravity via the IMU. It should be chosen so that the error 1. from acceleration measurements not due to gravity (which gets worse when the constant is reduced) and 2. from integration of angular velocities (which gets worse when the constant is increased) is balanced.

int32 num_odometry_states Maximum number of previous odometry states to keep.

cartographer.mapping_2d.proto.SubmapsOptions submaps_options Not yet documented.

bool use_imu_data True if IMU data should be expected and used.

cartographer.mapping_2d.proto.RangeDataInserterOptions

double hit_probability Probability change for a hit (this will be converted to odds and therefore must be greater than 0.5).

double miss_probability Probability change for a miss (this will be converted to odds and therefore must be less than 0.5).

bool insert_free_space If 'false', free space will not change the probabilities in the occupancy grid.

cartographer.mapping_2d.proto.SubmapsOptions

double resolution Resolution of the map in meters.

double half_length Half the width/height of each submap, its "radius".

int32 num_range_data Number of scans before adding a new submap. Each submap will get twice the number of scans inserted: First for initialization without being matched against, then while being matched.

bool output_debug_images If enabled, submap%d.png images are written for debugging.

cartographer.mapping_2d.proto.RangeDataInserterOptions range_data_inserter_options Not yet documented.

cartographer.mapping_2d.scan_matching.proto.CeresScanMatcherOptions

double occupied_space_weight Scaling parameters for each cost functor.

double translation_weight Not yet documented.

double rotation_weight Not yet documented.

double covariance_scale Scale applied to the covariance estimate from Ceres.

cartographer.common.proto.CeresSolverOptions ceres_solver_options Configure the Ceres solver. See the Ceres documentation for more information: https://code.google.com/p/ceres-solver/

cartographer.mapping_2d.scan_matching.proto.FastCorrelativeScanMatcherOp

double linear_search_window Minimum linear search window in which the best possible scan alignment will be found.

double angular_search_window Minimum angular search window in which the best possible scan alignment will be found.

int32 branch_and_bound_depth Number of precomputed grids to use.

cartographer.mapping_2d.scan_matching.proto.RealTimeCorrelativeScanMatch

double linear_search_window Minimum linear search window in which the best possible scan alignment will be found.

double angular_search_window Minimum angular search window in which the best possible scan alignment will be found.

double translation_delta_cost_weight Weights applied to each part of the score.

double rotation_delta_cost_weight Not yet documented.

cartographer.mapping_3d.proto.KalmanLocalTrajectoryBuilderOptions

bool use_online_correlative_scan_matching Whether to solve the online scan matching first using the correlative scan matcher to generate a good starting point for Ceres.

cartographer.mapping_2d.scan_matching.proto.RealTimeCorrelativeScanMatcherOptions real_time_correlative_scan_matcher Not yet documented.

cartographer.kalman_filter.proto.PoseTrackerOptions pose_tracker_options Not yet documented.

double scan_matcher_variance Not yet documented.

double odometer_translational_variance Not yet documented.

double odometer_rotational_variance Not yet documented.

cartographer.mapping_3d.proto.LocalTrajectoryBuilderOptions

cartographer.mapping_3d.proto.MotionFilterOptions

double max_time_seconds Threshold above which a new scan is inserted based on time.

double max_distance_meters Threshold above which a new scan is inserted based on linear motion.

double max_angle_radians Threshold above which a new scan is inserted based on rotational motion.

cartographer.mapping_3d.proto.OptimizingLocalTrajectoryBuilderOptions

double high_resolution_grid_weight Not yet documented.

double low_resolution_grid_weight Not yet documented.

double velocity_weight Not yet documented.

double translation_weight Not yet documented.

double rotation_weight Not yet documented.

double odometry_translation_weight Not yet documented.

double odometry_rotation_weight Not yet documented.

cartographer.mapping_3d.proto.RangeDataInserterOptions

double hit_probability Probability change for a hit (this will be converted to odds and therefore must be greater than 0.5).

double miss_probability Probability change for a miss (this will be converted to odds and therefore must be less than 0.5).

int32 num_free_space_voxels Up to how many free space voxels are updated for scan matching. 0 disables free space.

cartographer.mapping_3d.proto.SubmapsOptions

double high_resolution Resolution of the 'high_resolution' map in meters used for local SLAM and loop closure.

double high_resolution_max_range Maximum range to filter the point cloud to before insertion into the 'high_resolution' map.

double low_resolution Resolution of the 'low_resolution' version of the map in meters used for local SLAM only.

int32 num_range_data Number of scans before adding a new submap. Each submap will get twice the number of scans inserted: First for initialization without being matched against, then while being matched.

cartographer.mapping_3d.proto.RangeDataInserterOptions range_data_inserter_options Not yet documented.

cartographer.mapping_3d.scan_matching.proto.CeresScanMatcherOptions

double translation_weight Scaling parameters for each cost functor.

double rotation_weight Not yet documented.

bool only_optimize_yaw Whether only to allow changes to yaw, keeping roll/pitch constant.

cartographer.common.proto.CeresSolverOptions ceres_solver_options Configure the Ceres solver. See the Ceres documentation for more information: https://code.google.com/p/ceres-solver/

cartographer.mapping_3d.scan_matching.proto.FastCorrelativeScanMatcherOp

int32 branch_and_bound_depth Number of precomputed grids to use.

int32 full_resolution_depth Number of full resolution grids to use, additional grids will reduce the resolution by half each.

int32 rotational_histogram_size Number of histogram buckets for the rotational scan matcher.

double min_rotational_score Minimum score for the rotational scan matcher.

double linear_xy_search_window Linear search window in the plane orthogonal to gravity in which the best possible scan alignment will be found.

double linear_z_search_window Linear search window in the gravity direction in which the best possible scan alignment will be found.

double angular_search_window Minimum angular search window in which the best possible scan alignment will be found.

cartographer.sensor.proto.AdaptiveVoxelFilterOptions

float max_length 'max_length' of a voxel edge.

float min_num_points If there are more points and not at least 'min_num_points' remain, the voxel length is reduced trying to get this minimum number of points.

float max_range Points further away from the origin are removed.

Cartographer is a system that provides real-time simultaneous localization and mapping (SLAM) in 2D and 3D across multiple platforms and sensor configurations.

CHAPTER 2

Getting started

Cartographer is a standalone C++ library. To get started quickly, use our ROS integration.

Getting started with ROS

ROS integration is provided by the Cartographer ROS repository. You will find complete documentation for using Cartographer with ROS at the Cartographer ROS Read the Docs site.

Getting started without ROS

Please see our ROS integration as a starting point for integrating your system with the standalone library. Currently, it is the best available reference.

On Ubuntu 14.04 (Trusty):

```
# Install the required libraries that are available as debs.
sudo apt-get update
sudo apt-get install -y \
    cmake \
    g++ \
    git \
    google-mock \
    libboost-all-dev \
    libcairo2-dev \
    libeigen3-dev \
    libgflags-dev \
    libgoogle-glog-dev \
    liblua5.2-dev \
    libprotobuf-dev \
    libsuitesparse-dev \
    libwebp-dev \
    ninja-build \
```

```
    protobuf-compiler \
    python-sphinx
```

```
# Build and install Ceres.
git clone https://ceres-solver.googlesource.com/ceres-solver
cd ceres-solver
mkdir build
cd build
cmake .. -G Ninja
ninja
ninja test
sudo ninja install
```

```
# Build and install Cartographer.
cd cartographer
mkdir build
cd build
cmake .. -G Ninja
ninja
ninja test
sudo ninja install
```

CHAPTER 3

System Requirements

Although Cartographer may run on other systems, it is confirmed to be working on systems that meet the following requirements:

- 64-bit, modern CPU (e.g. 3rd generation i7)
- 16 GB RAM
- Ubuntu 14.04 (Trusty) and 16.04 (Xenial)
- gcc version 4.8.4 and 5.4.0

Known Issues

- 32-bit builds have libeigen alignment problems which cause crashes and/or memory corruptions.

CHAPTER 4

How to cite us

Background about the algorithms developed for Cartographer can be found in the following publication. If you use Cartographer for your research, we would appreciate it if you cite our paper.

W. Hess, D. Kohler, H. Rapp, and D. Andor, Real-Time Loop Closure in 2D LIDAR SLAM, in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016. pp. 1271–1278.