Coxgraph: Multi-Robot Reconstruction

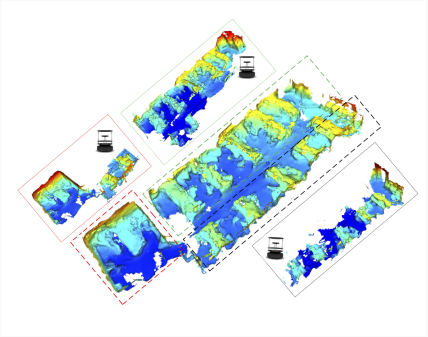

Figure 1: Coxgraph reconstruction result with 3 clients

Key features of Coxgraph:

- Multi-Robot Centralized Reconstruction
- **Low Bandwidth** by submap compression
- **Robustness** by dense-map-based loop closure outlier rejection
- **Global Consistency** by dense map optimization

Multi-Robot localization modules in this work are adapted from existing work *CORB_SLAM, VINS_CLIENT_SERVER, etc.*

Low Bandwidth by Submap Compression

TSDF Submap → Mesh Pack → Vector Trimming → **10% bw** → TSDF Submap

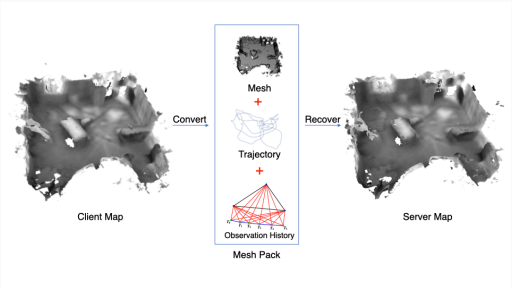

Figure 2: Components of *mesh packs*

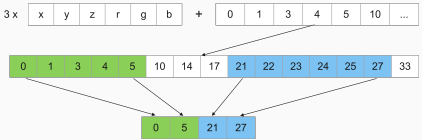

Figure 3: Index vector trimming

Robustness by Loop Closure Outlier Rejection

Two steps of outlier rejection based on dense map information:

- Trajectory collision check
- Fitness check

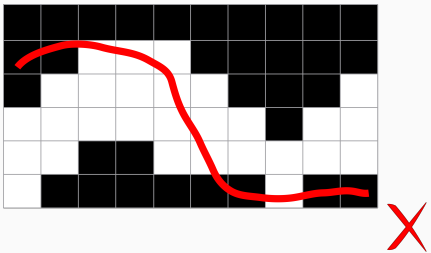

Figure 4: Trajectory collision check

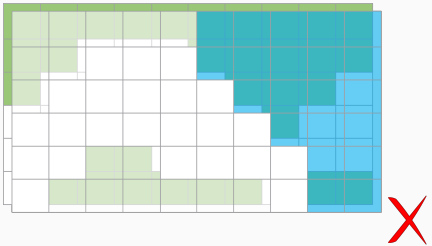

Figure 5: Fitness check

Global Consistency by Optimization

Optimize submap poses by 3 constraint types (similar to *Voxgraph*[a]):

- Loop Closure
- Odometry
- Registration(ESDF-based correspondence-free alignment)

[a]Reijgwart, Victor, et al. "Voxgraph: Globally consistent, volumetric mapping using signed distance function submaps." IEEE Robotics and Automation Letters 5.1 (2019): 227-234.

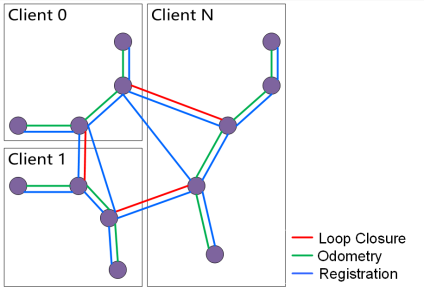

Figure 6: Pose graph

Experiment Results

Submap Compression:

- 10% 20% bandwidth usage,
- ≈0.05m RMSE on Vicon Room.

Multi-Robot Reconstruction:

- 0.116m/0.129m surface RMSE with/w.o. submap compression.

Platform Experiment:

- ≈25 KB/s for mesh packs, and ≈20KB/s for keyframes,
- 110%/180% cpu usage for submap building/odometry.

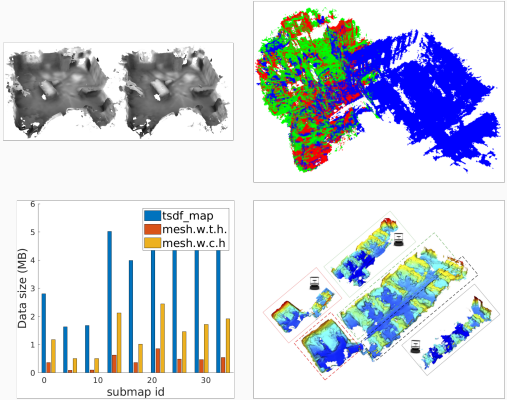

Figure 7: Experiment results