



Kinect Fusion Implementation

Real-Time Dense Reconstruction and Device Tracking

Video: https://drive.google.com/file/d/1cV9vhr9uYyCPRHhnGPZifWGO1F4ZKJZv/view?usp=share_link

Zhejin Huang

Abstract

- Implemented a dense scene reconstruction and device tracking pipeline
- Evaluated reconstruction quality, tracking quality and latency
- Optimized on these aspects
- Proposed potential further optimizations

Background

- Space Carving**
 - Needs precise pose
 - Performs poorly on concave objects
- Structure from Motion**
 - Needs human-labeled correspondences
 - Suffers from ambiguity
- Kinect Fusion**
 - No pose needed, instead estimates pose
 - Performs well on complex-shaped objects
 - Does not need correspondences
 - Limitations (satisfied by using Kinect)
 - Assumes small motion
 - Needs depth input

Method

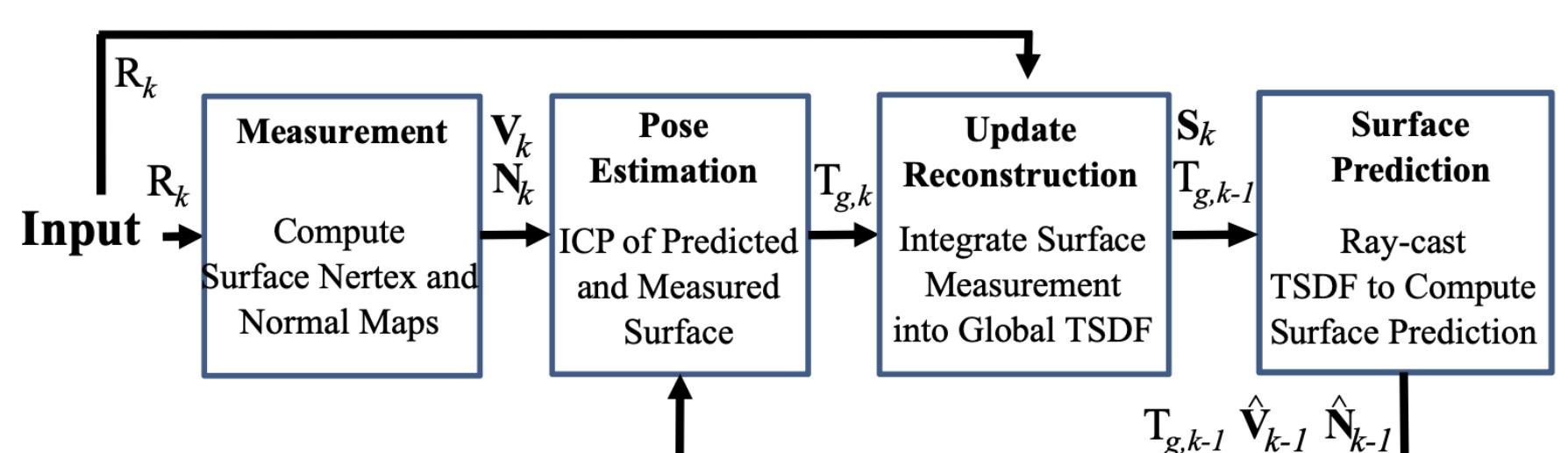


Figure 1. Kinect Fusion system workflow [1]

Results and Optimizations

- Optimized latency by down-sampling point cloud

	Overall	Depth Streaming	Surface Measurement	Pose Estimation	TSDF Update	Surface Prediction
Before	123.9 s	6.3 ms	7.0 s	110.4 s	29.7 ms	6.4 s
After	987 ms	3.6 ms	10.9 ms	133.385 ms	24.4 ms	814.1 ms

Table 1: Wall time before and after optimization

- Optimized reconstruction by filtering out invalid depths

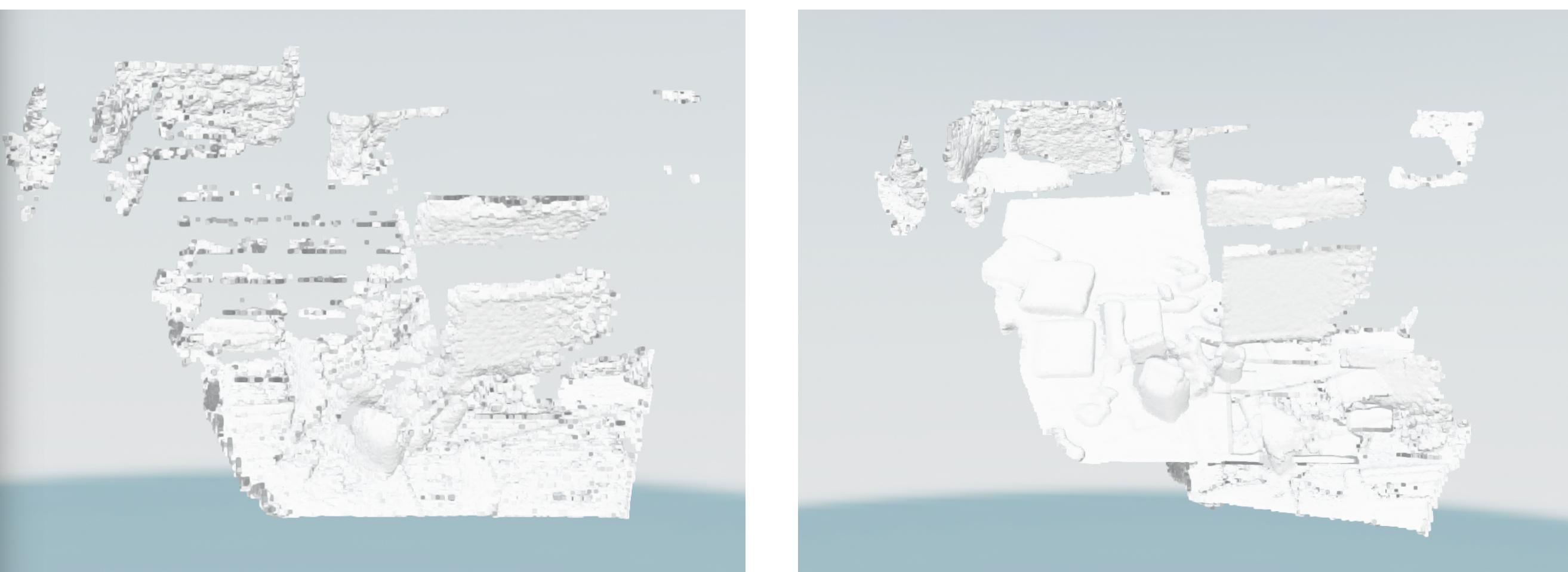


Figure 2: Noisy result at frame 10 before optimization Figure 3: Result at frame 50 after optimization

- Optimized pose estimation by ICP the delta



Figure 4 (left): Trajectories at frame 50, 100, 150, 200. Failure starts from around frame 100

Figure 5 (right): Reconstruction result at frame 100

Figure 6 (left): Trajectories at frame 100, 200, 300, 400 after optimization

Figure 7 (right): Reconstruction result at the last (798th) frame after optimization

- Remaining artifact at loop closure (from another data sequence)

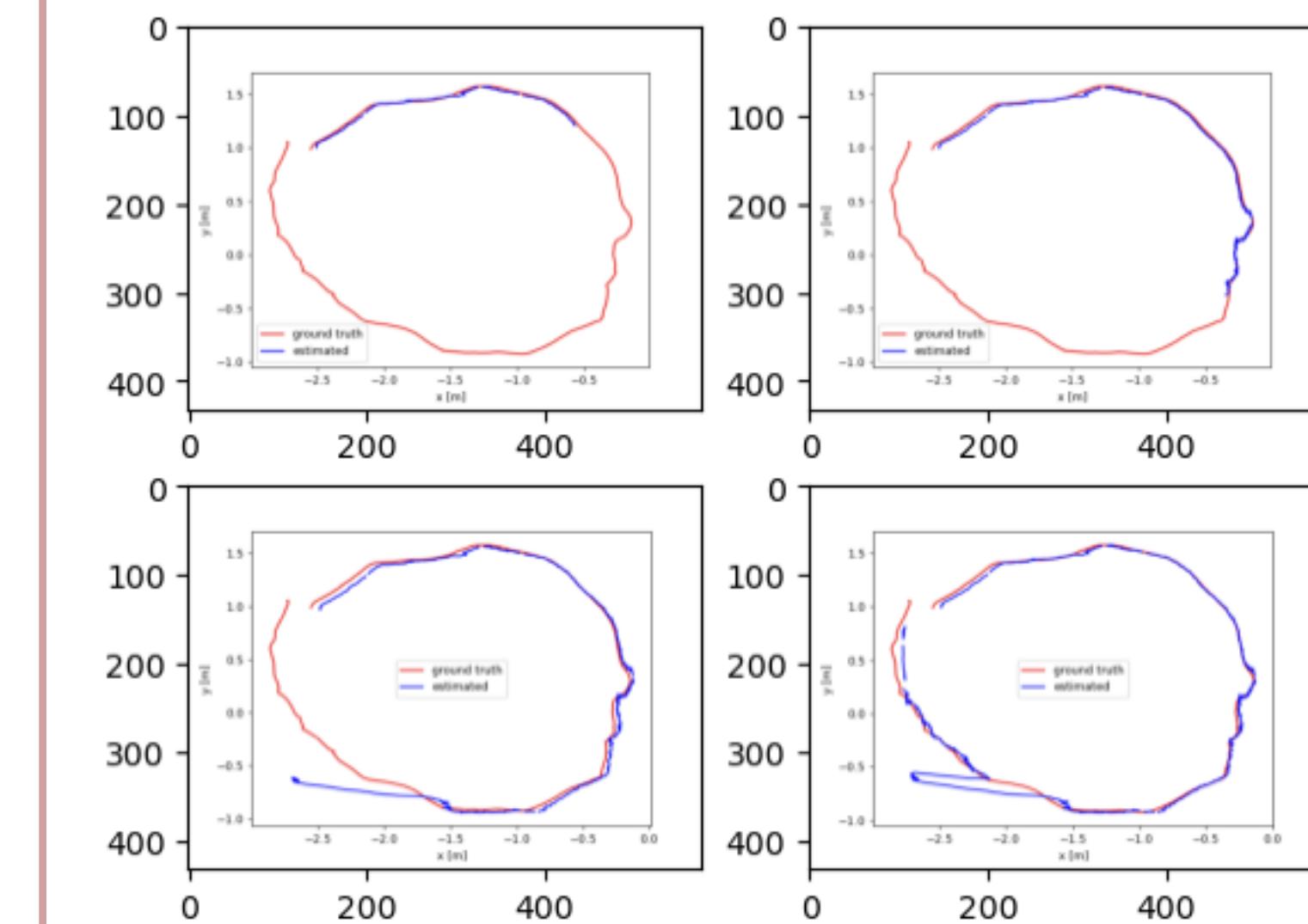


Figure 8: Trajectories at frame 250, 500, 750, 1000. Deviation starts from around frame 750



Figure 9: The incorrect Reconstruction result at frame 750

The causes of this artifact

- Pose deviated a little bit across the frames
- The camera loops back and saw a surface that has been mapped
- The small deviation causes the reconstruction to overlay on mapped surface
 - Potential solution: remove or smooth reconstruction conflicts
- The broken surface brings noise to pose estimation
 - Potential solution: use more robust pose estimation
- Both reconstruction and pose estimation fails quickly

Conclusions

- Got reasonable reconstruction and tracking results
- Proposed potential optimizations to improve robustness
 - Latency:** process on GPU to make it real-time
 - Device tracking**
 - Use tracked feature points for ICP
 - Use a higher-quality SLAM system
 - Loop closure**
 - Adjust the past trajectory at loop closure
 - Update the reconstruction accordingly

References

- [1] R. A. Newcombe et al., "KinectFusion: Real-time dense surface mapping and tracking," 2011 10th IEEE International Symposium on Mixed and Augmented Reality, Basel, Switzerland, 2011, pp. 127-136, doi: 10.1109/ISMAR.2011.6092378.