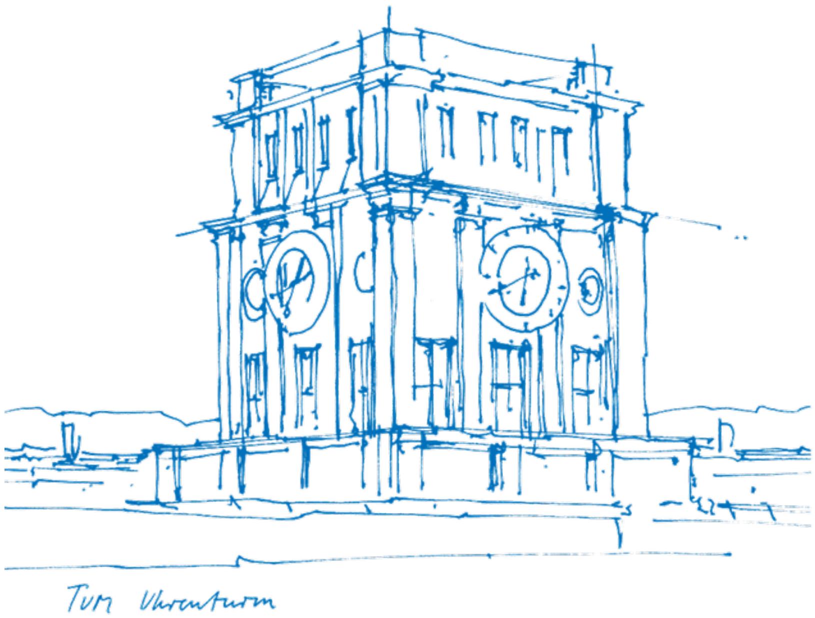
Robust Reconstruction of Indoor Scenes

Zygimantas Marcinkus
Technische Universität München
Garching bei München, 4. September 2020



The problem

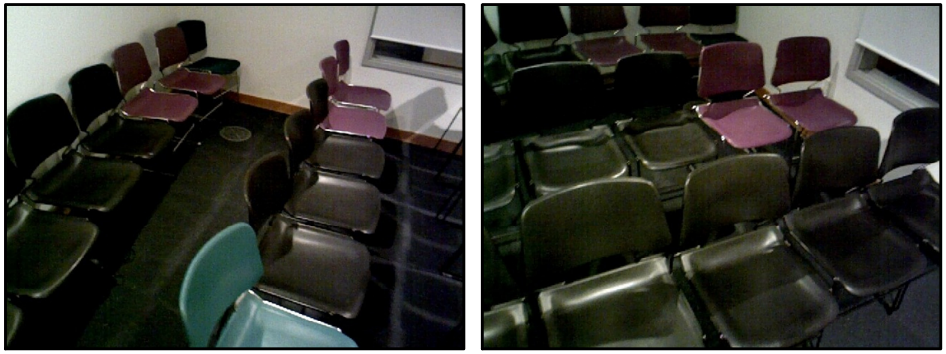


Figure 1: The real scene

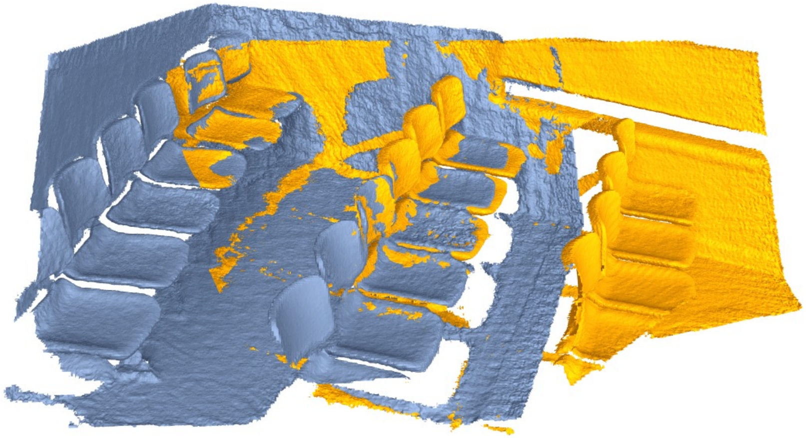


Figure 2: Reconstruction example

Input: the ICL-NUIM dataset

- RGB-D Video (augmented)

(a) Real-image (b) Depth-map

Figure 3: Real image and depth map comparison

- Ground truth scene

Input: the ICL-NUIM dataset

- RGB-D Video (augmented)
- Ground truth scene

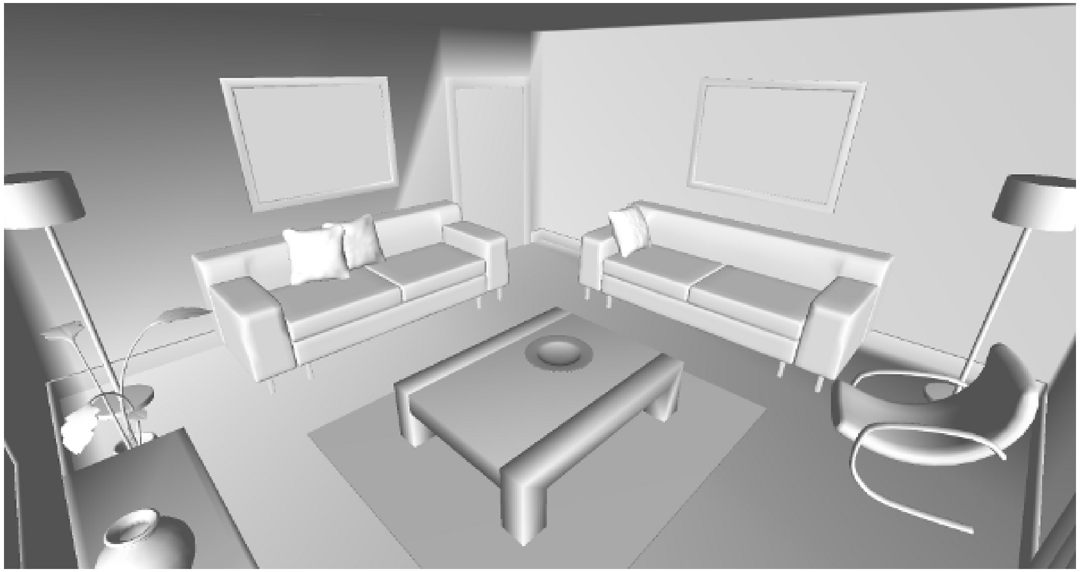


Figure 4: An example of a ground truth scene from ICL-NUIM

Challenges: odometry drift and loop closure

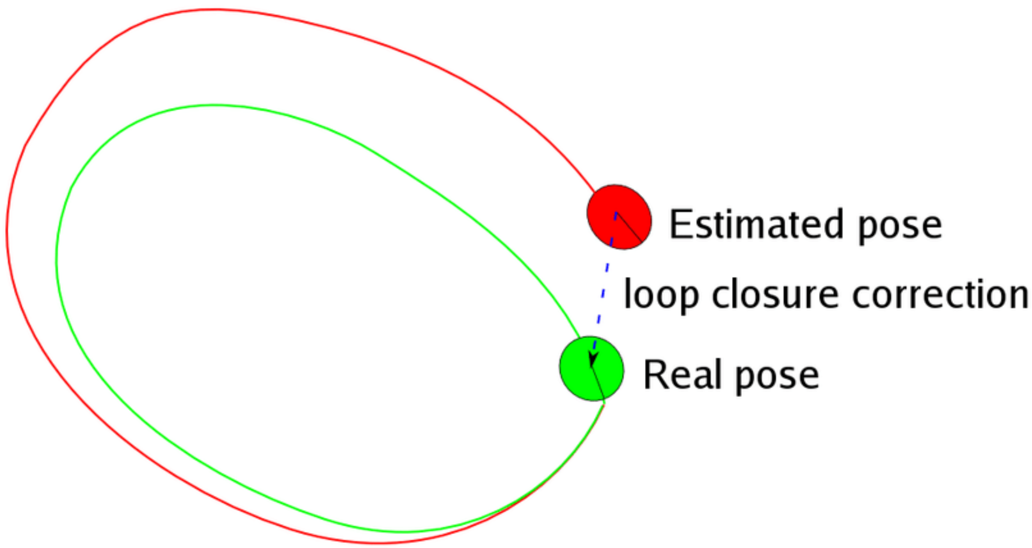


Figure 5: The general concept of loop closure

Challenges: camera noise

Figure 6: An image before and after the removal of camera noise

The paper: Robust reconstruction of indoor scenes

- Authors: Sungjoon Choi, Qian-Yi Zhou, Vladlen Koltun
- Conference on Computer Vision and Pattern Recognition 2015

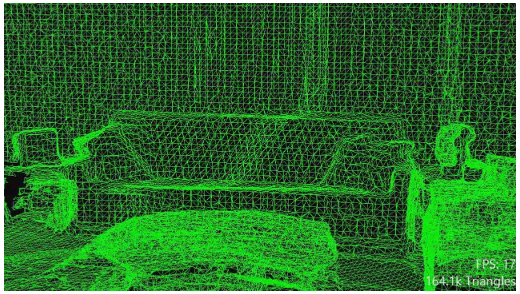
An overview of the steps for reconstruction discussed in the paper:

1. Fragment construction
2. Geometric registration
3. Robust optimization
4. Finalization

Fragment construction

Steps:

1. Divide the RGB-D video into k-frame segments ($k = 50$ in all experiments)
2. Use RGB-D odometry to estimate the camera trajectory
3. Fuse the range images to obtain a surface mesh for each segment



This results in a vertex set $P_i = \{p\}$ and a rigid transformation R_i obtained from RGB-D odometry that aligns P_i and P_{i+1} for each fragment i.

Rigid transformations and geometric registration

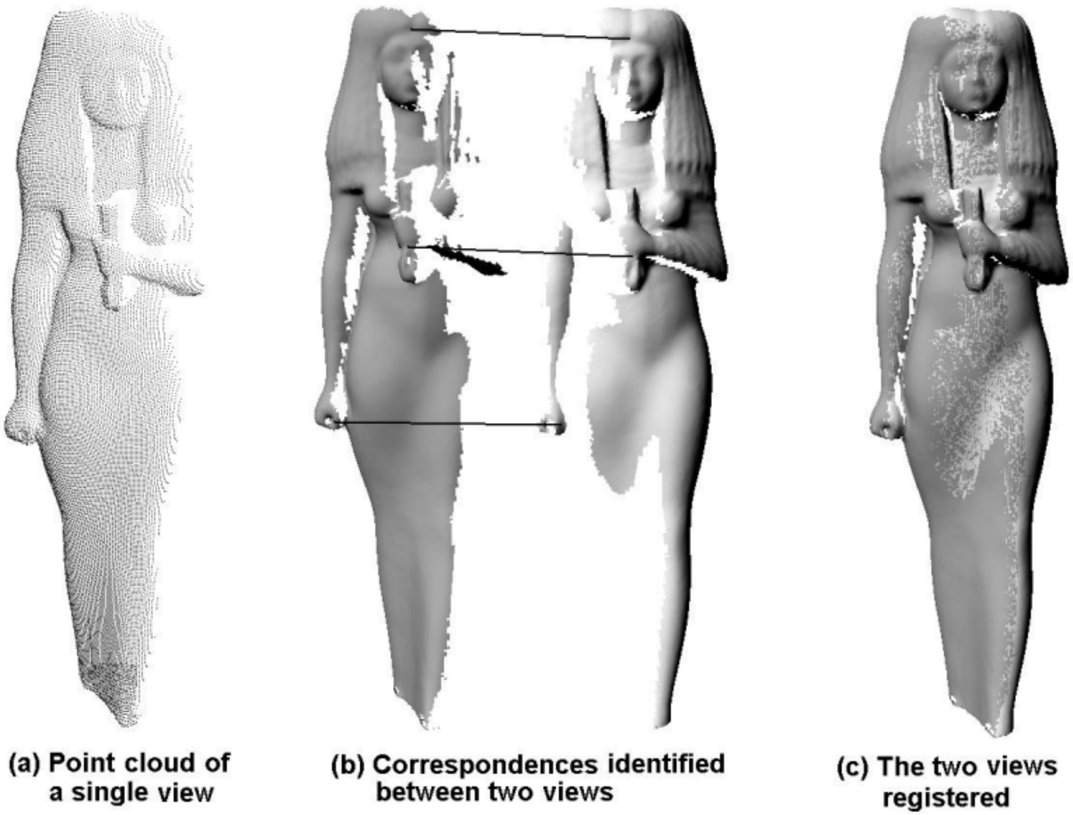


Figure 7: Visualizaion of registration with rigid transformations

Geometric registration: analysis of the algorithms

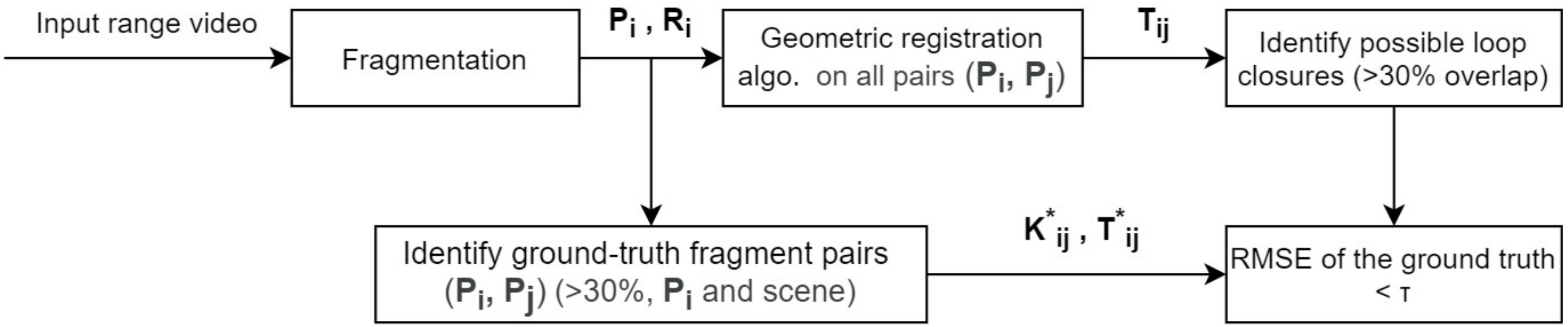


Figure 8: A graph representing the process of choosing the algorithm used in the paper

$$\frac{1}{|K_{ij}^*|} \sum_{(\mathbf{p}^*,\mathbf{q}^*)} ||T_{ij}p^* - q^*||^2 < \tau = 0.2^2 \tag{1}$$

- T_{ij} - analysed transformation
- T_{ij}^* - ground-truth transformation
- K_{ij}^* - set of point-to-point correspondences

Geometric registration: analysis of the algorithms

	OpenCV	4PCS	Super 4PCS	PPF Integral	PCL	PCL modified
Recall (%)	5.3	20.0	17.8	32.5	44.9	**59.2**
Precision (%)	1.6	8.9	10.4	7.1	14.0	**19.6**
Runtime (sec)	10	380	62	83	**3**	8

Table 1. Performance of geometric registration algorithms. Average running time for aligning two fragments was measured using a single thread on an Intel Core i7-3770 CPU clocked at 3.5 GHz.

Geometric registration: PCL modified

input : A pair of fragments $(\mathbf{P}_i, \mathbf{P}_j)$
output : Transformation $\mathbf{T}_{ij}$ and correspondence set $\mathcal{K}_{ij}$

Downsample $\mathbf{P}_i = \{\mathbf{p}\}$ and $\mathbf{P}_j = \{\mathbf{q}\}$;
Compute normals $\{\mathbf{n_p}\}$ and $\{\mathbf{n_q}\}$;
Compute FPFH features $\{\mathbf{F}(\mathbf{p})\}$ and $\{\mathbf{F}(\mathbf{q})\}$;
$\mathbf{T}_{ij} \leftarrow \emptyset$, $\mathcal{K}_{ij} \leftarrow \emptyset$;
max_correspondences $\leftarrow 0$;

for $i \leftarrow 1$ **to** *max_iteration* **do**

- *// RANSAC iteration*;
- Randomly pick four points $(\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ from $\mathbf{P}_i$;
- Find matching samples $(\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ on $\mathbf{P}_j$ using equation (1);
- Compute transformation $\mathbf{T}$ that aligns these two sets of samples;
- *// Validation*;
- **if** $\angle(\mathbf{T}\mathbf{n}_{\mathbf{p}_k}, \mathbf{n}_{\mathbf{q}_k}) > 30^\circ$ **then**
  - continue;
- **if** $\|\mathbf{p}_k - \mathbf{p}_{k+1}\| < 0.9\,\|\mathbf{q}_k - \mathbf{q}_{k+1}\|$ or vice versa **then**
  - continue;
- Compute correspondences $\mathcal{K}$ between $\mathbf{T}\mathbf{P}_i$ and $\mathbf{P}_j$;
- **if** $|\mathcal{K}| < \frac{1}{3}\min(|\mathbf{P}_i|, |\mathbf{P}_j|)$ **then**
  - continue;
- *// Update*;
- **if** $|\mathcal{K}| >$ *max_correspondences* **then**
  - $\mathbf{T}_{ij} \leftarrow \mathbf{T}$, $\mathcal{K}_{ij} \leftarrow \mathcal{K}$;
  - max_correspondences $\leftarrow |\mathcal{K}|$;

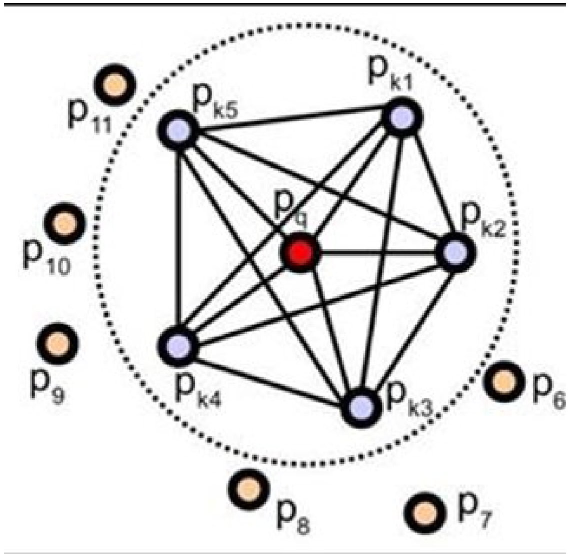


Figure 9: Influence region diagram for the point p_q

$$\mathbf{q_p} = \arg\min_{\mathbf{q} \in \mathbf{P}_j} \|\mathbf{F}(\mathbf{p}) - \mathbf{F}(\mathbf{q})\|^2.$$

Figure 10: Equation 1 in the algorithm

Robust optimization: pose graph

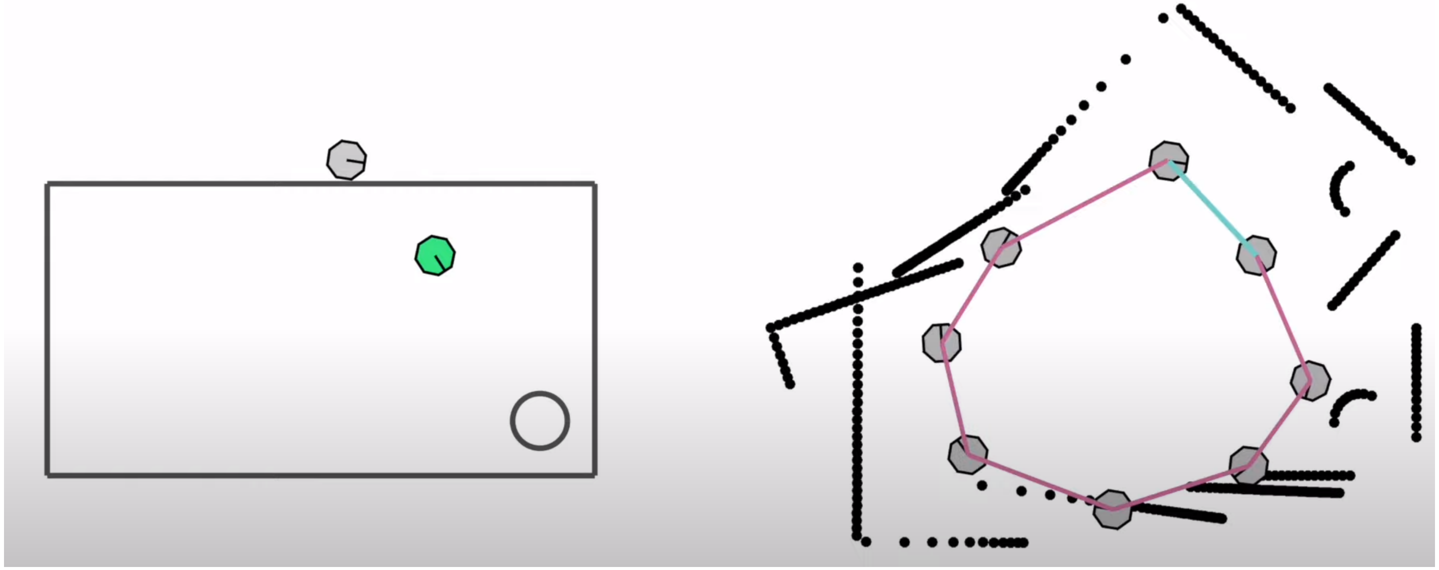


Figure 11: Pose Graph before optimization.

Robust optimization: pose graph

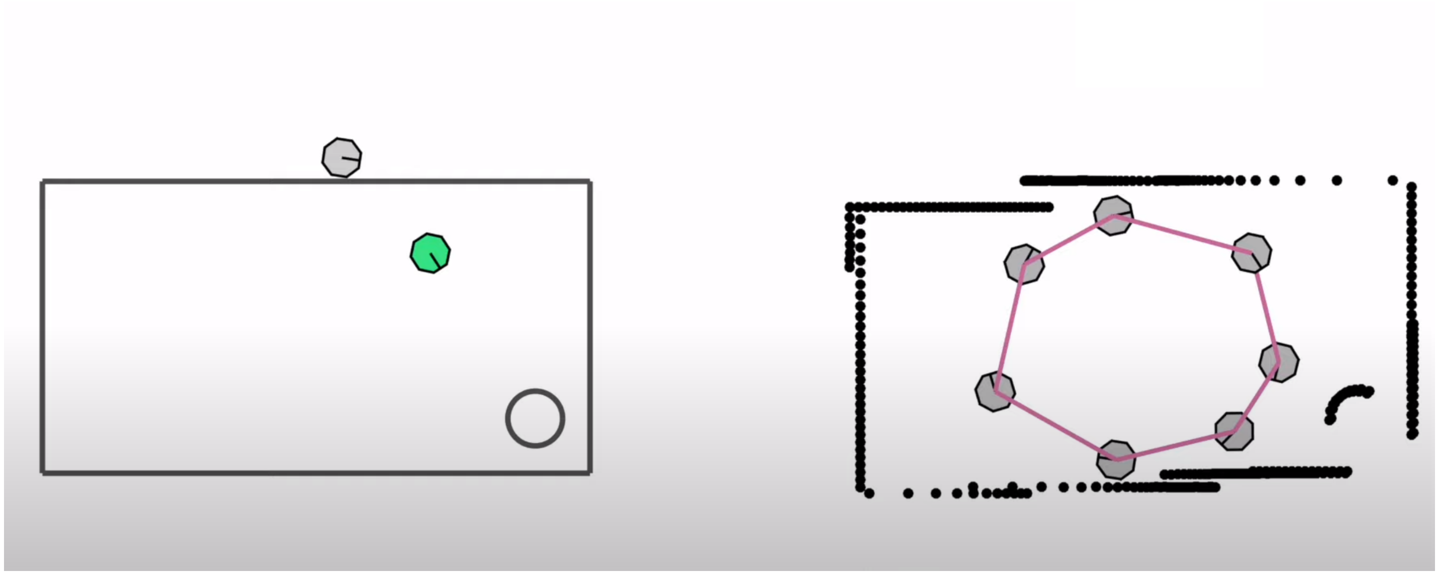


Figure 12: Pose graph after optimization.

Robust optimization: pose graph

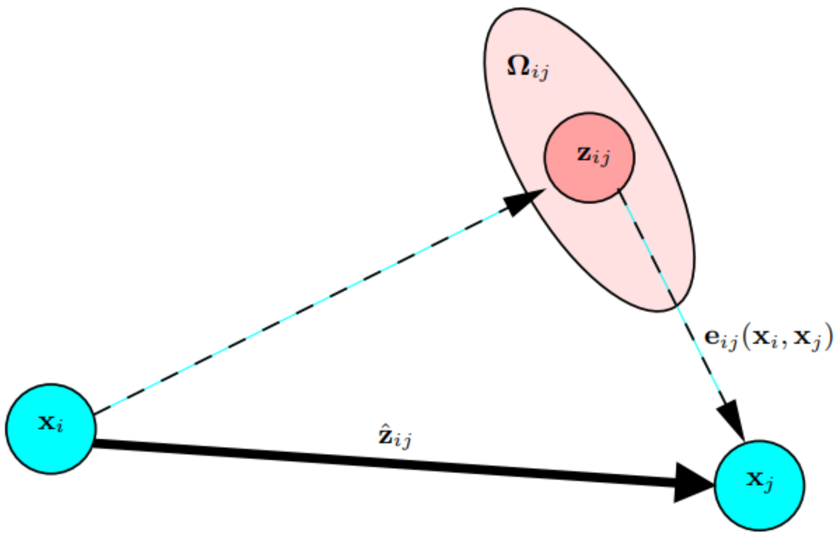


Figure 13: Aspects of an edge connecting the vertex x_i and the vertex x_j.

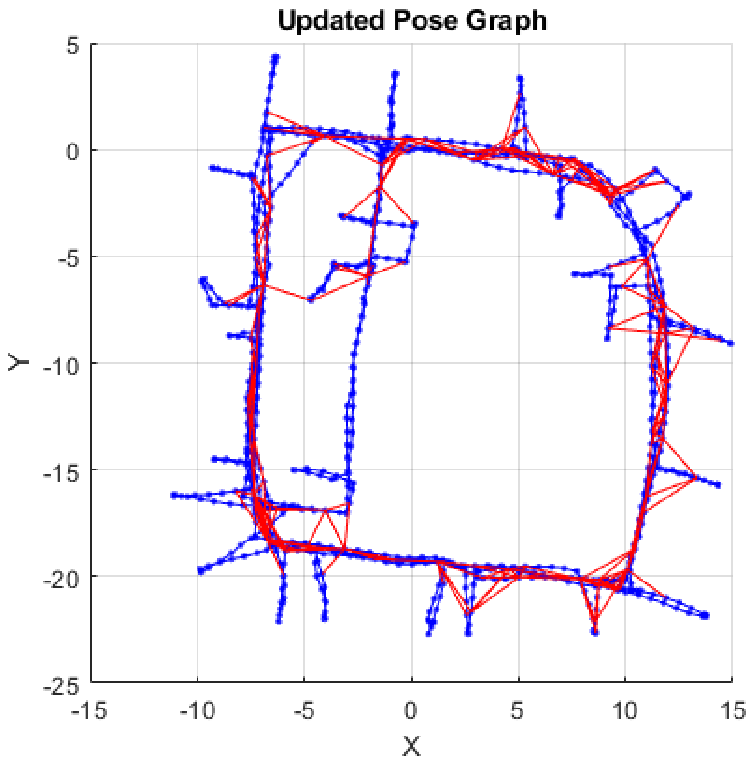


Figure 14: Example of a pose graph

Robust optimization: pose graph

Output from previous steps:

- Fragments $\{P_i\}$
- Transformations $\{R_i\}$ and T_{ij}

Goal: compute localized set of poses $\{T_i\}$. Concretely, solve:

$$E(\mathbb{T}) = \sum_i f(\mathbf{T}_i, \mathbf{T}_{i+1}, \mathbf{R}_i) + \sum_{i,j} f(\mathbf{T}_i, \mathbf{T}_j, \mathbf{T}_{ij}). \quad (1)$$

,

with $f(T_i, T_j, X)$ measuring inconsistencies between T_i and T_j and the relative pose X.

Robust optimization: alignment term f

$$
\begin{aligned}
f(\mathbf{T}_i, \mathbf{T}_j, \mathbf{X}) &= \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \|\mathbf{T}_i\mathbf{p} - \mathbf{T}_j\mathbf{q}\|^2 & (3)\\
&\approx \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \|\mathbf{T}_i\mathbf{p} - \mathbf{T}_j\mathbf{X}\mathbf{p}\|^2 & (4)\\
&= \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \|\mathbf{X}^{-1}\mathbf{T}_j^{-1}\mathbf{T}_i\mathbf{p} - \mathbf{p}\|^2. & (5)
\end{aligned}
$$

Figure 15: First glance at the alignment term with K_{ij} as correspondences between XP_i and P_j within $\varepsilon = 0.05$m.

Robust optimization: computing f

Local parametrization representing $X^{-1}T_j^{-1}T_i$ as $\xi = (\omega, t) = (\alpha, \beta, \gamma, a, b, c)$ and $T_j^{-1}T_i \approx X_i^{-1}$ yields

$$\mathbf{X}^{-1}\mathbf{T}_j^{-1}\mathbf{T}_i \approx \begin{pmatrix} 1 & -\gamma & \beta & a \\ \gamma & 1 & -\alpha & b \\ -\beta & \alpha & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{6}$$

$$\mathbf{X}^{-1}\mathbf{T}_j^{-1}\mathbf{T}_i\mathbf{p} \approx \mathbf{p} + \omega \times \mathbf{p} + \mathbf{t}.$$

Figure 16: Local parametrization of $X^{-1}T_j^{-1}T_i$ results

Robust optimization: computing f

Using approximation (6) in f results in (with $-[p]_x$ is the skew-symmetric matrix form of the cross-product with p, and I is the 3×3 identity matrix:

$$\begin{aligned} f(\mathbf{T}_i, \mathbf{T}_j, \mathbf{X}) &\approx \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \|\omega \times \mathbf{p} + \mathbf{t}\|^2 \\ &= \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \left\| \left[- [\mathbf{p}]_\times \mid \mathbf{I}\right] \xi \right\|^2, \quad (7) \\ &= \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \xi^\top \mathbf{G}_\mathbf{p}^\top \mathbf{G}_\mathbf{p} \xi \\ &= \xi^\top \left(\sum_{(\mathbf{p},\mathbf{q})\in\mathcal{K}_{ij}} \mathbf{G}_\mathbf{p}^\top \mathbf{G}_\mathbf{p} \right) \xi. \quad (8) \end{aligned}$$

with $G_p = [-[p]_x | I]$

Robust optimization: line process


(b) Optimization with all pairwise alignments


(c) Optimization with line processes

Figure 17: ICL-NUIM living room 1 with (c) and without (b) line process

Robust optimization: line process

Add a line process $L = \{l_{ij}\}$ with penalty function $\Psi(l) = (\sqrt{l} - 1)^2$:

$$\begin{aligned} E(\mathbb{T}, \mathbb{L}) &= \sum_i f(\mathbf{T}_i, \mathbf{T}_{i+1}, \mathbf{R}_i) \\ &+ \sum_{i,j} l_{ij} f(\mathbf{T}_i, \mathbf{T}_j, \mathbf{T}_{ij}) \\ &+ \mu \sum_{i,j} \Psi(l_{ij}). \end{aligned}$$

Figure 18: The objective including a line process

- $l_{ij} \to 0$ means $\Psi(l_{ij}) \to 1$
- $l_{ij} \to 1$ means $\Psi(l_{ij}) \to 0$
- $\mu = \tau^2 k$ where k - average cardinality of correspondance sets K_{ij} and $\tau = 0.2^2$

Finalization

1. use g^2o to optimize objective (2)
2. prune loop closures with $l_{ij} < 0.25$
3. refine using ICP
4. use pose graph optimization to obtain final fragment poses
5. (optional) use nonrigid refinement
6. fuse into a global mesh by volumetric integration

	Before pruning		After pruning	
	Recall (%)	Precision (%)	Recall (%)	Precision (%)
Living room 1	61.2	27.2	57.6	95.1
Living room 2	49.7	17.0	49.7	97.4
Office 1	64.4	19.2	63.3	98.3
Office 2	61.5	14.9	60.7	100.0
Average	59.2	19.6	57.8	97.7

Table 2. The effect of robust optimization. The optimization increases the average precision of the loop closure set from 19.6% to 97.7%.

Results: synthetic scenes

	Kintinuous	DVO SLAM	SUN3D SfM	Ours	GT trajectory
Living room 1	0.22	0.21	0.09	**0.04**	0.04
Living room 2	0.14	0.06	0.07	**0.07**	0.04
Office 1	0.13	0.11	0.13	**0.03**	0.03
Office 2	0.13	0.10	0.09	**0.04**	0.03
Average	0.16	0.12	0.10	**0.05**	0.04

Table 4. Reconstruction accuracy on ICL-NUIM sequences. Mean distance of each reconstructed model to the ground-truth surface (in meters). Our approach reduces the average error by a factor of 2 relative to the closest alternative approach.

Results: real-world scenes

- No ground-truth scene
- Pair-wise comparisons for each input sequence
- BRE as a measure ($[-1; 1]$)

	Kintin-uous	DVO SLAM	SUN3D SfM	Ours	GT trajectory
Living room 1	-0.53	-0.90	0.02	**0.47**	0.94
Living room 2	-0.89	-0.65	-0.13	**0.66**	0.89
Office 1	-0.71	-0.41	-0.15	**0.09**	0.98
Office 2	-0.83	-0.57	-0.11	**0.58**	0.90
Average	-0.74	-0.63	-0.09	**0.45**	0.93

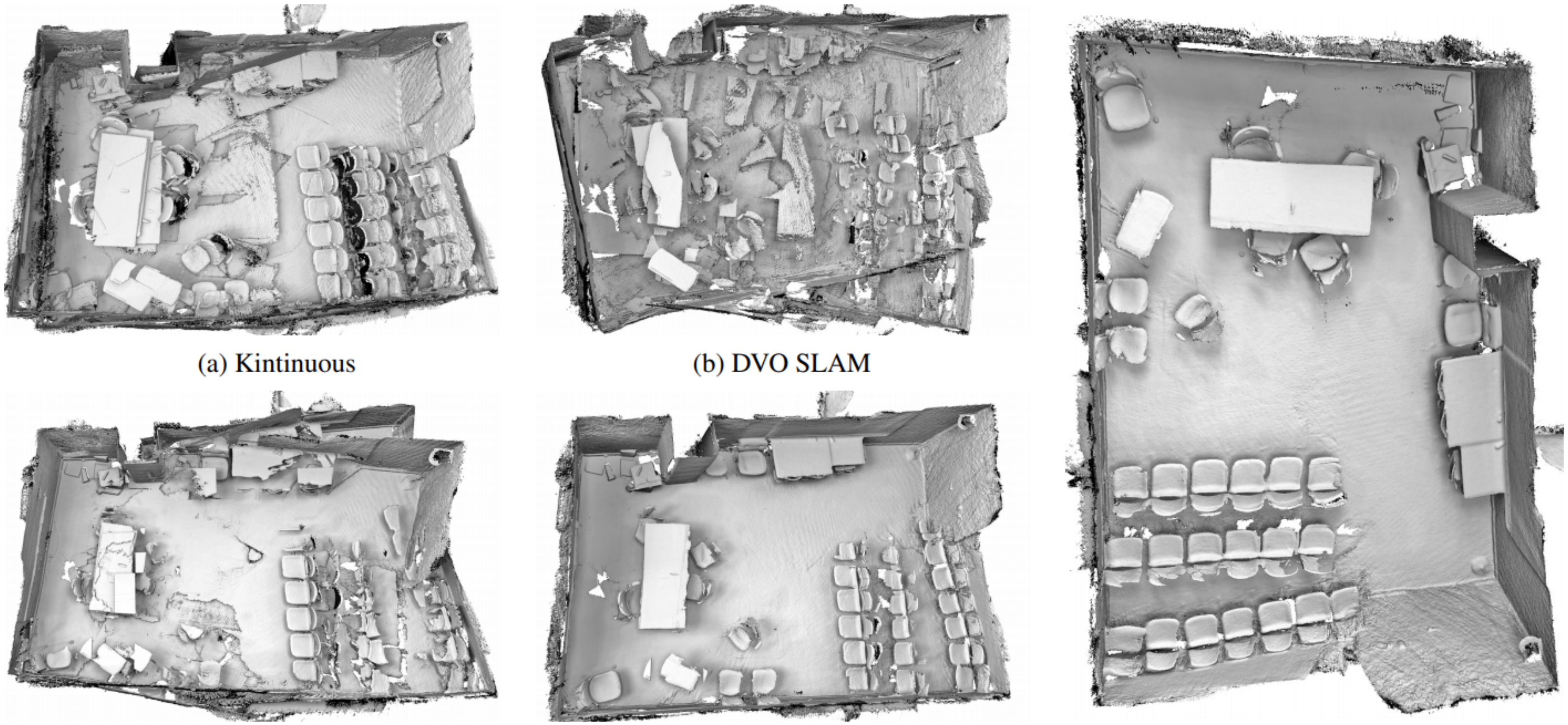
Figure 19: ICL-NUIM scenes evaluated with this procedure

Results: real-world scenes

	DVO SLAM	Kintin-uous	SUN3D SfM	Ours	SUN3D manual
hotel_umd	-0.61	-0.45	-0.02	**0.66**	0.56
harvard_c5	-0.49	-0.01	-0.65	**0.94**	0.11
harvard_c6	-0.97	0.05	-0.01	**0.96**	-0.15
harvard_c8	-0.70	-0.61	0.39	**0.65**	0.46
mit_32_d507	-0.78	-0.28	-0.02	**0.74**	0.36
mit_76_studyroom	-0.52	-0.47	0.35	**0.50**	0.19
mit_dorm_next_sj	-0.26	-0.20	-0.23	**0.10**	0.65
mit_lab_hj	-0.12	-0.57	0.03	**0.22**	0.50
Average	-0.56	-0.32	-0.02	**0.60**	0.33

Table 7. Perceptual evaluation on SUN3D scenes. BRE scores computed from pairwise comparisons performed on Amazon Mechanical Turk. The presented approach outperforms all other automatic reconstruction pipelines.

Results: real-world scenes



(a) Kintinuous

(b) DVO SLAM

(c) SUN3D SfM

(d) Our result

(e) Optional non-rigid refinement

Figure 4. Reconstruction of the mit_32_d507 scene from the SUN3D dataset. (a) Reconstruction produced by Kintinuous [61]. (b) Reconstruction produced by DVO SLAM [34]. (c) Reconstruction produced by the off-line RGB-D structure-from-motion pipeline of Xiao et al. [65]. (d) Reconstruction produced by our approach. (e) An optional non-rigid refinement of our result using SLAC [71].

Summary and Notes

- Reconstruction using global optimization based on line processes
- Robust to erroneous geometric alignments
- Significant accuracy increase
- Potential problem: no actual loop closures
- Potential problem 2: catastrophic odometry failure