PlaneMatch: Patch Coplanarity Prediction for Robust RGB-D Registration Supplemental Material

Anonymous ECCV submission

Paper ID 184

1 Outline

In this supplemental material, we provide the following additional information and results:

- Section 2 provides an overview of the dataset of our coplanarity benchmark (COP).
- Section 3 gives more evaluation results for our coplanarity network, including a comparison of different masking schemes (Section 3.1), evaluation on patch pairs proposed from real cases of scene reconstruction (Section 3.2), and visual qualitative results of coplanarity matching (Section 3.3).
- Section 4 provides more evaluations of the reconstruction algorithm. Specifically, we evaluate the robustness of the registration against the initial ratio of incorrect pairs (Section 4.1), and we show more visual results of reconstructions for scenes from various datasets.
- Section 5 discusses the limitations of our method.
- Finally, Section 6 provides the formulation for a variatnt of our method that only utilizes coplanarity constraints (Section 6.1), the optimization procedure used for that variant (Section 6.2), and the stability analysis used for achieving a robust optimization in that variant (Section 6.3).

2 COP Benchmark Dataset

Figure 1 and 2 provide an overview of our coplanarity benchmark datasets, COP-S (organized in decreasing patch size) and COP-D (in increasing patch distance), respectively. For each subset, we show both positive and negative pairs, each with two pairs. Note how non-trivial the negative pairs are in our dataset, for example, the negative pairs of S3 and D1.

Fig. 1: An overview of the patch pairs (both positive and negative) in the benchmark dataset COP-S. The dataset is organized according to patch size. S1: 0.25~10 m^2. S2: 0.05~0.25 m^2. S3: 0~0.05 m^2.

Fig. 2: An overview of the patch pairs (both positive and negative) in the benchmark dataset COP-D. The dataset is organized according to pair distance. D1: 0~0.3 m. D2: 0.3~1 m. D3: 1~5 m.

3 Network Evaluations

This section provides further studies and evaluations of the performance of our coplanarity prediction network.

3.1 Different Masking Schemes

We first investigate several alternative masking schemes for the local and global inputs of our coplanarity network. The proposed masking scheme is summarized as follows (see Figure 3 (right)). The local mask is binary, with the patch of interest in white and the rest of the image in black. The global mask, in contrast, is continuous, with the patch of interest in white and then a smooth decay to black outside the patch boundary.

We compare in Figure 3 our masking scheme (global decay) with several alternatives including 1) using distance-based decaying for both local and global scale (both decay), 2) using distance-based decaying only for local scale (local decay), 3) without decaying for either scale (no decay), and 4) without using a mask at all (no mask). Over the entire COP-D benchmark dataset, we test the above methods and plot the PR curves. The results demonstrate the advantage of our specific design choice of masking scheme (using decaying for global scale but not for local).

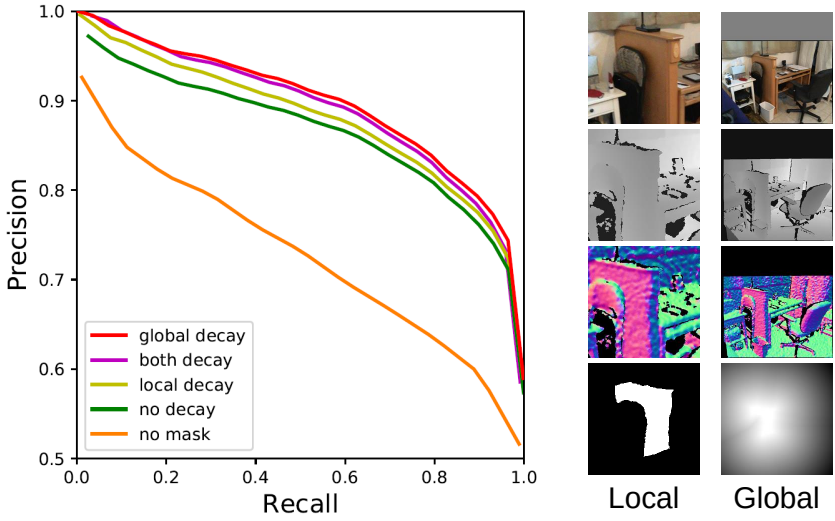

Fig. 3: Comparison of different masking schemes on the entire COP-D dataset. 'Global decay' is our scheme.

3.2 Performance on Patches Proposed during Reconstructions

Our second study investigates the network performance for a realistic balance of positive and negative patch pairs. The performance of our coplanarity network has so far been evaluated over the COP benchmark dataset, which contains comparable numbers of positive and negative examples. To evaluate its performance in a reconstruction setting, we test on patch pairs proposed during the reconstruction of two scenes (the full sequence of 'fr1/desk' and 'fr2/xyz' from the TUM dataset). The ground-truth coplanarity matching is detected based on the ground-truth alignment provided with the TUM dataset.

Figure 4 shows the plot of PR curves for both intra- and inter-fragment reconstructions. The values for intra-fragment are averaged over all fragments. For patches from the real case of scene reconstruction, our network achieves a precision of $> 20\%$, when the recall rate is 80%. This accuracy is sufficient for our robust optimization for frame registration, which can be seen from the evaluation in Figure 6; see Section 4.1.

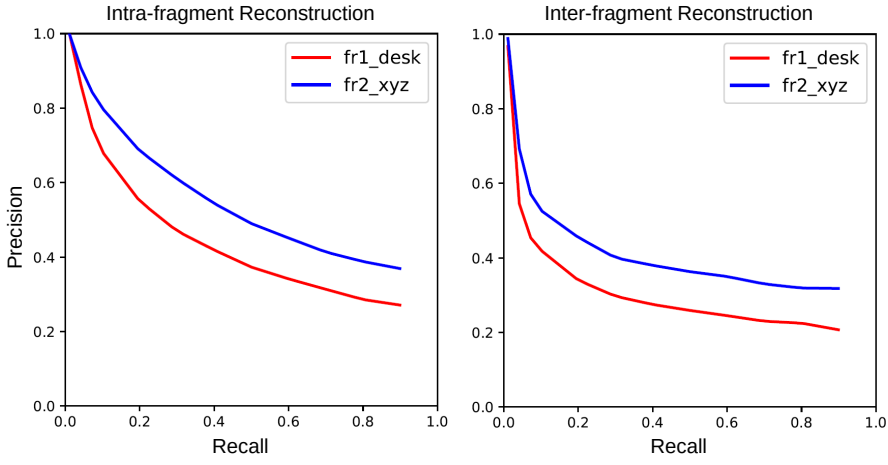

Fig. 4: Performance of our coplanarity network on patch pairs proposed from the reconstruction of sequences 'fr1/desk' and 'fr2/xyz' from the TUM dataset. The PR curves for both intra- (left) and inter-fragment (right) reconstruction are shown.

3.3 More Visual Results of Coplanarity Matching

Figure 5 shows some visual results of coplanarity matching. Given a query patch in one frame, we show all patches in another frame, which are color-coded with the dissimilarity predicted by our coplanarity network (blue is small and red is large). The results show that our network produces correct coplanarity embedding, even for patches observed across many views.

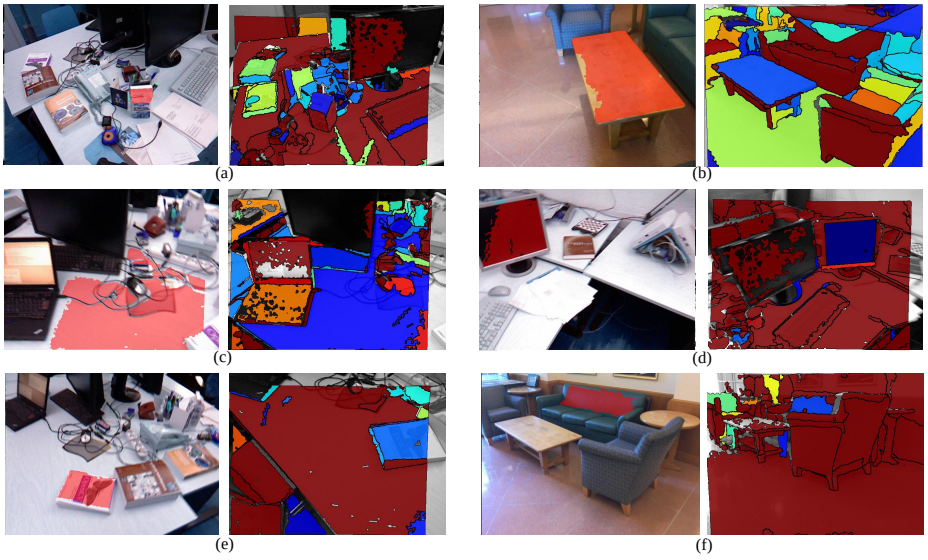

Fig. 5: Visualization of coplanarity matching for six query patches. For each example, the query patch is selected in the left image. In the right image, all patches are color-coded with the dissimilarity predicted by our coplanarity network (blue is small and red is large).

4 Reconstruction Evaluations

4.1 Robustness to Initial Coplanarity Accuracy

To evaluate the robustness of our optimization for coplanarity-based alignment, we inspect how tolerant the optimization is to the initial accuracy of the coplanarity prediction. In Figure 6, we plot the reconstruction error of our method on two sequences (full) from TUM dataset, with varying ratio of incorrect input pairs. In our method, given a pair of patches, if their feature distance in the embedding space is smaller than 2.5, it is used as a hypothetical coplanar pair being input to the optimization. The varying incorrect ratios are thus obtained via gradually introducing more incorrect predictions by adjusting the feature distance threshold.

Reconstruction error is measured by the absolute trajectory error (ATE), i.e., the root-mean-square error (RMSE) of camera positions along a trajectory. The results demonstrate that our method is quite robust against the initial precision of coplanarity matching, for both intra- and inter-fragment reconstructions. In particular, the experiments show that our method is robust for a precision 20% (incorrect ratio of 80%), while keeping the recall rate no lower than 80%.

4.2 More Visual Results of Reconstruction

Figure 7 shows more visual results of reconstruction on 17 sequences, including 9 from the ScanNet dataset [1] and 8 new ones scanned by ourselves. The sequences

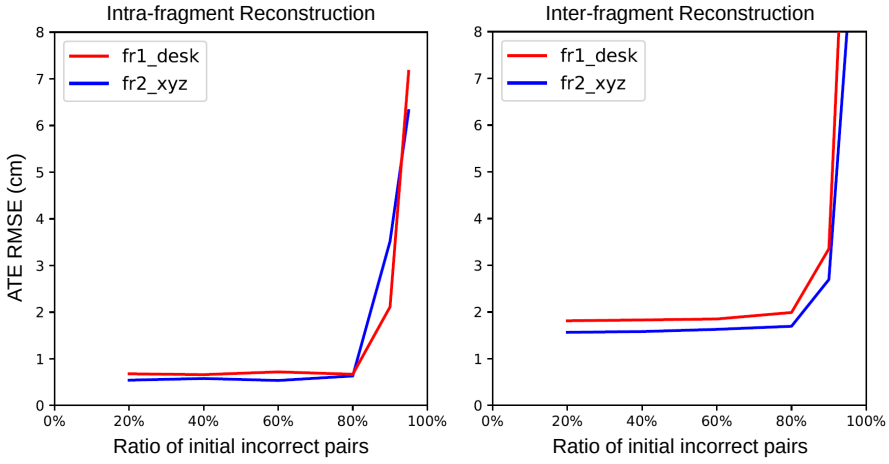

Fig. 6: Evaluation of the robustness of our coplanarity-based alignment on sequences 'fr1/desk' and 'fr2/xyz' from the TUM dataset. The plots shows the ATE RMSE (in cm) over different precisions. The results for both intra- (left) and inter-fragment (right) reconstruction are shown.

scanned by ourselves have very sparse loop closure due the missing parts. Our method works well for all these examples. Figure 8 shows the reconstruction of 4 sequences from the Sun3D dataset [2]. Since the registration of Sun3D sequences is typically shown without fusion in previous works (e.g., [2, 3]), we only show the point clouds.

5 Limitations and Failure Cases

Our work has several limitations, which suggest topics for future research.

First, coplanarity correspondences alone are not always enough to constrain camera poses uniquely in some environments – e.g., the pose of a camera viewing only a single flat wall will be under-constrained. Therefore, coplanarity is *not* a replacement for traditional features, such as key-points, lines, etc.; rather, we argue that coplanarity constraints provide additional signal and constraints which are critical in many scanning scenarios, thus helping to improve the reconstruction results. This becomes particularly obvious in scans with a sparse temporal sampling of frames.

Second, for the cases where short-range coplanar patches dominate long-range ones (e.g., a bending wall), our method could reconstruct an overly flat surface due to the coplanarity regularization by false positive coplanar patch pairs between adjacent frames. For example, in Figure 9, we show a tea room scanned by ourselves. The top wall is not flat, but the false positive coplanar pairs detected between adjacent frames could over-regularize the registration, making it mistakenly flattened. This in turn causes the loop cannot be closed at the wall in the bottom.

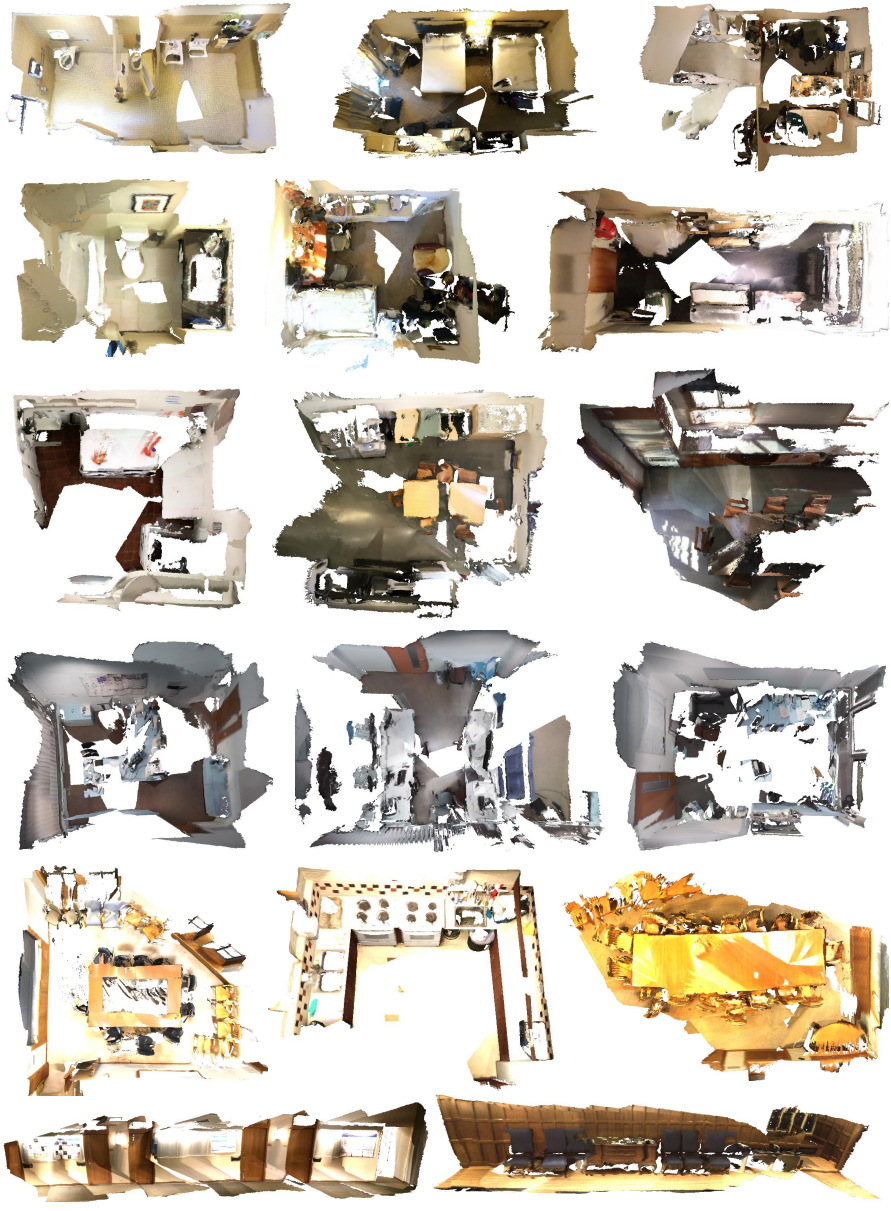

Fig. 7: Reconstruction results on 17 sequences, including 9 from ScanNet [1] (first three rows) and 8 scanned by ourselves (last three rows).

Third, our optimization is currently a computational bottleneck – it takes approximately 20 minutes to perform the robust optimization in typical scans shown in the paper. Besides exploiting the highly parallelizable intra-fragment

Fig. 8: Reconstruction results of four sequences from the Sun3D dataset [2].

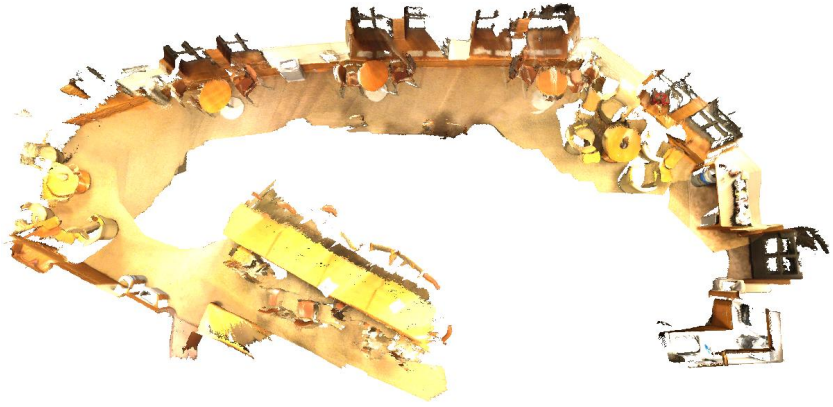

Fig. 9: The coplanarity constraint could cause over-regularization: A curvy wall (top) is mistakenly flattened causing the loop cannot be closed at the bottom wall for which long-range coplanarity is not available.

registrations, a more efficient optimization is a worthy direction for future investigation.

6 Coplanarity-only Robust Registration

At lines 526-529 of the main paper and in Table 1b, we provide an ablation study in which our method is compared to a variant (called "Coplanarity only") that uses only predicted matches of coplanar patches to constrain camera poses – i.e., without keypoint matches. In order to produce that one comparative result, we implemented an augmented version of our algorithm that includes a new method for selecting coplanar patch pairs in order to increase the chances of fully constraining the camera pose DoFs with the seletected patch pairs. The following subsections describe that version of the algorithm. Although it is not part of our method usually, we describe it in full here for the sake of reproducibility of the "Coplanarity only" comparison provided in the paper.

6.1 Formulation

Objective Function: The objective of coplanarity-only registration contains three terms, including the *coplanarity data term* (Equation (3) of the main paper), the *coplanarity regularization term* (Equation (4) of the main paper), and a newly introduced *frame regularization term* for regularizing the optimization based on the assumption that the transformation between adjacent frames is small:

$$E(T, s) = E_{\text{data-cop}}(T, s) + E_{\text{reg-cop}}(s) + E_{\text{reg-frm}}(T) \tag{1}$$

The frame regularization term $E_{\text{reg-frm}}$ makes sure the system is always solvable, by weakly constraining the transformations of adjacent frames to be as close as possible:

$$E_{\text{reg-frm}}(T) = \lambda \sum_{i \in \mathcal{F}} \sum_{\mathbf{v} \in \mathcal{V}_i} ||\mathbf{T}_i \mathbf{v} - \mathbf{T}_{i+1} \mathbf{v}||^2, \tag{2}$$

where $\mathcal{V}_i$ is a sparse set of points sampled from frame i. λ is set to 0.001 by default.

When using coplanarity constraints only (without key-points), our coplanarity-based alignment may become under-determined or unstable along some DoF, when there are too few coplanar patch pairs that can be used to pin down that DoF. In this case, we must be more willing to keep pairs constraining that DoF, to keep the system stable. To this end, we devise an anisotropic control variable, μ, for patch pair pruning: If some DoF is detected to be unstable and enforcing p_k and q_k to be coplanar can constrain it, we set $\mu(\pi_k)$ to be large. The alignment stability is estimated by analyzing the eigenvalues of the 6-DoF alignment error covariance matrix (gradient of the point-to-plane distances w.r.t. the six DoFs in $\mathbf{R}$ and $\mathbf{t}$) as in [4](See details in Section 6.3). Since the stability changes during the optimization, μ should be updated dynamically, and we describe an optimization scheme with dynamically updated μ below.

Algorithm 1: Coplanarity-based Registration

Input : RGB-D frames $\mathcal{F}$ and co-planar patch pairs $\Pi = \cup_{(i,j)\in\mathcal{P}} \Pi_{ij}$; $\gamma_t = 0.5$m.

Output: Frame poses $T = \{(\mathbf{R}_i, \mathbf{t}_i)\}$.

1 $\mathbf{R}_i \leftarrow \mathbf{I}$, $\mathbf{t}_i \leftarrow \mathbf{0}$; // Initialize transformations
2 $\mu_i^d \leftarrow 0.1$m ; // Initialize control variables
3 **repeat**
4 **while** *not converged* **do**
5 Fix s, solve Equation (1) for T ;
6 Fix T, solve Equation (1) for s ;
7 $\{\gamma_i^d\} \leftarrow$ `EstimateStability` (Π, s) ;
8 **foreach** $i \in \mathcal{F}$ **do** // for each frame
9 **foreach** $d \in \{\mathrm{X}, \mathrm{Y}, \mathrm{Z}\}$ **do** // for each DoF
10 **if** $\gamma_i^d > \gamma_t$ **then**
11 $\mu_i^d = \mu_i^d * 0.5$;
12 $\gamma_{\max} \leftarrow \max_{i,d}\{\gamma_i^d\}$;
13 **until** $\gamma_{\max} < \gamma_t$ or *max. # of iterations reached*;
14 **return** T ;

6.2 Optimization

The optimization process is given in Algorithm 1. The core part is solving Equation (1) via alternating optimization of transformations and selection variables (the inner loop in Line 4~6). The iterative process converges when the relative value change of each unknown is less than 1×10^{-6}, which usually takes less than 20 iterations.

A key step of the optimization is stability analysis and stability-based anisotropic pair pruning (Line 7~12). Since our coplanarity-based alignment is inherently orientation-based, it suffices to inspect the stability of the three translational DoFs. Given a frame i, we estimate its translational stability values, denoted by γ_i^d (d is one of the labels of X, Y, and Z-axis), based on the alignment of all frame pairs involving i (see Section 6.3 for details). One can check the stability of frame i along DoF d by examining whether the stability value γ_i^d is greater than a threshold γ_t.

Stability-based anisotropic pair pruning is achieved by dynamically setting the pruning parameter for a patch pair, $\mu(\pi)$ in the coplanarity regularization term (Equation (4) of the main paper). To this end, we set for each frame and each DoF an independent pruning parameter: μ_i^d ($i \in \mathcal{F}$ and $d = \mathrm{X}, \mathrm{Y}, \mathrm{Z}$). They are uniformly set to a relatively large initial value (0.1m), and are decreased in each outer loop to gradually allow more pairs to be pruned. For some μ_i^d, however, if its corresponding stability value γ_i^d is lower than γ_t, it stops decreasing to avoid unstableness. At any given time, the pruning parameter $\mu(\pi)$, with $\pi = (p, q)$, is set to:

$$\mu(\pi) = \min\{\mu_i^{d(p)}, \mu_j^{d(q)}\},$$

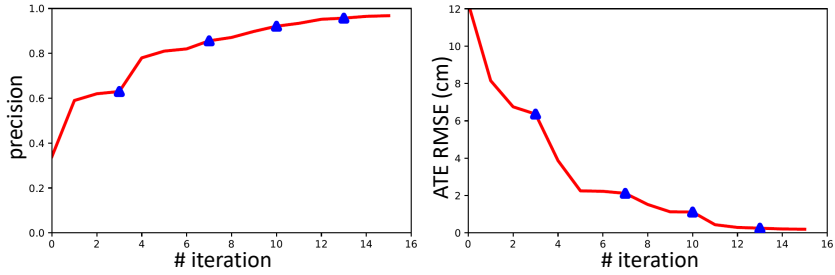

Fig. 10: The percentage of correct co-planar patch pairs increases and trajectory error (see the measure in Section 4 of the paper) decreases, as the iterative optimization proceeding. The blue marks indicate the outer loop.

where $d(p)$ is the DoF closest to the normal of patch p. The whole process terminates when the stability of all DoFs becomes less than γ_t.

To demonstrate the capability of our optimization to prune incorrect patch pairs, we plot in Figure 10 the ratio of correct coplanarity matches at each iteration step for a ground-truth set. We treat a pair π as being kept if its selection variable $s(\pi) > 0.5$ and discarded otherwise. With more and more incorrect pairs pruned, the ratio increases while the registration error (measured by absolute camera trajectory error (ATE); see Section 4 of the paper) decreases.

6.3 Stability Analysis

The stability analysis of coplanar alignment is inspired by the work of Gelfand et al. [4] on geometrically stable sampling for point-to-plane ICP. Consider the point-to-plane alignment problem found in the data term of our coplanarity-based registration (see Equation (3) in the main paper). Let us assume we have a collection of points $\mathbf{v}_p \in \mathcal{V}_p$ sampled from patch p, and a plane $\phi_q = (\mathbf{p}_q, \mathbf{n}_q)$ defined by patch q. We want to determine the optimal rotation and translation to be applied to the point set $\mathcal{V}_p$, to bring them into coplanar alignment with the plane ϕ_q. In our formulation, source and target patches (p and q) are also exchanged to compute alignment error bilaterally (see Line 436 in paper). Below we use only patch p as the source for simplicity of presentation.

We want to minimize the alignment error

$$\mathcal{E} = \sum_{\mathbf{v}_p \in \mathcal{V}_p} \left[(\mathbf{R}\mathbf{v}_p + \mathbf{t} - \mathbf{p}_q) \cdot \mathbf{n}_q \right]^2, \tag{3}$$

with respect to the rotation $\mathbf{R}$ and translation $\mathbf{t}$.

The rotation is nonlinear, but can be linearized by assuming that incremental rotations will be small:

$$\mathbf{R} \approx \begin{pmatrix} 1 & -r_z & r_y \\ r_z & 1 & -r_x \\ -r_y & r_x & 1 \end{pmatrix}, \tag{4}$$

for rotations r_x, r_y, and r_z around the X, Y, and Z axes, respectively. This is equivalent to treating the transformation of $\mathbf{v}_p \in \mathcal{V}_p$ as a displacement by a

vector $[\mathbf{r} \times \mathbf{v}_p + \mathbf{t}]$, where $\mathbf{r} = (r_x, r_y, r_z)$. Substituting this into Equation (3), we therefore aim to find a 6-vector $[\mathbf{r}^T, \mathbf{t}^T]$ that minimizes:

$$\mathcal{E} = \sum_{\mathbf{v}_p \in \mathcal{V}_p} (\mathbf{v}_p - \mathbf{p}_q) \cdot \mathbf{n}_q + \mathbf{r} \cdot (\mathbf{v}_p \times \mathbf{n}_q) + \mathbf{t} \cdot \mathbf{n}_q. \tag{5}$$

We solve for the aligning transformation by taking partial derivatives of Equation (5) with respect to the transformation parameters in $\mathbf{r}$ and $\mathbf{t}$. This results in a linear system $C\mathbf{x} = \mathbf{b}$ where $\mathbf{x} = [\mathbf{r}^T, \mathbf{t}^T]$ and $\mathbf{b}$ is the residual vector. C is a 6×6 "covariance matrix" of the rotational and translational components, accumulated from the sample points:

$$C = \begin{bmatrix} \mathbf{v}_p^1 \times \mathbf{n}_q & \cdots & \mathbf{v}_p^k \times \mathbf{n}_q \\ \mathbf{n}_q & \cdots & \mathbf{n}_q \end{bmatrix} \begin{bmatrix} (\mathbf{v}_p^1 \times \mathbf{n}_q)^T & \mathbf{n}_q \\ \vdots & \vdots \\ (\mathbf{v}_p^k \times \mathbf{n}_q)^T & \mathbf{n}_q \end{bmatrix}.$$

This covariance matrix encodes the increase in the alignment error due to the movement of the transformation parameters from their optimum:

$$\Delta\mathcal{E} = 2 \left[\Delta\mathbf{r}^T \ \Delta\mathbf{t}^T \right] C \begin{bmatrix} \Delta\mathbf{r} \\ \Delta\mathbf{t} \end{bmatrix}. \tag{6}$$

The larger this increase, the greater the stability of the alignment, since the error landscape will have a deep, well-defined minimum. On the other hand, if there are incremental transformations that cause only a small increase in alignment error, it means the alignment is relatively unstable along that degree of freedom. The analysis of stability can thus be conducted by finding the eigenvalues of matrix C. Any small eigenvalues indicate a low-confidence alignment. In our paper, we analyze translational stabilities based on the eigenvalues corresponding to the three translations, γ^d ($d = \mathrm{X, Y, Z}$).

References

1. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR. (2017)
2. Xiao, J., Owens, A., Torralba, A.: Sun3d: A database of big spaces reconstructed using sfm and object labels. In: Proc. ICCV, IEEE (2013) 1625–1632
3. Halber, M., Funkhouser, T.: Fine-to-coarse global registration of rgb-d scans. arXiv preprint arXiv:1607.08539 (2016)
4. Gelfand, N., Ikemoto, L., Rusinkiewicz, S., Levoy, M.: Geometrically stable sampling for the ICP algorithm. In: Proceedings of the International Conference on 3-D Digital Imaging and Modeling (3DIM). (2003) 260–267