Geometric Correspondence Consistency in RGB-D Relative Pose Estimation

Sourav Kumar Chiang-Heng Chien Benjamin Kimia
School of Engineering, Brown University

Abstract

Relative pose estimation for RGB-D cameras is crucial in a number of applications. A typical approach relies on RANSAC to find a triplet pair of 3D point correspondences from which relative pose can be derived. A key aspect to this work ensures the geometric consistency of the triplet, i.e.*, pairwise distances between 3D points are preserved between the two views. Observe, however, that depth values are typically an order of magnitude less precise than feature locations, leading to large distance thresholds and admission of numerous false positives. This paper proposes that the constraint of 3D distance can be cast as a 2D constraint which we refer to as the* Geometric Correspondence Constraint (GCC). *This constraint states that given one pair of correspondences, the two images are partitioned into a family nested of curves such that corresponding points must lie on corresponding curves. This can act as a filter in the RANSAC process with significant savings in computation and with increased robustness and accuracy as demonstrated in experiments using TUM, ICL-NUIM, and RGBD Scene v2 datasets.*

1. Introduction

Relative pose estimation from image pairs is a fundamental and ubiquitous problem for many computer vision tasks, *e.g.* visual odometry [9, 13, 47, 48, 52], SLAM [14, 25, 39, 41, 55], 3D scene reconstruction [23, 44, 49] and scene completion [38, 43, 51], *etc*. A robust feature-based estimation process typically follows a three-step paradigm [11], namely, (*i*) detect and extract features, *e.g.*, SIFT [24] or SuperPoint [8]; (*ii*) measure pairwise feature similarity and form a rank-ordered list of potential matches; (*iii*) apply RANSAC by selecting a certain number of matches from the top M rank-ordered list that is large enough to support the formation of hypotheses but small enough to have a small rate of outliers, *e.g.*, M = 150 [12], or a ratio of the number of matches such as 0.2 [26] and 1 in [1, 33] (taking all matches). The selected matches are used to calculate a camera pose as a competing hypothesis, and iterate N loops to achieve a certain level of success p. The output is a hypothesis approximately consistent with inliers which is a comparably large subset of all the matches.

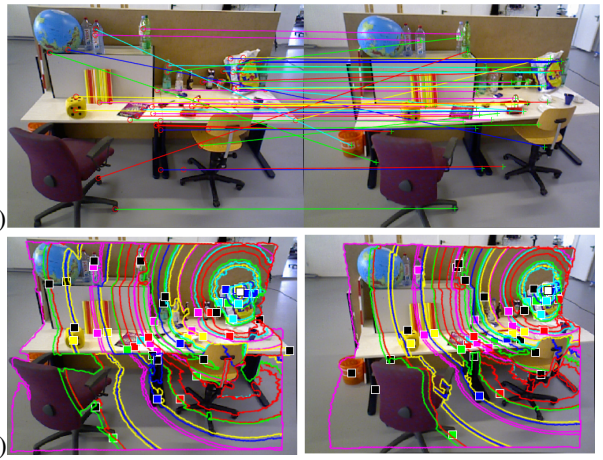


Figure 1. (a) 50 potential matches selected from a rank-ordered list of correspondences between a pair of RGB-D images. (b) A pair of correspondence which is manually determined to be veridical is selected (white square tokens). Each remaining correspondence is probed as to whether the pair falls on corresponding curves using the proposed geometric correspondence consistency constraint. Those potential matches that fail this test are shown in black tokens and excluded as nonviable correspondences.

While this RANSAC approach to feature-based relative pose estimation has had tremendous success in visual odometry (VO), SLAM, and structure-from-motion (SfM) pipelines, there are also instances of failure, especially *(i)* when image pairs experience partial overlap [36], *(ii)* images are blurred due to drastic camera motion [3], and *(iii))* when there is repetitive textures [15]. However, existing methods typically set a maximal RANSAC iterations as efficiency prioritizes over accuracy, *e.g.*, N_{max} = 300 [28], 320 [26], 1000 [1], 8000 [33], or 10000 [30]. Thus, any method that can further boost up efficiencies such that a higher number of RANSAC iterations can be applied will lead to more successful systems.

Relative pose estimation for RGB-D cameras also falls under this RANSAC based paradigm. RGB-D images can be both viewed as images with an augmented depth map or as surfaces in 3D as represented, for example, by an unorganized cloud of 3D points or a mesh. The classic (Horn's) method [17] uses three pairs of correspondences to solve for

a relative pose. A RANSAC approach is used in presence of outliers. However, in contrast to RGB relative pose estimation where the five correspondences can be selected independently, the three correspondences are constrained under rigid transformation to preserve pairwise distances in the RGB-D case. This constraint can be used to sift out geometrically inconsistent triplets and significantly reduce non-veridical correspondences.

This paper argues that the enforcement of 3D geometric consistency maintains a large number of false positives because of a large discrepancy in depth accuracy and image feature accuracy. This motivates a reformulation of the 3D geometric consistency as a 2D geometric consistency constraint (GCC). Specifically, this paper shows that preservation of distances between pairs of features is equivalent to feature lying on corresponding curves in a partitioned image space. This reduces false positives as demonstrated by experiments. In addition, the paper also demonstrates the practical advantages of using the GCC as a filter in a RANSAC scheme for improved efficiency and accuracy.

2. Related Work

Approaches of finding relative pose from a pair of RGB-D images can be briefly categorized into three families:

Classic Method (Horn's Method): The classic approach [17, 34] typically begins by extracting *visual descriptors* [24] from 2D images or *geometric descriptors* [29] from RGB-D scans, constructs 3D-3D correspondences based on descriptor similarity, and finds an analytic solution of rigid transformation that optimally aligns the correspondences in a least squares sense. The relative pose can be found by using all correspondences [2, 5], or by running under a RANSAC scheme for robust estimation where a minimal of 3D triplet correspondence forms one RANSAC hypothesis [31, 37, 46]. Typically, geometric constraints, *e.g.*, pairwise 3D point distances should be preserved under rigid transformation [17, 37], are used to rule out non-veridical correspondences before solving for a relative pose.

Iterative Reweighted Least-Square (IRLS) Method: The least-square structure from the classic method can be cast as an energy function and solved iteratively by a gradient-based numerical optimizer, or, it can be framed as a weighted Procrustes problem [10, 40] and solved by a weighted variant of Kabsch's algorithm [6]. This is particularly popular in many learning based methods [10, 38, 40, 45, 46, 54] as part of its differentiable alignment module. The confidence of each correspondence is adaptively weighted based on descriptor similarity [4, 38, 46], geometric constraints from pairwise distances or surface normal angles [20, 43], *etc*.

Iterative Closest Point (ICP): Given an initial pose, correspondences can be constructed by first projecting points arising from the first image to the second image using an initial pose, and find the nearest point observed on the second image. The relative pose is then solved by iteratively minimizing an energy function based on photometric [7] and geometric residuals [27, 48] of correspondences. This iterative closest point method is especially popular in many real-time RGB-D visual odometry or SLAM pipelines [14, 22, 28, 55] as the frame baselines are typically small so that an initial pose sources from some camera motion model.

3. Notation and Formulation

Consider first the process of estimating the relative pose of two RGB cameras with unknown relative pose $(\mathcal{R}, \mathcal{T})$, where $\mathcal{R}$ is the rotation matrix and $\mathcal{T}$ is the translation vector. Consider an RGB image point $\gamma_i = (\xi_i, \eta_i, 1)^T$ with depth ρ_i in the image of camera one that is in correspondence with an RGB-D point $\overline{\gamma}_i = (\overline{\xi}_i, \overline{\eta}_i, 1)^T$ with depth $\overline{\rho}_i$ in the image of camera two. Let $\Gamma_i = \rho_i \gamma_i$ and $\overline{\Gamma}_i = \overline{\rho}_i \overline{\gamma}_i$ be the corresponding 3D points in each camera, respectively, so that $\overline{\Gamma}_i = \mathcal{R}\Gamma_i + \mathcal{T}$, or

$$\overline{\rho}_i \overline{\gamma}_i = \rho_i \mathcal{R} \gamma_i + \mathcal{T} \quad i = 1, 2, \ldots, 5. \tag{1}$$

Thus, each correspondence $(\gamma_i, \overline{\gamma}_i)$ gives three equations. In the case of RGB images the depths $(\rho_i, \overline{\rho}_i)$ are unknown, so that two of these equations are used to eliminate the unknown depths, leaving only one equation to constrain $(\mathcal{R}, \mathcal{T})$, the well-known epipolar constraint, $\overline{\gamma}_i^T [\mathcal{T}]_\times \mathcal{R} \gamma_i = 0$. Due to metric ambiguity, namely, the simultaneous scaling of $\mathcal{T}$ and depths $(\rho_i, \overline{\rho}_i)$ which leaves the equations intact, there are only five unknowns in $(\mathcal{R}, \mathcal{T})$, thus requiring five correspondences to solve for the pose. What is important to note is that in the general case, the knowledge of one correspondence $(\gamma_i, \overline{\gamma}_i)$ *does not constrain* the remaining correspondences $\left(\gamma_j, \overline{\gamma}_j\right)$ in any form.

In the process of estimating the relative pose of RGB-D, however, the depths $(\rho_i, \overline{\rho}_i)$ are known, so that the Equation $\overline{\Gamma}_i = \mathcal{R}\Gamma_i + \mathcal{T}$ provides three equations for each correspondence. Observe that since depths are known, there is no longer any metric ambiguity so that the pose unknowns $(\mathcal{R}, \mathcal{T})$ involve six unknowns. It would then appear that only two pairs of correspondences which give six equations should suffice to solve for relative pose. However, it is well-known that three pairs of correspondences are required. Thus, the resulting nine equations in six unknowns must be constrained. Indeed, the key observation is that once a first pair of correspondences $((\Gamma_i = (\gamma_i, \rho_i), \overline{\Gamma}_i = (\overline{\gamma}_i, \overline{\rho}_i)))$ is fixed, the second pair of correspondences $(\Gamma_j = (\gamma_j, \rho_j), \overline{\Gamma}_j = (\overline{\gamma}_j, \overline{\rho}_j)))$ must be consistent with it. Specifically, observe that

$$\begin{cases} \overline{\Gamma}_i = \mathcal{R}\Gamma_i + \mathcal{T} \\ \overline{\Gamma}_j = \mathcal{R}\Gamma_j + \mathcal{T} \end{cases} \Rightarrow \overline{\Gamma}_i - \overline{\Gamma}_j = \mathcal{R}(\Gamma_i - \Gamma_j), \tag{2}$$

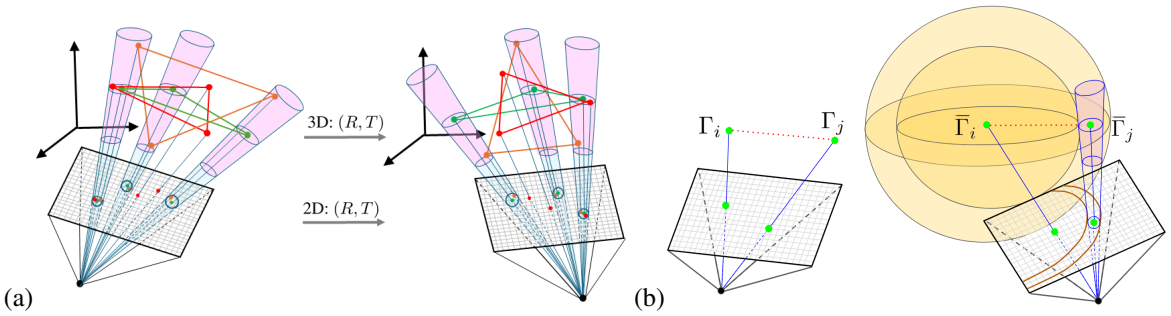


Figure 2. (a) Geometric constraint in pairwise triplet 3D point distances. Note how the constraint (orange) are largely affected by the depth error depicted by the truncated cones (pink), compared to the error-less case (green). Large perturbations of feature locations (shown in red) do not necessarily endow inconsistency of triplet 3D point distance constraint (red), leading to a false positive when using the 3D constraint. (b) In practice, the threshold used for the 3D constraint (the shell of the sphere centered at $\overline{\Gamma}_i$) must be generous to include the perturbations of $\overline{\Gamma}_j$, whereas the perturbation on the 2D image space (a curve band) is small.

so that

$$\|\Gamma_i - \Gamma_j\|^2 = \left\|\overline{\Gamma}_i - \overline{\Gamma}_j\right\|^2. \tag{3}$$

The simple geometric interpretation is that lengths between corresponding features must be preserved under a rigid transformation. This constraint reduces the number of equations per correspondence from three equations to two equations. Similarly, a third pair of correspondences $(\Gamma_k = (\gamma_k, \rho_k), \overline{\Gamma}_k = (\overline{\gamma}_k, \overline{\rho}_k))$ must be consistent with both the first and the second correspondences. These three constraints reduce the nine equations to six equations in six unknowns. The important observation is that contrary to RGB relative pose estimation where the selection of five correspondences is an independent process, the selection of three correspondences for RGB-D relative pose estimation is constrained: pairwise distances among correspondences must be the same in the two cameras.

The 3D registration algorithms have indeed taken advantages of this constraint. Specifically, SAC-COT [42] samples triplet correspondences from two clouds of 3D points and rank-orders them based on the 3D pairwise distance constraint, forming a guided sampling scheme under a RANSAC loop; FGR [53] and similarly [20] filter out spurious triplet 3D correspondences by a tuple test on point-to-point distance before aligning point cloud fragments. Such geometric consistency is also used to score triplet correspondences in [43], effectively guiding a network to learn relative pose from wide-baseline RGB-D images.

4. Reformatting the 3D Constraint as a 2D Constraint

It is well-known that the image feature locations are known fairly precisely, while the depth is relatively less precise by an order of magnitude, as sketched in Figure 2(a). The feature localization error is represented by a disc of allowable perturbation on the image plane while depth localization is an interval along the ray from the camera center to the image feature resulting in a truncated cone whole length is typically an order of magnitude longer than the radius of the base. The constraint that the distance between 3D points must be preserved across views, is

$$\left|\|\Gamma_i - \Gamma_j\| - \left\|\overline{\Gamma}_i - \overline{\Gamma}_j\right\|\right| < \tau_{3D}. \tag{4}$$

The threshold τ_{3D} must represent the allowable perturbations of the 3D points within the truncated cone. Consider $(\Gamma_i, \overline{\Gamma}_i)$ as a given pair and Γ_j as a 3D point whose corresponding 3D point $\overline{\Gamma}_j$ needs to be found. Considering only the error $\overline{\Gamma}_j$ experiences under truncated cone, the threshold τ_{3D} must be generous enough to include the perturbations of $\overline{\Gamma}_j$, *i.e.*, the truncated cone in Figure 2(b). Such a large threshold, however, permits the inclusion of numerous erroneous correspondences that are not necessarily close in the image space. The goal of this paper is to consider perturbations in the image space, which as will turn out, include only points with a small 2D band, Figure 2(b). This is much more selective in seeking proper correspondences.

This observation motivates a reformulation of the 3D constraint directly in the 2D image domain where perturbation of image features can be directly controlled. The 2D formulation also assumes a given point pair correspondence $(\Gamma_i = (\gamma_i, \rho_i), \overline{\Gamma}_i = (\overline{\gamma}_i, \overline{\rho}_i))$ and probes the consistency of a candidate point correspondence $(\Gamma_j = (\gamma_j, \rho_j), \overline{\Gamma}_j = (\overline{\gamma}_j, \overline{\rho}_j))$ by consideration in the image domain, *i.e.*, Equation 3 is expanded as $\|\rho_i\gamma_i - \rho_j\gamma_j\|^2 = \left\|\overline{\rho}_i\overline{\gamma}_i - \overline{\rho}_j\overline{\gamma}_j\right\|^2$. In this approach, the first correspondence is given so that (γ_i, ρ_i) and $(\overline{\gamma}_i, \overline{\rho}_i)$ are known. Similarly, in probing the pairing $((\gamma_j, \rho_j), (\overline{\gamma}_j, \overline{\rho}_j))$, the first point (γ_j, ρ_j) can be assumed to be known, while searching $(\overline{\gamma}_j, \overline{\rho}_j)$ which are consistent with the givens. Finally, consider that the depth map is known so that $\overline{\rho}_j$ can be derived given $\overline{\gamma}_j$. Expanding this equation gives.

$$\begin{aligned}&\overline{\rho}_j^2\overline{\gamma}_j^T\overline{\gamma}_j - 2\overline{\rho}_i\overline{\rho}_j(\overline{\gamma}_i^T\overline{\gamma}_j) + \overline{\gamma}_i^T\overline{\gamma}_i\overline{\rho}_i^2 \\ =&\rho_j^2\gamma_j^T\gamma_j - 2\rho_i\rho_j(\gamma_i^T\gamma_j) + \rho_i^2(\gamma_i^T\gamma_i) = \|\Gamma_i - \Gamma_j\|^2 = r^2,\end{aligned} \tag{5}$$

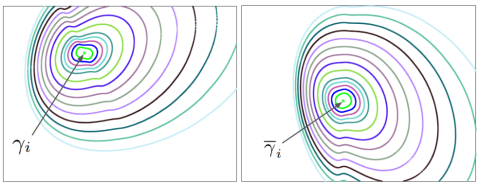


Figure 3. A veridical correspondence $(\gamma_i, \overline{\gamma}_i)$ partitions the space of correspondences $(\gamma_j, \overline{\gamma}_j)$ into a nested set of curves (identified by a common color) so that if γ_j falls on a curve in image one, $\overline{\gamma}_j$ must fall on the corresponding curve in image two, and vice versa.

where the only unknowns are $(\overline{\xi}_j, \overline{\eta}_j)$, the coordinate of $\overline{\gamma}_j = (\overline{\xi}_j, \overline{\eta}_j, 1)^T$. This single equation in two unknowns gives the locus of $\overline{\gamma}_j$ for any γ_j, namely, a curve!

Several observations are in order. First, observe that the symmetry in Equation 5 implies that if the locus of correspondences $\overline{\gamma}_j$ for a given γ_j is a curve, then given any point $\overline{\gamma}_j$ on that curve, the space of γ_j consistent with $\overline{\gamma}_j$ is also a curve. Furthermore, any pair of points each selected from the corresponding pair of points satisfy Equation 5. This is in analogy with a pair of corresponding epipolar lines when any point on one line matches any point on the second line.

Second, observe that when the correspondence $(\gamma_i, \overline{\gamma}_i)$ is given, a perturbation of γ_j changes the right side of Equation 5 denoted by r^2. The resulting pair of curves that result from this perturbations cannot intersect the previous pair of curves. Thus, further changes in γ_j produces a family of non-intersecting, nested curves indexed by r as the example in Figure 3 illustrates. Since γ_j and conversely $\overline{\gamma}_j$ are free to be anywhere in the image, the constraint of consistency with $(\gamma_i, \overline{\gamma}_i)$ represented by Equation 5 is referred to as *geometric correspondence consistency (GCC)*, *partitioning two images into a collection of nested pairs of corresponding curves*, Figure 1. The GCC states that a pair of points $(\gamma_j, \overline{\gamma}_j)$ must live on corresponding pairs of curves.

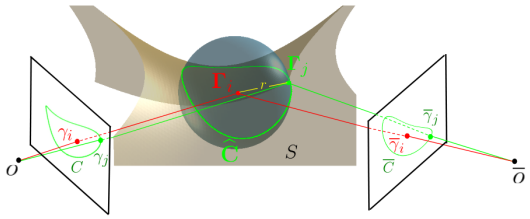


Figure 4. A scene surface S viewed by two cameras. Assuming the correspondence $(\gamma_i, \overline{\gamma}_i)$ both come from a 3D point Γ_i, a sphere of radius r centered at Γ_i (shown in red), and S intersect at a curve $\widehat{\mathbf{C}}$ (green). The curve $\widehat{\mathbf{C}}$ projects to 2D curves C and $\overline{C}$ in image i and image j, respectively. This demonstrates that any feature γ_j lying on curve C must have its correspondence on curve $\overline{C}$.

A geometrical interpretation of this constraint is illuminating. Consider a scene surface $\mathcal{S}$ which is viewed by two RGB-D cameras, Figure 4 whose relative pose is unknown. Assume that a correspondence, say $((\gamma_i, \rho_i), (\overline{\gamma}_i, \overline{\rho_i}))$ arising from a common point $\Gamma_i = \rho_i \gamma_i$ expressed in camera one, and expressed as $\overline{\Gamma}_i = \overline{\rho}_i \overline{\gamma}_i$ in camera two, is known, as described earlier. Then, for any point in camera one, γ_j with depth ρ_j, the 3D point $\Gamma_j = \rho_j \gamma_j$ is known in camera one, while the point corresponding to it, $\overline{\gamma}_j$, is unknown. Equation 3 describes the locus of $\overline{\Gamma}_j$ as a sphere centered at $\overline{\Gamma}_i$ with radius $r = |\Gamma_j - \Gamma_i|$. Since $\overline{\Gamma}_j$ also lies on the surface $\mathcal{S}$, the locus of $\overline{\Gamma}_j$ is the intersection of the sphere and scene surface, as shown by the green curve $\widehat{\mathbf{C}}$ in Figure 4, which when projected on the second camera, it traces out 2D curves. Similarly, this curve when projected on the first image gives curve C. Thus, any point on $\mathcal{C}$ can only have its correspondences on $\overline{\mathcal{C}}$, and conversely, any point on $\overline{C}$ can only have its correspondences on $\mathcal{C}$.

In practice, given $(\gamma_j, \overline{\gamma}_j)$, the distance of $\overline{\gamma}_j$ from $\overline{C}$ must be below a threshold,

$$d\left(\overline{\gamma}_j, \overline{C}\left(\gamma_j | (\gamma_i, \overline{\gamma}_i)\right)\right) < \tau_{2D}, \tag{6}$$

where τ_{2D} is twice the feature location error Δ, Figure 5. The critical point is that the false positives allowed by this constraint in the image plane are significantly fewer than those in 3D, Figure 2. This observation is experimentally validated in Figure 7.

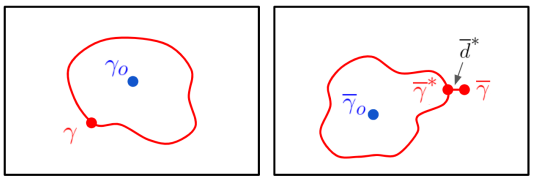


Figure 5. The geometric correspondence consistency constrains a correspondence (γ, ρ) and $(\overline{\gamma}, \overline{\rho})$ to lie on the corresponding curves with respect to a reference point correspondence (γ_0, ρ_0) and $(\overline{\gamma}_0, \overline{\rho}_0)$. Due to noise in feature location and depth measurement, the observed correspondence $\overline{\gamma}_i$ is a perturbation of the true corresponding point $\overline{\gamma}^*$ by $\overline{d}^*$.

More specifically, the computation of the distance from $\overline{\gamma}_j$ from the curve $\overline{C}\left(\gamma_j | (\gamma_i, \overline{\gamma}_i)\right)$ requires an exploration of this curve. However, a first-order approximation of this distance can be derived by first defining a radial map:

Definition 4.1. The radial map of an RGB-D image with respect to a reference point (γ_0, ρ_0) is defined as

$$r(\xi, \eta) = \|\rho(\xi, \eta)\gamma(\xi, \eta) - \rho_0 \gamma_0\|. \tag{7}$$

The key use of a radial map is that given a candidate pair of correspondences $\left((\gamma_j, \rho_j), (\overline{\gamma}_j, \overline{\rho}_j)\right)$, γ_j and $\overline{\gamma}_j$ must share the same radial value. Now, consider a pair of correspondences $(\gamma, \overline{\gamma})$. Let $\hat{\gamma}$ denote an arbitrary point of the curve $\overline{C}$ corresponding to γ. Define the closest point to $\overline{\gamma}$ on $\overline{C}$ as $\overline{\gamma}^*$ and the distance of $\overline{\gamma}$ from the curve as $\overline{d}^*$, *i.e.*,

$$\overline{\gamma}^* = \underset{\hat{\gamma}, \overline{r}(\hat{\gamma}) = r(\gamma)}{\arg\min}\ d(\overline{\gamma}, \hat{\gamma}), \quad \overline{d}^* = \underset{\hat{\gamma}, \overline{r}(\hat{\gamma}) = r(\gamma)}{\min}\ d(\overline{\gamma}, \hat{\gamma}). \tag{8}$$

Then, $\overline{d}^*$ can be estimated as:

Proposition 1. Let r and $\overline{r}$ be the radial maps of the first and second images, with respect to (γ_0, ρ_0) and $(\overline{\gamma}_0, \overline{\rho}_0)$, respectively. Given a putative correspondence, $(\gamma, \overline{\gamma})$, the first-order approximation of $\overline{d}^*$ is

$$\overline{d}^* = \frac{\overline{r}(\overline{\xi},\overline{\eta})|r(\xi,\eta) - \overline{r}(\overline{\xi},\overline{\eta})|}{\left\| \left(\overline{\rho}||\overline{\gamma}||^2 - \overline{\rho}_0\overline{\gamma}_0^T\overline{\gamma}\right)\nabla\overline{\rho} + \overline{\rho}\begin{bmatrix}\overline{\rho}\overline{\xi} - \overline{\rho}_0\overline{\xi}_0 \\ \overline{\rho}\,\overline{\eta} - \overline{\rho}_0\overline{\eta}_0\end{bmatrix}\right\|}, \quad (9)$$

where $\overline{\rho}(\overline{\xi},\overline{\eta})$ is the depth at $\overline{\gamma}(\overline{\xi},\overline{\eta})$.

The proof is given in the supplementary materials. The above proposition allows for an examination of each candidate correspondence $(\gamma, \overline{\gamma})$: if $\overline{d}^* < \tau_{2D} = 2\Delta$, then the correspondence is consistent with $(\gamma_0, \overline{\gamma}_0)$; otherwise, it is discarded.

GCC is immune to depth errors: The question arises as the extent by which the expected large depth errors affect the performance of GCC. The distance $\overline{d}^*$ in Equation 9 has dependency on depth values $(\rho_0, \overline{\rho}_0, \rho, \overline{\rho})$ which are notoriously noisy. These errors, in a typical RGBD sensor, *e.g.*, the Microsoft Kinect which is used in the TUM-RGBD and RGBD Scene v2 datasets, are in the range of 1-3 cm for depths of 2-5 m [18]. Similarly, the depth error distribution of SIFT features for the synthetically perturbed depths modeled in the ICL-NUIM dataset, is in the range of 1-4 cm, Figure 6(a). Differentiating $\overline{d}^*$ with respect to each of the variables $(\rho_0, \overline{\rho}_0, \rho, \overline{\rho})$ measures how slight changes in depth affect $\overline{d}^*$, as shown in Figure 6(b) which demonstrates the effect of realistic levels of depth errors on $\overline{d}^*$ is in the range of subpixel for $(\rho, \overline{\rho})$ and in the range below 2 pixels for $(\rho_0, \overline{\rho}_0)$, both well below the threshold of 3 pixels used throughout this paper.

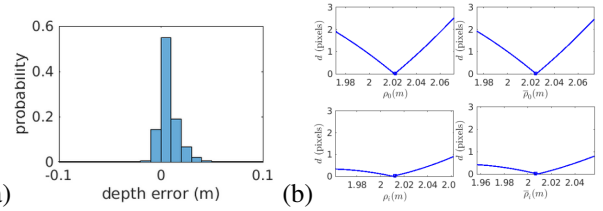


Figure 6. (a) The synthetically modeled depth error distribution of SIFT features in the ICL-NUIM dataset. Bin size is 0.01 (m). (b) The effect of changes in ρ_0, $\overline{\rho}_0$, ρ, and $\overline{\rho}$ on $\overline{d}^*$ is well below threshold.

Robustness of GCC relative to depth noise. Gaussian is added to feature locations and the depths in this experiment. Following realistic data models, the variance of depth noise is a factor higher than the image localization noise. Figure 7 compares the translation error for the 3D filtering and our 2D GCC filtering. It is clear that 2D GCC filter pose estimation is more accurate.

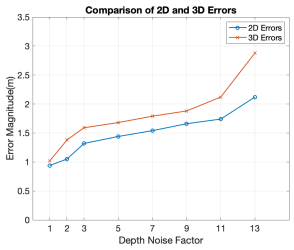


Figure 7. Relative translation errors for varying amount of depth noise, 2D vs 3D GCC

5. GCC Filtered RANSAC

The GCC constraint provides a filter where a triplet of correspondences can be checked for consistency, thus aborting the expensive validation phase for inconsistent triplets in a RANSAC scheme. Observe Table 1 that the overall cost of RANSAC is dominated by the second validation stage which dwarfs the first hypothesis formulation stage by a factor of 32μs compared to 1μs. The GCC filter cost is $\sim 0.73\mu$s so that over 1% of the hypotheses need to be filtered out for it to be cost effective. This of course depends on the outlier ratio.

The GCC filter increases the hypothesis formulation cost but it avoids the costly second stage for many hypotheses. Formally, in order to achieve a success rate p, the number of RANSAC iterations N is required to be higher than

$$N \simeq \frac{\log(1-p)}{\log(1-(1-e)^s)}, \quad (10)$$

where e is the proportion of outliers, and s is the number of samples required to form a hypothesis (s = 3 in our case). For example, with e = 70% and p = 99%, the required number of iterations is 169. This number changes rapidly with outlier ratio so that with $e = 80\%$, $N = 574$.

The main effect of the GCC filter is to remove inconsistent hypotheses, thus effectively reducing the outlier ratio, Figure 8(a). This in turn reduces the required number of RANSAC iterations by a factor μ,

$$\mu = \frac{N}{\overline{N}} = \frac{[\log(1-(1-\overline{e})^s)]}{[\log(1-(1-e)^s)]}. \quad (11)$$

It is interesting that the ratio μ is independent of the probability of success p and is exponentially increasing with

Steps	Classic (μs)	GCC (μs)
Hypothesis formulation cost	0.960	1.691
Absolute Pose Estimation per hypothesis	1.203	1.203
Find Number of inliers per hypothesis	31.061	31.061
Hypothesis support measurement cost	32.264	32.264
Average cost of evaluating a hypothesis	33.224	33.955

Table 1. The computation cost of the classic RANSAC scheme is dominated by the second stage of hypothesis support measurement as compared to the first hypothesis formulation stage.

outlier ratio e, Figure 8(b). Table 2 summarizes the time savings as a result of this filter, where the hypotheses are selected from the top $M = 250$ of the rank-ordered list. Observe that the GCC filter significantly reduces the computational cost of the RANSAC scheme, or equivalently, it increases the success rate under the same time budget.

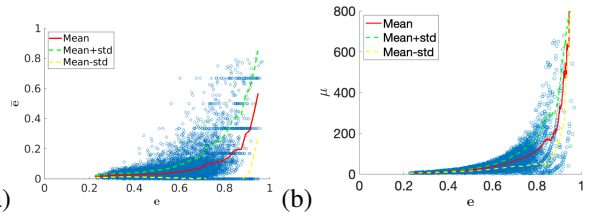


Figure 8. (a) The scatter plot of e and $\bar{e}$, namely, the outlier ratios before and after the GCC filter is applied and (b) the ratio of the number of required iterations before and after applying the GCC filter to TUM-RGBD [35] dataset for success probability of 0.99. Note that the scale is too small to appropriate that at (0.2, 0.3, 0.4, 0.5, 0.6) the value of μ is (5, 7, 12, 21, 37), respectively.

6. Algorithm

The detailed algorithm of the GCC-filtered RANSAC, Algorithm 1, is presented here. First, gradient depths of all image feature correspondences are computed. In each RANSAC iteration, a nested examination of candidate correspondences $(\gamma_1, \overline{\gamma}_1)$ and $(\gamma_2, \overline{\gamma}_2)$ given $(\gamma_0, \overline{\gamma}_0)$ based on the first-order approximation of $\overline{d}^*$, Equation 9, is used. Specifically, two random point pairs $(\gamma_0, \overline{\gamma}_0)$ and $(\gamma_1, \overline{\gamma}_1)$ are picked from the top rank-ordered list of correspondences, from which the higher ranked correspondence is $(\gamma_0, \overline{\gamma}_0)$. Secondly, since one image does not necessarily prevail another, a bidirectional examination is adopted, *i.e.*, the forward direction distance $d_{f,1}$ of $\overline{\gamma}_1$ from the curve $\overline{C}(\gamma_1|(\gamma_0, \overline{\gamma}_0))$ and the backward direction distance $d_{b,1}$ of γ_1 from the curve $C(\overline{\gamma}_1|(\overline{\gamma}_0, \gamma_0))$ are computed. If both distances $d_{f,1}$ and $d_{b,1}$ are greater than τ_{2D}, $(\gamma_0, \overline{\gamma}_0)$ and $(\gamma_1, \overline{\gamma}_1)$ are discarded and the RANSAC loop is reiterated. Otherwise, a third correspondence $(\gamma_2, \overline{\gamma}_2)$ is randomly picked from the top rank-ordered list of correspondences and examined by the same bidirectional approach, giving distances $d_{f,2}$ and $d_{b,2}$. Thus, only if the four distances $d_{f,1}$, $d_{b,1}$, $d_{f,2}$, and $d_{b,2}$ are below τ_{2D} does a camera pose hypothesis is formulated and validated which follows the standard RANSAC scheme. Code is publicly available in https://github.com/Brown-LEMS/RGBD_Geometric_Correspondence_Consistency.

Algorithm 1: GCC Filtered RANSAC

Input : Feature correspondences and their depths
Output : Relative camera pose
Initialization: $N_{max} = 0$
Compute interpolated gradient depths of all correspondences
for $i = 1$ **to** *MAX_RANSAC_Iterations* **do**
  $(\gamma_0, \overline{\gamma}_0)$ and $(\gamma_1, \overline{\gamma}_1) \leftarrow$ top rank-ordered list
  Compute $r(\xi, \eta)$ and $\overline{r}(\overline{\xi}, \overline{\eta})$
  $d_{f,1} \leftarrow \overline{d}^*$ // forward direction
  $d_{b,1} \leftarrow d^*$ // backward direction
  if $d_{f,1} < \tau_{2D}$ **and** $d_{b,2} < \tau_{2D}$ **then**
    $(\gamma_2, \overline{\gamma}_2) \leftarrow$ top rank-ordered list
    $d_{f,2} \leftarrow \overline{d}^*$ // forward direction
    $d_{b,2} \leftarrow d^*$ // backward direction
    if $d_{f,2} < \tau_{2D}$ **and** $d_{b,2} < \tau_{2D}$ **then**
      // hypothesis formation
      Estimate the camera relative pose (R, T).
      // hypothesis support measurement
      $N \leftarrow \#$. hypothesis supports.
      if $N > N_{max}$ **then**
        $(R^*, T^*) \leftarrow (R, T)$
        $N \leftarrow N_{max}$
      end
    else
      Back to step 4.
    end
  else
    Back to step 4.
  end
end
return (R^*, T^*)

7. Experiments

The development of the GCC filter speedup the RANSAC process significantly, especially for cases with a large number of outliers. The significant speedup can boost the RANSAC efficiency in some applications or it can enable its use when it was previously viewed as prohibitive. Specifically, we demonstrate this on three examples: *(i)* pose esti-

	e = 60-70% 32,318 image pairs		e = 70-80% 22,975 image pairs		e = 80-90% 13,808 image pairs		e = 90-95% 6,235 image pairs		e = 95-99% 4,532 image pairs	
	Classic	**GCC- F**	**Classic**	**GCC- F**	**Classic**	**GCC- F**	**Classic**	**GCC- F**	**Classic**	**GCC- F**
# of RANSAC iterations (99% success rate)	169	44	420	85	3752	533	21375	876	681274	14374
Hypothesis formation cost (ms)	0.16	0.29	0.40	0.71	3.60	6.35	20.52	36.15	654.02	1152.04
Hypothesis support measurement cost (ms)	5.45	1.43	13.55	2.74	121.05	17.19	689.64	28.26	21980.62	463.76
Total Cost (ms)	5.61	1.72	13.95	3.45	124.65	23.54	710.16	64.41	22634.64	1615.80

Table 2. A comparison of the computation cost of the classic RANSAC and the GCC-filtered (GCC-F) RANSAC for 99% success rate over the entire TUM-RGBD dataset, 132,946 image pairs, with successful pose estimation defined as having less than 0.5 degree in rotation and 0.05 meters in translation. Image pairs are generated by each image and one subsequent image at intervals ranging from 1 to 30 frames. The resultant image pairs are then categorized by the observed outlier ratio.

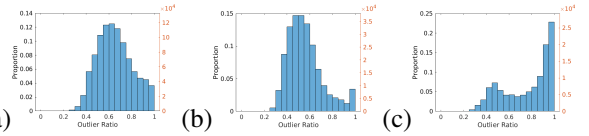


Figure 9. Distribution of outlier ratio $\mathbf{e}$ for image pairs generated from the (a) TUM-RGBD [35], (b) ICL-NUIM [16], and (c) RGBD Scene v2 [19] datasets. Number of image pairs are 132,946, 38,085, and 39,325, respectively. Bin size is 0.05.

mation in wide baseline cameras when the number of features is reduced and the ratio of outliers is increased, *(ii)* in visual odometry, and *(iii)* as a replacement of the motion constant stage in a visual odometry or visual SLAM by doing a full pose estimation instead. Each is discussed after discussing datasets and metrics used in the experiments, and the performances of GCC in 2D against GCC in 3D.

Datasets: For all the experiments, we use three popular datasets, namely, TUM-RGBD [35], ICL-NUIM [16], and RGBD Scenes v2 [19]. First, six sequences are selected from the TUM-RGBD dataset, namely, *fr1_desk* (fr1/desk), *fr1_room* (fr1/room), *fr1_xyz* (fr1/xyz), *fr2_desk* (fr2/desk), *fr3_long_office_household* (fr3/office), and *fr3_structure_texture_near* (fr3/struct) are used. These sequences were chosen to cover a diverse set of conditions: The first three sequences exhibit blurry images and illumination variations; the fourth sequence exhibits a generic textureless scene; and, the last two sequences exhibit mixtures of texture/textureless and planar/non-planar scenes. Second, all eight sequences of the ICL-NUIM dataset are used, exhibiting low contrast and low texture synthetic indoor scenes with artificial depth noise. Finally, all 14 sequences of RGBD Scene v2 dataset are used, exhibiting low illumination, repetitive features, homogeneous indoor scenes with a large portion of the image having no depth values. Image resolutions are identical and relatively small (480×640).

Pairs of images were selected from each dataset by pairing each image with an image that is some interval of frames away, ranging from 1 to 30 time-steps. The generated pairs were then categorized by the outlier ratio into bins. Outlier ratio distributions are shown in Figure 9, for 132,946, 38,085, and 39,325 image pairs generated from the TUM-RGBD, ICL-NUIM, and RGBD Scene v2 datasets, respectively. Observe that all datasets have pairs with a high outlier ratio, but in particular, the high outlier ratio category dominates the RGBD Scene v2 dataset.

Metrics: The relative pose error (RPE) [50] is used to measure the estimation accuracy. RPE measures both rotation and translation drifts of one frame n with respect to another frame $n-\Delta$, where Δ is the number of frames apart. $\Delta = 1$ in our experiments, if otherwise specified.

GCC in 3D versus GCC in 2D for Relative Pose Estimation: Equation 4 shows how GCC in 3D is done in practice. However, since depth uncertainty typically grows with its value, in practice the distance error between pairwise 3D points can be normalized, *i.e.*, the corresponding 3D point pairs $(\Gamma_i, \overline{\Gamma}_i)$ and $(\Gamma_j, \overline{\Gamma}_j)$ are consistent if

$$\frac{2\left|\|\Gamma_i - \Gamma_j\| - \left\|\overline{\Gamma}_i - \overline{\Gamma}_j\right\|\right|}{\left|\|\Gamma_i - \Gamma_j\| + \left\|\overline{\Gamma}_i - \overline{\Gamma}_j\right\|\right|} < \tau_{3D}. \tag{12}$$

We refer Equation 4 and 12 as the unnormalized and normalized constraint for GCC in 3D, respectively, both of which are used to compare their effectiveness with that in 2D in terms of relative pose estimation accuracy. As demonstrated in Table 3 running over the TUM-RGBD dataset, GCC in 2D is superior in both outlier removal (speed) and accuracy, including normalized constraint for GCC in 3D, *e.g.*, there is a 47.5% improvement for translation in wide baseline image pairs.

Baseline (frames)	GCC in 3D (Unnormalized)	GCC in 3D (Normalized)	GCC in 2D (Eq. (6))
14	1.08 / 0.34	1.62 / 0.35	**0.77** / **0.32**
22	1.27 / 0.37	1.71 / **0.36**	**0.81** / 0.38
30	1.31 / **0.40**	1.90 / 0.45	**0.90** / 0.41

Table 3. The relative pose accuracy (translation (cm) / rotation (degree)) comparisons for GCC acting in the 2D and 3D space for wide-baseline image pairs in fr3/office sequence of TUM-RGBD dataset. The thresholds are empirically optimal for each method, *i.e.*, 0.1 (m) and 0.05 for GCC in 3D for unnormalized and normalized cases, and 1 (pixel) for GCC in 2D.

Wide-Baseline Relative Pose Estimation in RGBD Sequences: Table 4 shows the performances of the GCC filter as compared to the classic RANSAC with geometry constraint acting in 3D space on a diverse set of baselines of TUM-RGBD pairs that are 1, 7, 14, 221 and 30 frames apart. The comparison is close roughly under a fixed budget of computational time so that the number of RANSAC iterations is 3000 and 8000 for the traditional and GCC RANSAC, respectively, but even then the GCC was significantly faster. The estimation errors are significantly smaller for GCC, especially in the wide baseline category. More experiments are given in the supplementary material.

Name	Narrow 1f **C / G-F**	7f apart **C / G-F**	14f Med. **C / G-F**	22f apart **C / G-F**	Wide 30f **C / G-F**
fr1/ desk	1.14 / 0.9 0.56 / 0.5	5.85 / 3.01 0.67 / 0.65	7.90 / 4.23 4.46 / 0.79	48.89 / 4.16 40.92 / 0.80	81.41 / 7.02 68.00 / 1.02
fr1/ room	0.77 / 0.69 0.41 / 0.36	17.85 / 4.38 13.55 / 1.18	66.29 / 5.70 26.57 / 2.60	137.54 / 6.3 68.52 / 3.40	157.41 / 8.80 89.10 / 6.10
fr1/ xyz	0.53 / 0.49 0.32 / 0.36	1.30 / 1.00 0.67 / 0.62	1.90 / 1.41 0.94 / 0.82	34.32 / 3.24 34.23 / 1.8	38 / 3.8 40.62 / 2.7
fr2/ desk	2.62 / 0.89 0.72/ 0.49	6.9 / 2.47 6.4 / 2.3	9.81 / 3.56 9.79/ 3.49	34.32 / 4.39 12.9 / 4.22	14.41 / 4.97 13.6 / 5.1
fr3/ struct	1.21 / 0.99 0.81/ 0.60	1.4 / 1.01 0.81 / 0.73	1.53 / 1.02 0.87 / 0.77	1.53 / 1.07 0.90 / 0.8	1.96 / 1.4 1.03/ 0.9
fr3/ office	0.57 /0.45 0.69 / 0.24	0.83 / 0.75 0.51 / 0.3	1.09 / 0.77 0.75 / 0.32	1.30 / 0.81 0.81 / 0.38	1.88 / 0.9 0.87 / 0.41

Table 4. Pose estimation errors ($\text{RPE}_{\text{trans}}$ (cm) in the top row and RPE_{rot} (degree) in the bottom row) for classic RANSAC (**C**) and GCC-filtered RANSAC (**G-F**) across different baselines.

GCC in Visual Odometry: The performance of the GCC-filtered RANSAC is compared to prevailing methods including ACO [21], Edge DVO [7], and PLP-SLAM [32] in Tables 5, 6, and 7 for the three datasets. The number of RANSAC iterations is set to 3000. It is important to note that no refinement or bundle adjustment has been incorporated in the GCC filter results. The GCC filer gives by a wide margin the best pose estimations accuracy in ICL-NUIM and RGBD Scene v2 datasets. In the TUM-RGBD dataset, the accuracy is generally the best but not always and not necessarily by a wide margin. Note that for PLP-SLAM, loop closure and global bundle adjustment are turned off.

	fr1/desk	fr1/room	fr1/xyz	fr2/desk	fr3/struct	fr3/office
ACO	**1.00/0.59**	**0.56/0.39**	0.88/1.12	**0.49**/0.57	1.59/0.83	**0.47**/0.35
Edge DVO	17.32/15.17	×	1.57/5.37	1.34/2.76	1.63/0.98	1.04/0.56
PLP-SLAM	1.07/0.84	1.68/2.57	4.86/1.32	3.43/3.93	2.56/3.66	4.12/1.93
Ours	1.05/**0.59**	0.78/**0.38**	**0.56/0.36**	0.72/0.49	**1.38/0.76**	0.75/**0.32**

Boldfaced: the best. Underlined: the second best. ×: Estimations diverged.

Table 5. $\text{RPE}_{\text{trans}}$(cm) / RPE_{rot}(degree) comparisons on selected sequences of the TUM-RGBD dataset against modern VO pipelines.

	s01	s02	s03	s04	s05	s06	s07	s08	s09	s10	s11	s12	s13	s14
ACO	1.98	2.27	2.48	2.94	3.57	3.82	3.64	3.84	1.74	1.72	2.15	1.91	1.71	2.82
	1.52	1.41	1.08	1.47	1.59	1.55	1.48	2.11	1.16	1.32	1.06	2.07	1.30	1.46
Edge DVO	2.00	2.02	2.01	2.01	2.98	2.97	2.98	2.99	3.01	3.00	3.01	3.00	×	×
	2.22	2.42	2.40	2.26	1.03	0.99	1.00	1.15	1.05	1.08	1.17	1.01		
PLP-SLAM	2.09	2.78	3.69	2.55	2.81	3.11	2.32	3.33	1.82	1.30	2.49	1.45	2.89	3.06
	1.39	1.76	1.36	2.14	2.06	1.98	1.26	1.19	1.25	1.84	3.24	2.98	1.74	2.61
Ours	**0.68**	**0.70**	**0.75**	**0.83**	**0.96**	**1.03**	**1.02**	**1.07**	**0.71**	**0.70**	**0.75**	**0.69**	**0.67**	**0.49**
	0.1	**0.09**	**0.10**	**0.11**	**0.15**	**0.15**	**0.14**	**0.15**	**0.13**	**0.12**	**0.12**	**0.11**	**0.14**	**0.16**

Boldfaced: the best. Underlined: the second best. ×: Estimations diverged.

Table 7. $\text{RPE}_{\text{trans}}$(cm) and RPE_{rot}(degree) comparisons of the RGBD Scene v2 dataset, with $\text{RPE}_{\text{trans}}$ above and RPE_{rot} below in each method.

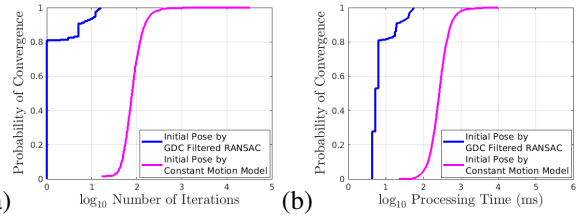
GCC with Refinement for Visual Odometry: In a typical visual odometry, robust camera pose estimation is typically achieved by a two-stage approach: first producing an initial pose by a constant motion assumption [22, 25, 28, 39], and second, refining the initial pose by minimizing some energy function, *e.g.*, reprojection errors [28], 3D point cloud distance [22], color and depth rendering loss [39], *etc*. The constant motion assumption is used because RANSAC pose estimation is too expensive. However, the significant speedup in the GCC filter while maintaining or improving accuracy is enabling a new approach: replace the initial stage with pose estimation. This is expected to improve the initial pose so that the refinement stage iterations to convergence are reduced. In addition, the probability of converging to local minima is reduced. In an experiment to verify this approach, the initial pose estimation stage of the CVO-SLAM [22] is replaced with the GCC filter with the number of GCC iterations set to 100. Note that for this experiment, CVO-SLAM is operating in a tracking mode.

Figure 10 (a) and (b) report the cumulative distribution function (CDF) of the $\log_{10}$ iteration numbers for the pose refinement stage and the $\log_{10}$ processing times in milliseconds of the robust estimation procedure, respectively. Evidently, 80% of the poses from the GCC filtered RANSAC are so accurate that it reaches to the convergence condition in the first iteration of the refinement stage, giving around two orders of magnitude fewer iterations. The curve for the processing time in the case of using constant motion model is consistent with the iteration numbers as the time of producing an initial pose is ignorable. Nevertheless, as the number of refinement iteration grows, the processing time becomes around 1.5 orders of magnitude slower than the GCC filtered RANSAC. Note that the processing time of the GCC filtered RANSAC includes the time for gradient depth computation in Equation 9.

	lr kt0	lr kt1	lr kt2	lr kt3	of kt0	of kt1	of kt2	of kt3
ACO	2.19	2.46	3.12	2.79	1.59	2.13	3.36	1.76
	0.55	0.48	0.56	0.48	0.51	0.49	0.55	0.39
Edge DVO	×	1.51	3.68	×	1.95	×	2.46	×
		0.18	0.12		**0.16**		0.36	
PLP-SLAM	0.61	0.97	0.44	1.41	1.79	2.03	**0.71**	1.12
	0.33	0.55	0.58	0.73	1.52	0.32	**0.25**	1.23
Ours	**0.12**	**0.11**	**0.14**	**0.15**	**0.45**	**0.29**	1.18	**0.19**
	0.03	**0.02**	**0.02**	**0.06**	0.19	**0.07**	0.92	**0.04**

Boldfaced: the best. Underlined: the second best. ×: Estimations diverged.

Table 6. $\text{RPE}_{\text{trans}}$(cm) and RPE_{rot}(degree) comparisons of the ICL-NUIM dataset, with $\text{RPE}_{\text{trans}}$ above and RPE_{rot} below in each method.



(a) (b)

Figure 10. The cumulative distribution function (CDF) of the (a) $\log_{10}$ iteration numbers in the refinement stage, and (b) processing time (ms) of pose estimation when an initial pose is given by the GCC-RANSAC versus the constant speed assumption. The values were calculated from a total of 49,600 image pairs from the TUM-RGBD, ICL-NUIM, and RGBD Scene v2 datasets. Being more accurate is interpreted as having the curve close to the top left.

8. Conclusion

This paper proposes a novel way to enforce the 3D geometric constraint of length preservation in the image domain. Casting this constraint in 2D avoids the large number of false positives arising from unreasonably large distance thresholds that arise from asymmetric error distributions between depth and image localization. The paper demonstrates that working directly in the image domain is more accurate and more robust and validates it by both experiments in a synthetic setting and in realistic applications.

Acknowledgment. The support of NSF award 2312745 is gratefully acknowledged.

References

[1] OpenSfM. https://github.com/mapillary/OpenSfM/tree/main. 1

[2] Jun-Jee Chao, Selim Engin, Nicolai Häni, and Volkan Isler. Category-level global camera pose estimation with multi-hypothesis point cloud correspondences. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3800–3807. IEEE, 2023. 2

[3] David Charatan, Hongyi Fan, and Benjamin Kimia. Benchmarking pedestrian odometry: The brown pedestrian odometry dataset (bpod). In *2022 International Conference on 3D Vision (3DV)*, pages 1–11. IEEE, 2022. 1

[4] Congjia Chen, Xiaoyu Jia, Yanhong Zheng, and Yufu Qu. RGBD-Glue: General feature combination for robust RGB-D point cloud registration. *arXiv preprint arXiv:2405.07594*, 2024. 2

[5] Chiang-Heng Chien, Chiang-Ju Chien, and Chen-Chien Hsu. Hw/sw co-design and fpga acceleration of a feature-based visual odometry. In *2019 4th International Conference on Robotics and Automation Engineering (ICRAE)*, pages 148–152. IEEE, 2019. 2

[6] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2514–2523, 2020. 2

[7] Kevin Christensen and Martial Hebert. Edge-direct visual odometry. *arXiv preprint arXiv:1906.04838*, 2019. 2, 8

[8] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 1

[9] Yaqing Ding, Chiang-Heng Chien, Viktor Larsson, Karl Åström, and Benjamin Kimia. Minimal solutions to generalized three-view relative pose problem. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8156–8164, 2023. 1

[10] Mohamed El Banani, Luya Gao, and Justin Johnson. Unsupervisedr&r: Unsupervised point cloud registration via differentiable rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7129–7139, 2021. 2

[11] Mohamed El Banani, Ignacio Rocco, David Novotny, Andrea Vedaldi, Natalia Neverova, Justin Johnson, and Ben Graham. Self-supervised correspondence estimation via multiview registration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1216–1225, 2023. 1

[12] Ricardo Fabbri, Timothy Duff, Hongyi Fan, Margaret H Regan, David da C de Pinho, Elias Tsigaridas, Charles W Wampler, Jonathan D Hauenstein, Peter J Giblin, Benjamin Kimia, et al. Trifocal relative pose from lines at points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 1

[13] Alejandro Fontan, Javier Civera, and Rudolph Triebel. Information-driven direct RGB-D odometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2020. 1

[14] Alejandro Fontan, Riccardo Giubilato, Laura Oliva, Javier Civera, and Rudolph Triebel. SID-SLAM: Semi-direct information-driven RGB-D SLAM. *IEEE Robotics and Automation Letters*, 2023. 1, 2

[15] Lin Ge, Xingyue Wei, Yayu Hao, Jianwen Luo, and Yan Xu. Unsupervised histological image registration using structural feature guided convolutional neural network. *IEEE Transactions on Medical Imaging*, 41(9):2414–2431, 2022. 1

[16] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J Davison. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In *2014 IEEE international conference on Robotics and automation (ICRA)*, pages 1524–1531. IEEE, 2014. 7

[17] Berthold KP Horn. Closed-form solution of absolute orientation using unit quaternions. *Josa a*, 4(4):629–642, 1987. 1, 2

[18] Gregorij Kurillo et. al. Evaluating the accuracy of the azure kinect and kinect v2. *Sensors*, 22(7):2469, 2022. 5

[19] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3D scene labeling. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3050–3057, 2014. 7

[20] Junha Lee, Seungwook Kim, Minsu Cho, and Jaesik Park. Deep hough voting for robust global registration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 15994–16003, 2021. 2, 3

[21] Tzu-Yuan Lin, William Clark, Ryan M Eustice, Jessy W Grizzle, Anthony Bloch, and Maani Ghaffari. Adaptive continuous visual odometry from RGB-D images. *arXiv preprint arXiv:1910.00713*, 2019. 8

[22] Xi Lin, Yewei Huang, Dingyi Sun, Tzu-Yuan Lin, Brendan Englot, Ryan M Eustice, and Maani Ghaffari. A robust keyframe-based visual slam for RGB-D cameras in challenging scenarios. *IEEE Access*, 2023. 2, 8

[23] Pengpeng Liu, Guixuan Zhang, Hu Guan, Jie Liu, Shuwu Zhang, and Zhi Zengi. Relative pose estimation for rgb-d human input scans via human completion. In *2021 International Conference on Culture-oriented Science & Technology (ICCST)*, pages 471–474. IEEE, 2021. 1

[24] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004. 1, 2

[25] Hidenobu Matsuki, Riku Murai, Paul HJ Kelly, and Andrew J Davison. Gaussian splatting slam. *arXiv preprint arXiv:2312.06741*, 2023. 1, 8

[26] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. OpenMVG: Open multiple view geometry. In *Reproducible Research in Pattern Recognition: First International Workshop, RRPR 2016, Cancún, Mexico, December 4, 2016, Revised Selected Papers 1*, pages 60–74. Springer, 2017. 1

[27] Fernando I Ireta Munoz and Andrew I Comport. Point-to-hyperplane rgb-d pose estimation: Fusing photometric and geometric measurements. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 24–29. IEEE, 2016. 2

[28] Raul Mur-Artal and Juan D Tardós. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017. 1, 2, 8

[29] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ international conference on intelligent robots and systems*, pages 3384–3391. IEEE, 2008. 2

[30] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1

[31] Ilankaikone Senthooran, Manzur Murshed, Jan Carlo Barca, Joarder Kamruzzaman, and Hoam Chung. An efficient ransac hypothesis evaluation using sufficient statistics for rgb-d pose estimation. *Autonomous Robots*, 43:1257–1270, 2019. 2

[32] Fangwen Shu, Jiaxuan Wang, Alain Pagani, and Didier Stricker. Structure PLP-SLAM: Efficient sparse mapping and localization using point, line and plane for monocular, RGB-D and stereo cameras. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2105–2112. IEEE, 2023. 8

[33] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International journal of computer vision*, 80:189–210, 2008. 1

[34] Olga Sorkine-Hornung and Michael Rabinovich. Least-squares rigid motion using svd. *Computing*, 1(1):1–5, 2017. 2

[35] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 573–580, 2012. 6, 7

[36] Che Sun, Yunde Jia, Yi Guo, and Yuwei Wu. Global-aware registration of less-overlap RGB-D scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6357–6366, 2022. 1

[37] Lei Sun. RANSIC: Fast and highly robust estimation for rotation search and point cloud registration using invariant compatibility. *IEEE Robotics and Automation Letters*, 7(1): 143–150, 2021. 2

[38] Haiping Wang, Yuan Liu, Zhen Dong, Yulan Guo, Yu-Shen Liu, Wenping Wang, and Bisheng Yang. Robust multiview point cloud registration with reliable pose graph initialization and history reweighting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9506–9515, 2023. 1, 2

[39] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023. 1, 8

[40] Ziming Wang, Xiaoliang Huo, Zhenghao Chen, Jing Zhang, Lu Sheng, and Dong Xu. Improving rgb-d point cloud registration by learning multi-scale local linear transformation. In *European Conference on Computer Vision*, pages 175–191. Springer, 2022. 2

[41] Chi Yan, Delin Qu, Dong Wang, Dan Xu, Zhigang Wang, Bin Zhao, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. *arXiv preprint arXiv:2311.11700*, 2023. 1

[42] Jiaqi Yang, Zhiqiang Huang, Siwen Quan, Zhaoshuai Qi, and Yanning Zhang. SAC-COT: Sample consensus by sampling compatibility triangles in graphs for 3-d point cloud registration. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2021. 3

[43] Zhenpei Yang, Jeffrey Z Pan, Linjie Luo, Xiaowei Zhou, Kristen Grauman, and Qixing Huang. Extreme relative pose estimation for RGB-D scans via scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4531–4540, 2019. 1, 2, 3

[44] Zhenpei Yang, Siming Yan, and Qixing Huang. Extreme relative pose network under hybrid representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2455–2464, 2020. 1

[45] Zhinan Yu, Zheng Qin, Yijie Tang, Yongjun Wang, Renjiao Yi, Chenyang Zhu, and Kai Xu. Nerf-guided unsupervised learning of rgb-d registration. *arXiv preprint arXiv:2405.00507*, 2024. 2

[46] Mingzhi Yuan, Kexue Fu, Zhihao Li, Yucong Meng, and Manning Wang. Pointmbf: A multi-scale bidirectional fusion network for unsupervised rgb-d point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17694–17705, 2023. 2

[47] Rong Yuan, Hongyi Fan, and Benjamin B Kimia. Dissecting scale from pose estimation in visual odometry. In *BMVC*, pages 1–12, 2017. 1

[48] Zikang Yuan, Ken Cheng, Jinhui Tang, and Xin Yang. RGB-D DSO: Direct sparse odometry with RGB-D cameras for indoor scenes. *IEEE Transactions on Multimedia*, 24:4092–4101, 2021. 1, 2

[49] Youmin Zhang, Fabio Tosi, Stefano Mattoccia, and Matteo Poggi. Go-slam: Global optimization for consistent 3d instant reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3727–3737, 2023. 1

[50] Zichao Zhang and Davide Scaramuzza. A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7244–7251. IEEE, 2018. 7

[51] Xinyue Zhao, Quanzhi Li, Yue Chao, Quanyou Wang, Zaixing He, and Dong Liang. Rt-less: a multi-scene rgb dataset for 6d pose estimation of reflective texture-less objects. *The Visual Computer*, pages 1–14, 2023. 1

[52] Xinyang Zhao, Qinghua Li, Changhong Wang, Hexuan Dou, and Bo Liu. Robust depth-aided RGBD-inertial odometry for indoor localization. *Measurement*, page 112487, 2023. 1

[53] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 766–782. Springer, 2016. 3

[54] Wenhui Zhou, Luwei Ren, Junle Yu, Nian Qu, and Guojun Dai. Boosting RGB-D point cloud registration via explicit position-aware geometric embedding. *IEEE Robotics and Automation Letters*, 2024. 2

[55] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12796, 2022. 1, 2