

Fast Global Registration

Qian-Yi Zhou Jaesik Park Vladlen Koltun

Intel Labs

Abstract. We present an algorithm for fast global registration of partially overlapping 3D surfaces. The algorithm operates on candidate matches that cover the surfaces. A single objective is optimized to align the surfaces and disable false matches. The objective is defined densely over the surfaces and the optimization achieves tight alignment with no initialization. No correspondence updates or closest-point queries are performed in the inner loop. An extension of the algorithm can perform joint global registration of many partially overlapping surfaces. Extensive experiments demonstrate that the presented approach matches or exceeds the accuracy of state-of-the-art global registration pipelines, while being at least an order of magnitude faster. Remarkably, the presented approach is also faster than local refinement algorithms such as ICP. It provides the accuracy achieved by well-initialized local refinement algorithms, without requiring an initialization and at lower computational cost.

1 Introduction

Registration of three-dimensional surfaces is a central problem in computer vision, computer graphics, and robotics. The problem is particularly challenging when the surfaces only partially overlap and no initial alignment is given. This difficult form of the problem is encountered in scene reconstruction [7, 39], 3D object retrieval [15, 29], camera relocalization [13], and other applications.

In order to deal with noisy data and partial overlap, practical registration pipelines employ iterative model fitting frameworks such as RANSAC [31]. Each iteration samples a set of candidate correspondences, produces an alignment based on these correspondences, and evaluates this alignment. If a satisfactory alignment is found, it is refined by a local registration algorithm such as ICP [30]. The combination of sampling-based coarse alignment and iterative local refinement is common in practice and is designed to produce a tight registration even with challenging input [7, 19, 39].

While such registration pipelines are common, they have significant drawbacks. Both the model fitting and the local refinement stages are iterative and perform computationally expensive nearest-neighbor queries in their inner loops. Much of the computational effort is expended on testing candidate alignments that are subsequently discarded. And the inelegant decomposition into a global alignment stage and a local refinement stage is itself a consequence of the low precision of global alignment frameworks.

In this paper, we present a fast global registration algorithm that does not involve iterative sampling, model fitting, or local refinement. The algorithm does not require initialization and can align noisy partially overlapping surfaces. It optimizes a robust

objective defined densely over the surfaces. Due to this dense coverage, the algorithm directly produces an alignment that is as precise as that computed by well-initialized local refinement algorithms.

This direct approach has substantial benefits. It accomplishes in a single stage what is commonly done in two. This single stage optimizes a clear global objective. The optimization does not require closest-point queries in the inner loop. As a result, the presented algorithm is more than an order of magnitude faster than existing global registration pipelines, while matching or exceeding their accuracy.

Furthermore, we show that the presented algorithm can be extended to direct global alignment of multiple partially overlapping surfaces. Such joint alignment is often necessary in applications such as scene reconstruction [7, 39]. Existing approaches to this problem exhaustively produce candidate alignments between pairs of surfaces and then compute a globally consistent set of poses based on these intermediate pairwise alignments. In contrast, we show that a joint alignment can be produced directly by a single optimization of a global objective.

We evaluate the presented global registration algorithm on multiple datasets. Extensive experiments demonstrate that the presented approach matches or exceeds the accuracy of state-of-the-art global registration pipelines, while being at least an order of magnitude faster. Remarkably, the presented approach is also faster than local refinement algorithms such as ICP, since it does not need to recompute correspondences. It provides the accuracy achieved by well-initialized local refinement algorithms, without requiring an initialization and at lower computational cost.

2 Related Work

Geometric registration has been extensively studied [15, 28, 36, 38]. The typical workflow consists of two stages: global alignment, which computes an initial estimate of the rigid motion between two surfaces, followed by local refinement, which refines this initial estimate to obtain a tight registration [7, 12, 19, 25, 27, 37, 39]. We review each of these stages in turn.

Most global alignment methods operate on candidate correspondences. Some pipelines use point-to-point matches based on local geometric descriptors [16, 40], others define correspondences on pairs or tuples of points [1, 8, 26, 29]. Once candidate correspondences are collected, alignment is estimated iteratively from sparse subsets of correspondences and then validated on the entire surface. This iterative process is typically based on variants of RANSAC [1, 19, 26, 29, 34] or pose clustering [8, 26, 35]. When the data is noisy and the surfaces only partially overlap, existing pipelines often require many iterations to sample a good correspondence set and find a reasonable alignment.

Another approach to global registration is based on the branch-and-bound framework [10, 12, 18, 23, 42]. These algorithms systematically explore the pose space in search of the optimal solution. The branch-and-bound framework is appealing due to its theoretical optimality guarantees. However, the systematic search can be extremely time-consuming. In practice, the sampling-based frameworks described earlier outperform the branch-and-bound approaches when large datasets are involved.

Local refinement algorithms begin with a rough initial alignment and produce a tight registration based on dense correspondences. Most such methods are based on the iterative closest point (ICP) algorithm and its variants [30, 33]. In its basic form, ICP begins with an initial alignment and alternates between establishing correspondences via closest-point lookups and recomputing the alignment based on the current set of correspondences. ICP can produce an accurate result when initialized near the optimal pose, but is unreliable without such initialization. A long line of work has explored various approaches to increasing the robustness of ICP. Fitzgibbon [11] introduced nonlinear least-squares optimization to develop a robust error function that increases the radius of convergence. Bouaziz et al. [5] introduced sparsity inducing norms to deal with outliers and incomplete data. Other works explored the utility of relaxed assignments [14, 24, 32], distance field representations [6], and mixture models [21, 41] for increasing the robustness of local registration. Nevertheless, these approaches still rely on a satisfactory initialization. Our work demonstrates that the accuracy achieved by well-initialized local refinement algorithms can be achieved reliably without an initialization, at a computational cost that is more than an order of magnitude lower than the coarse global alignment algorithms described earlier.

Joint global registration of multiple partially overlapping surfaces has also been considered [7, 20, 39]. However, existing approaches to joint global registration first align many pairs of surfaces and then optimize the joint global alignment based on these intermediate pairwise results. This indirect approach incurs significant computational overhead. In contrast, we show that joint global alignment of many partially overlapping surfaces can be optimized for directly.

3 Pairwise Global Registration

3.1 Objective

Consider two point sets \mathbf{P} and \mathbf{Q} . Our task is to find a rigid transformation \mathbf{T} that aligns \mathbf{Q} to \mathbf{P} . Our approach optimizes a robust objective on correspondences between \mathbf{P} and \mathbf{Q} . These correspondences are established by rapid feature matching that is performed before the objective is optimized. The correspondences are not recomputed during the optimization. For this reason, it is critical that the optimization be able to deal with very noisy correspondence sets. This is illustrated in Figure 1.

Let $\mathcal{K} = \{(\mathbf{p}, \mathbf{q})\}$ be the set of correspondences collected by matching points from \mathbf{P} and \mathbf{Q} as described in Section 3.3. Our objective is to optimize the pose \mathbf{T} such that distances between corresponding points are minimized, while spurious correspondences from \mathcal{K} are seamlessly disabled. The objective has the following form:

$$E(\mathbf{T}) = \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{K}} \rho(\|\mathbf{p} - \mathbf{T}\mathbf{q}\|). \quad (1)$$

Here $\rho(\cdot)$ is a robust penalty. The use of an appropriate robust penalty function is critical, because many of the terms in objective 1 are contributed by spurious constraints. To achieve high computational efficiency, we do not want to sample, validate, prune, or recompute correspondences during the optimization. A well-chosen estimator ρ will

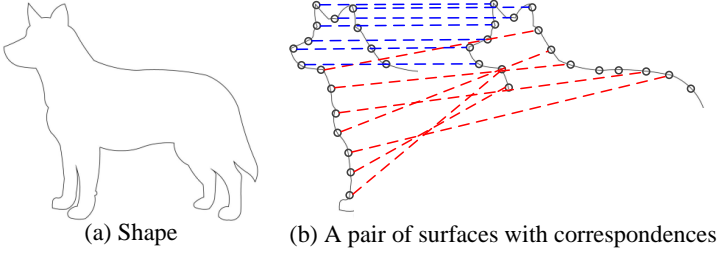


Fig. 1. An illustration with 2D point sets. (a) A latent shape. (b) Two partially overlapping surfaces and a set of point-to-point correspondences. The blue correspondences are genuine, the red correspondences are erroneous. For fast and accurate registration, the erroneous correspondences must be disabled without sampling, validation, pruning, or correspondence recomputation.

perform the validation and pruning automatically without imposing additional computational costs. We use a scaled Geman-McClure estimator:

$$\rho(x) = \frac{\mu x^2}{\mu + x^2}. \quad (2)$$

Figure 2(a) shows the Geman-McClure estimator for different values of μ . As can be seen in the figure, small residuals are penalized in the least-squares sense, while the sub-linear growth and rapid flattening out of the estimator neutralize outliers. The parameter μ controls the range within which residuals have a significant effect on the objective; its setting will be discussed in Section 3.2.

Objective 1 is difficult to optimize directly. We use the Black-Rangarajan duality between robust estimation and line processes [3]. Specifically, let $\mathbb{L} = \{l_{\mathbf{p},\mathbf{q}}\}$ be a line process over the correspondences. We optimize the following joint objective over \mathbf{T} and \mathbb{L} :

$$E(\mathbf{T}, \mathbb{L}) = \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}} l_{\mathbf{p},\mathbf{q}} \|\mathbf{p} - \mathbf{T}\mathbf{q}\|^2 + \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}} \Psi(l_{\mathbf{p},\mathbf{q}}). \quad (3)$$

Here $\Psi(l_{\mathbf{p},\mathbf{q}})$ is a prior, set to

$$\Psi(l_{\mathbf{p},\mathbf{q}}) = \mu(\sqrt{l_{\mathbf{p},\mathbf{q}}} - 1)^2. \quad (4)$$

For $E(\mathbf{T}, \mathbb{L})$ to be minimized, the partial derivative with respect to each $l_{\mathbf{p},\mathbf{q}}$ must vanish:

$$\frac{\partial E}{\partial l_{\mathbf{p},\mathbf{q}}} = \|\mathbf{p} - \mathbf{T}\mathbf{q}\|^2 + \mu \frac{\sqrt{l_{\mathbf{p},\mathbf{q}}} - 1}{\sqrt{l_{\mathbf{p},\mathbf{q}}}} = 0. \quad (5)$$

Solving for $l_{\mathbf{p},\mathbf{q}}$ yields

$$l_{\mathbf{p},\mathbf{q}} = \left(\frac{\mu}{\mu + \|\mathbf{p} - \mathbf{T}\mathbf{q}\|^2} \right)^2. \quad (6)$$

Substituting $l_{\mathbf{p},\mathbf{q}}$ into $E(\mathbf{T}, \mathbb{L})$, objective 3 becomes objective 1. Thus optimizing objective 3 yields a solution \mathbf{T} that is also optimal for the original objective 1.

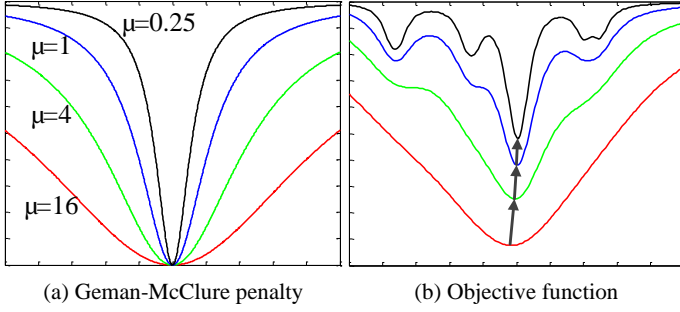


Fig. 2. Illustration of graduated non-convexity. As μ decreases, the objective function for the matching problem in Figure 1 becomes sharper and the registration more precise.

3.2 Optimization

The main benefit of the optimization objective defined in equation 3 is that the optimization can be performed extremely efficiently by alternating between optimizing \mathbf{T} and \mathbb{L} . The optimization performs block coordinate descent by fixing \mathbb{L} when optimizing \mathbf{T} and vice versa. Both types of steps optimize the same global objective (equation 3). Thus the alternating algorithm is guaranteed to converge.

When \mathbb{L} is fixed, objective 3 turns into a weighted sum of L^2 penalties on distances between point-to-point correspondences. This objective over \mathbf{T} can be solved efficiently in closed form [9]. However, such closed-form solution does not extend to joint registration of multiple surfaces, which we are interested in and will extend the presented approach to in Section 4. We therefore present a more flexible approach. We linearize \mathbf{T} locally as a 6-vector $\xi = (\omega, \mathbf{t}) = (\alpha, \beta, \gamma, a, b, c)$ that collates a rotational component ω and a translation \mathbf{t} . \mathbf{T} is approximated by a linear function of ξ :

$$\mathbf{T} \approx \begin{pmatrix} 1 & -\gamma & \beta & a \\ \gamma & 1 & -\alpha & b \\ -\beta & \alpha & 1 & c \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{T}^k. \quad (7)$$

Here \mathbf{T}^k is the transformation estimated in the last iteration. Equation 3 becomes a least-squares objective on ξ . Using the Gauss-Newton method, ξ is computed by solving a linear system:

$$\mathbf{J}_r^\top \mathbf{J}_r \xi = -\mathbf{J}_r^\top \mathbf{r}, \quad (8)$$

where \mathbf{r} is the residual vector and \mathbf{J}_r is its Jacobian. \mathbf{T} is updated by applying ξ to \mathbf{T}^k using equation 7, then mapped back into the $SE(3)$ group.

When \mathbf{T} is fixed, the objective in equation 3 has a closed-form solution. It is minimized when $l_{p,q}$ satisfies equation 6.

Graduated non-convexity. Objective 3 is non-convex and its shape is controlled by the parameter μ of the penalty function (equation 2). To set μ and alleviate the effect

of local minima we employ graduated non-convexity [4, 2]. From the standpoint of equation 3, μ balances the strength of the prior term and the alignment term. Large μ makes the objective function smoother and allows many correspondences to participate in the optimization even when they are not fit tightly by the transformation \mathbf{T} . The effect of varying μ is illustrated in Figure 2. Our optimization begins with a very large value $\mu = D^2$, where D is the diameter of the largest surface. The parameter μ is decreased during the optimization until it reaches the value $\mu = \delta^2$, where δ is a distance threshold for genuine correspondences.

3.3 Correspondences

To generate the initial correspondence set \mathcal{K} , we use the Fast Point Feature Histogram (FPFH) feature [34]. We have chosen this feature because it can be computed in a fraction of a millisecond and provides good matching accuracy across a broad range of datasets [16]. Let $\mathbf{F}(\mathbf{P}) = \{\mathbf{F}(\mathbf{p}) : \mathbf{p} \in \mathbf{P}\}$, where $\mathbf{F}(\mathbf{p})$ is the FPFH feature computed for point \mathbf{p} . Define $\mathbf{F}(\mathbf{Q}) = \{\mathbf{F}(\mathbf{q}) : \mathbf{q} \in \mathbf{Q}\}$ analogously.

For each $\mathbf{p} \in \mathbf{P}$, we find the nearest neighbor of $\mathbf{F}(\mathbf{p})$ among $\mathbf{F}(\mathbf{Q})$, and for each $\mathbf{q} \in \mathbf{Q}$ we find the nearest neighbor of $\mathbf{F}(\mathbf{q})$ among $\mathbf{F}(\mathbf{P})$. Let \mathcal{K}_I be the set that collects all these correspondences. This set could be used directly as the input to our approach. However, in practice \mathcal{K}_I has a very high fraction of outliers. We use two tests to improve the inlier ratio of the correspondence set used by the algorithm.

- **Reciprocity test.** A correspondence pair (\mathbf{p}, \mathbf{q}) is selected from \mathcal{K}_I if and only if $\mathbf{F}(\mathbf{p})$ is the nearest neighbor of $\mathbf{F}(\mathbf{q})$ among $\mathbf{F}(\mathbf{P})$ and $\mathbf{F}(\mathbf{q})$ is the nearest neighbor of $\mathbf{F}(\mathbf{p})$ among $\mathbf{F}(\mathbf{Q})$. The resulting correspondence set is denoted by \mathcal{K}_{II} .
- **Tuple test.** We randomly pick 3 correspondence pairs $(\mathbf{p}_1, \mathbf{q}_1), (\mathbf{p}_2, \mathbf{q}_2), (\mathbf{p}_3, \mathbf{q}_3)$ from \mathcal{K}_{II} and check if the tuples $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ and $(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3)$ are compatible. Specifically, we test if the following condition is met:

$$\forall i \neq j, \quad \tau < \frac{\|\mathbf{p}_i - \mathbf{p}_j\|}{\|\mathbf{q}_i - \mathbf{q}_j\|} < 1/\tau, \quad (9)$$

where $\tau = 0.9$. Intuitively, this test verifies that the correspondences are compatible. Correspondences from tuples that pass the test are collected in a set \mathcal{K}_{III} . This is the set used by the algorithm: $\mathcal{K} = \mathcal{K}_{III}$.

Algorithm 1 summarizes the pairwise registration algorithm used in all subsequent experiments.

4 Multi-Way Registration

Many applications require aligning multiple surfaces to obtain a model of a large scene or object. To solve this multi-way registration problem, existing approaches first compute pairwise alignments between pairs of surfaces and then attempt to synchronize these alignments to obtain a global registration [7, 20, 39]. This has two significant disadvantages. First, the pairwise alignment stage is computationally wasteful because it

Algorithm 1: Fast pairwise registration**input** : A pair of surfaces (\mathbf{P}, \mathbf{Q}) **output**: Transformation \mathbf{T} that aligns \mathbf{Q} to \mathbf{P} Compute normals $\{\mathbf{n}_\mathbf{P}\}$ and $\{\mathbf{n}_\mathbf{Q}\}$;Compute FPFH features $\mathbf{F}(\mathbf{P})$ and $\mathbf{F}(\mathbf{Q})$;Build \mathcal{K}_I by computing nearest neighbors between $\mathbf{F}(\mathbf{P})$ and $\mathbf{F}(\mathbf{Q})$;Apply reciprocity test on \mathcal{K}_I to get \mathcal{K}_{II} ;Apply tuple test on \mathcal{K}_{II} to get \mathcal{K}_{III} ; $\mathbf{T} \leftarrow \mathbf{I}, \mu \leftarrow D^2$;**while** not converged *or* $\mu > \delta^2$ **do** $\mathbf{J}_\mathbf{r} \leftarrow \mathbf{0}, \mathbf{r} \leftarrow \mathbf{0}$; **for** $(\mathbf{p}, \mathbf{q}) \in \mathcal{K}_{III}$ **do** Compute $l_{(\mathbf{p}, \mathbf{q})}$ using equation 6; Update $\mathbf{J}_\mathbf{r}$ and \mathbf{r} of objective 3; Solve equation 8 and update \mathbf{T} ; Every four iterations, $\mu \leftarrow \mu/2$;Verify whether \mathbf{T} aligns \mathbf{Q} to \mathbf{P} ;

is not apparent in advance which pairs will be useful. Second, pairwise registration can yield a suboptimal alignment due to local minima that could be disambiguated by a global approach that considers all surfaces jointly.

We develop an alternative approach: to directly align all surfaces based on raw dense point correspondences. Instead of optimizing separate pairwise alignments and then synchronizing the results, we can directly optimize a global registration objective over all surfaces.

4.1 Objective

Given a set of surfaces $\{\mathbf{P}_i\}$, our task is to estimate a set of poses $\mathbb{T} = \{\mathbf{T}_i\}$ that aligns the surfaces in a global coordinate frame. We begin by constructing a set of candidate correspondences \mathcal{K}_{ij} for each pair of surfaces $(\mathbf{P}_i, \mathbf{Q}_j), i < j$. Objective 1 is extended to the multi-way setting as follows:

$$E(\mathbb{T}) = \lambda \sum_i \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{K}_i} \|\mathbf{T}_i \mathbf{p} - \mathbf{T}_{i+1} \mathbf{q}\|^2 + \sum_{i < j} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{K}_{ij}} \rho(\|\mathbf{T}_i \mathbf{p} - \mathbf{T}_j \mathbf{q}\|). \quad (10)$$

This formulation incorporates initial odometry transformations $\{\mathbf{T}_i\}$ between consecutive surfaces, which are commonly available in surface reconstruction. The set \mathcal{K}_i collects correspondences between surfaces \mathbf{P}_i and \mathbf{P}_{i+1} under the odometry alignment. When available, the odometry terms are penalized directly with the L^2 norm and serve as a backbone that stabilizes the optimization.

Define a line process $\mathbb{L} = \{l_{\mathbf{p},\mathbf{q}}\}$. The objective can now be reformulated as follows:

$$E(\mathbb{T}, \mathbb{L}) = \lambda \sum_i \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}_i} \|\mathbf{T}_i \mathbf{p} - \mathbf{T}_{i+1} \mathbf{q}\|^2 + \sum_{i < j} \left(\sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}_{ij}} l_{\mathbf{p},\mathbf{q}} \|\mathbf{T}_i \mathbf{p} - \mathbf{T}_j \mathbf{q}\|^2 + \sum_{(\mathbf{p},\mathbf{q}) \in \mathcal{K}_{ij}} \Psi(l_{\mathbf{p},\mathbf{q}}) \right). \quad (11)$$

The prior term $\Psi(l_{\mathbf{p},\mathbf{q}})$ is defined as in equation 4.

4.2 Optimization

We again use alternating optimization to solve the minimization problem. In each iteration, $E(\mathbb{T}, \mathbb{L})$ is first minimized with respect to the line process variables \mathbb{L} . This has a closed-form solution:

$$l_{\mathbf{p},\mathbf{q}} = \left(\frac{\mu}{\mu + \|\mathbf{T}_i \mathbf{p} - \mathbf{T}_j \mathbf{q}\|^2} \right)^2. \quad (12)$$

Next, $E(\mathbb{T}, \mathbb{L})$ is minimized with respect to all poses \mathbb{T} . Let \mathbf{T}_i^k denote the i -th transformation estimated in the previous iteration. \mathbf{T}_i can be locally linearized with a 6-vector $\xi_i = (\omega_i, \mathbf{t}_i) = (\alpha_i, \beta_i, \gamma_i, a_i, b_i, c_i)$:

$$\mathbf{T}_i \approx \begin{pmatrix} 1 & -\gamma_i & \beta_i & a_i \\ \gamma_i & 1 & -\alpha_i & b_i \\ -\beta_i & \alpha_i & 1 & c_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{T}_i^k. \quad (13)$$

Let Ξ be a $6|\mathbb{T}|$ -vector that collates $\{\xi_i\}$. $E(\mathbb{T}, \mathbb{L})$ becomes a least-squares objective on Ξ . It is minimized by solving the linear system

$$\mathbf{J}_r^\top \mathbf{J}_r \Xi = -\mathbf{J}_r^\top \mathbf{r} \quad (14)$$

and updating \mathbf{T}_i accordingly. Here \mathbf{J}_r and \mathbf{r} are the Jacobian matrix and the residual vector, respectively.

Note that the correspondences are never updated. Each iteration performs only two steps: evaluate a line process variable for each point correspondence, then build and solve a linear system with $6|\mathbb{T}|$ variables. Both steps are very efficient.

5 Results

5.1 Pairwise registration

We evaluate the presented pairwise registration algorithm on synthetic range data, the UWA benchmark [27], and the global registration benchmark of Choi et al. [7]. We compare our algorithm with a number of prior global registration methods. `GoICP` is the algorithm of Yang et al. [42]. `GoICP-Trimming` is its trimming variant that supports

partial overlap. We use a 10% trimming percentage and use only 1,000 data points, as suggested by Yang et al. [42]. Without downsampling, GoICP and GoICP-Trimming take hours to run on our point clouds. Super4PCS is the algorithm of Mellado et al. [26]. OpenCV is a recent OpenCV implementation of the surface registration algorithm of Drost et al. [8]. PCL is a Point Cloud Library implementation of the algorithm of Rusu et al. [34, 19]. CZK is the variant of Rusu’s algorithm used by Choi et al. [7].

We also conduct controlled comparisons with local registration algorithms. PCL ICP is a Point Cloud Library implementation of the ICP algorithm [19]. Sparse ICP is the algorithm of Bouaziz et al. [5]. We tested these algorithms with both point-to-point and point-to-plane distance measures [33].

All execution times are measured using a single thread on an Intel Core i7-5960X CPU clocked at 3.00GHz.

Synthetic range data. We begin by performing a series of controlled experiments on synthetic data. The availability of precise ground truth enables a detailed evaluation. To conduct controlled experiments, we used three well-known models from the AIM@SHAPE repository (Bimba, Dancing Children, and Chinese Dragon), the Berkeley Angel dataset [22], and the Stanford Bunny. For each model, we synthesized five pairs of partially overlapping range images and then corrupted these range images with 3D Gaussian noise. We used three noise levels, defined by setting the standard deviation of the Gaussian distribution to $\sigma = 0$ (no noise), $\sigma = 0.0025$, and $\sigma = 0.005$. The unit of σ is the diameter of the surface. For each noise level, there are 25 partially overlapping global registration tests in total. The number of points in each range image varies between 8,868 and 19,749. The overlap ratio varies between 47% and 90%.

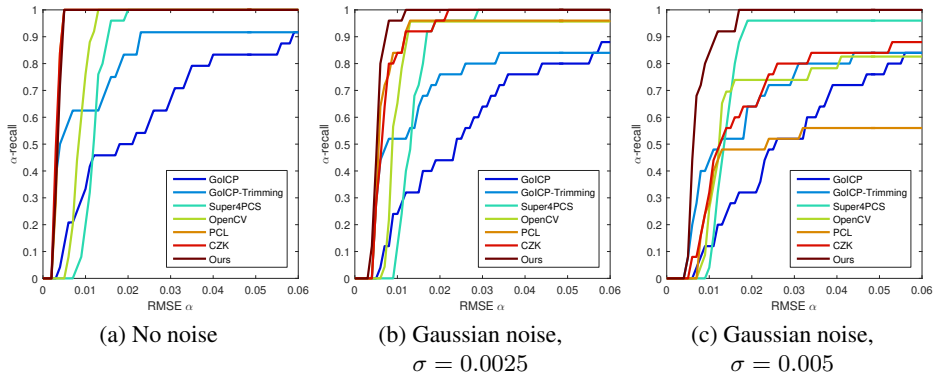


Fig. 3. Controlled experiments on synthetic data. α -recall is the fraction of tests for which a given method achieves an $\text{RMSE} < \alpha$. Higher is better. The RMSE unit is the diameter of the surface. Our algorithm is more robust to noise and is more accurate than prior approaches, while being more than an order of magnitude faster.

Figure 3 shows the accuracy achieved by the different global registration algorithms on the 25 tests at each noise level. For each algorithm and each RMSE level α , the fig-

Table 1. Average and maximal RMSE achieved by global registration algorithms on synthetic range images with noise level σ . Maximal RMSE is the maximum among the 25 RMSE values obtained for individual pairwise registration tests. Our approach outperforms other methods by a large margin when noise is present. Specifically, for $\sigma = 0.005$, the average RMSE of our approach is more than 2 times lower than the lowest average RMSE of any prior approach, and the maximal RMSE of our approach is 5.6 times lower.

	$\sigma = 0$		$\sigma = 0.0025$		$\sigma = 0.005$	
	Average RMSE	Maximal RMSE	Average RMSE	Maximal RMSE	Average RMSE	Maximal RMSE
GoICP [42]	0.029	0.130	0.032	0.133	0.037	0.127
GoICP-Trimming [42]	0.035	0.473	0.039	0.475	0.044	0.478
Super 4PCS [26]	0.012	0.019	0.014	0.029	0.017	0.095
OpenCV [8]	0.009	0.013	0.018	0.212	0.032	0.242
PCL [34, 19]	0.003	0.005	0.009	0.061	0.111	0.414
CZK [7]	0.003	0.005	0.008	0.022	0.035	0.274
Our approach	0.003	0.005	0.006	0.011	0.008	0.017

ure plots the α -recall, defined as the fraction of tests for which the method achieved an $\text{RMSE} < \alpha$. (Higher is better.) The RMSE is computed on the distances between ground-truth correspondences after alignment. Table 1 summarizes the average and maximal RMSE for each method. (Lower is better.) For synthetic data with no noise, our method, PCL, and CZK produce tight alignment in 100% of the tests. ($\text{RMSE} \leq 0.005$.) The accuracy of OpenCV and Super4PCS is worse by a multiplicative factor of at least 3, presumably due to their reliance on matching tuples of points rather than optimizing for fully dense surface registration. GoICP-Trimming produces accurate alignment in many cases but suffers from poor accuracy on others, presumably because its computational costs necessitate operation on downsampled point clouds.

On noisy data, our method is much more robust and accurate than others. For $\sigma = 0.005$, the average RMSE of our approach is more than 2 times smaller than the lowest average RMSE of any prior approach, and the maximal RMSE of our approach is 5.6 times smaller than the lowest maximal RMSE among prior approaches. This is presumably because our approach optimizes over dense correspondences rather than matching point tuples. A qualitative comparison with GoICP-Trimming and PCL is provided in Figure 4.

Table 2. Running times of global registration methods, measured in seconds. GoICP and its trimming variant operate on point clouds downsampled to 1,000. All other methods operate on full-resolution point clouds. Our algorithm is 50 times faster than the fastest prior global registration method.

	Average # of points	GoICP [42]	GoICP-Trimming [42]	OpenCV [8]	Super 4PCS [26]	PCL [34, 19]	CZK [7]	Our approach
Bimba	9,416	19.3	19.4	41.0	311.4	18.2	12.8	0.13
Children	11,148	21.0	19.2	136.3	238.2	4.8	6.6	0.20
Dragon	11,232	94.1	38.4	57.7	483.7	8.6	11.9	0.23
Angel	12,072	21.0	20.4	80.9	171.5	8.7	11.3	0.26
Bunny	13,357	74.7	72.4	12.3	283.8	55.6	12.7	0.28
Average	11,445	46.0	34.0	65.6	297.7	19.2	11.1	0.22

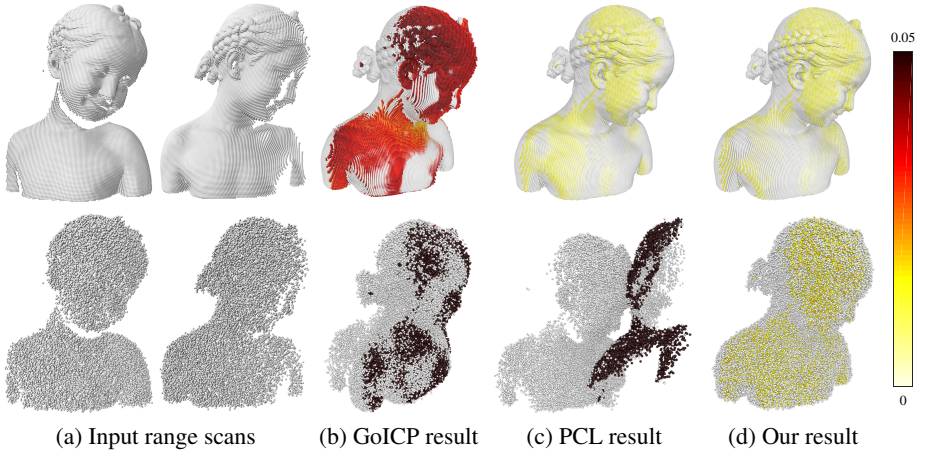
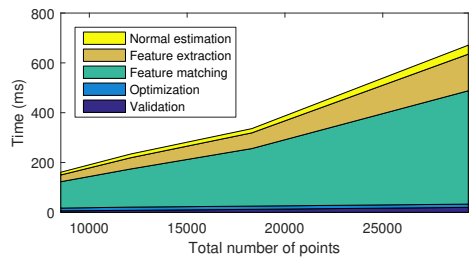


Fig. 4. Visual comparison with GoICP-Trimming and PCL. Our method operates on dense point clouds and produces a tight alignment with RMSE 0.004 on clean data (top row) and RMSE 0.007 on noisy data (bottom row, $\sigma = 0.005$). In contrast, the prior approaches break down in the presence of noise: RMSE 0.129 for GoICP-Trimming and 0.326 for PCL in the bottom row. Error magnitude is coded by color, with black indicating error above 0.05.

The major benefit of our approach is that it is faster by more than an order of magnitude than prior approaches. Table 2 shows the average computation time of each global registration method on each object. Our method improves registration speed by a factor of 50 relative to the fastest prior global registration algorithm (CZK). While previous methods require tens of seconds, our method takes 0.2 seconds on average. This is because our method avoids expensive nearest-neighbor lookups in the optimization loop.

We further analyze the computational requirements of our approach by varying the size (i.e., the point count) of the synthesized range images and measuring the execution time of individual components of our algorithm. The results are shown on the right. The majority of the time is spent on computing the FPFH features and building the input correspondences.

These operations are performed only once, before the optimization, and the correspondences are never updated. The optimization itself is extremely fast. Its execution time is below 30 milliseconds even for point clouds with more than 20,000 points. In addition, our method performs validation only once, after optimization has converged. This one-time validation consumes on average 3.3% of our computation time. In contrast, sampling-based methods such as PCL perform validation thousands of times in the RANSAC loop.



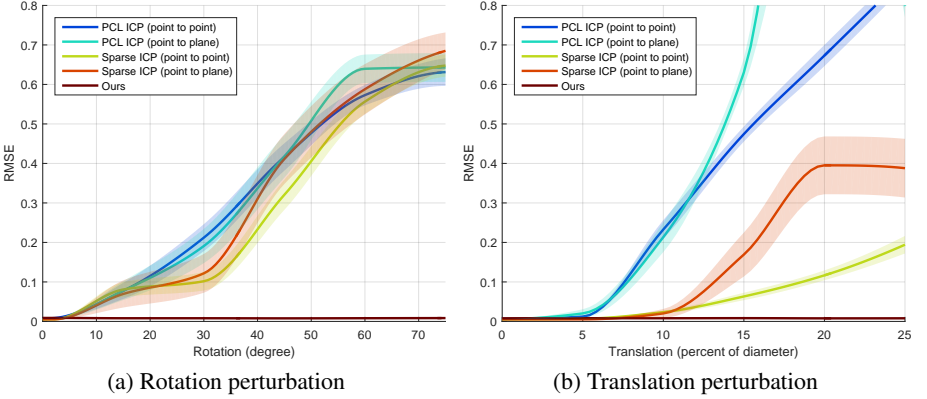


Fig. 5. Controlled comparison with local methods. Local registration algorithms are initialized with a transformation generated by adding a perturbation in rotation (left) or translation (right) to the ground-truth alignment. The plots show the mean (bold curve) and standard deviation (shaded region) of the RMSE of each method. Lower is better. Our algorithm matches the accuracy achieved by the local algorithms when they are initialized near the ground-truth pose, but does not require an initialization.

We also compare our global registration algorithm with local refinement methods such as ICP and its variants. To perform a controlled evaluation, we varied the accuracy of the initial transformation provided to the local methods. The results are shown in Figure 5. We performed two sets of experiments. In one, the local algorithms were initialized with the ground-truth translation and varying degrees of rotation. In the other, the local algorithms were initialized with the ground-truth rotation and varying degrees of translation. As shown in Figure 5, our algorithm matches the accuracy of the local refinement methods in the idealized case when these methods are initialized with the ground-truth transformation. However, the accuracy of the local methods degrades when the initialization deviates sufficiently from the ground-truth pose: 5 degrees in rotation or 5-10% of the point cloud diameter in translation. In contrast, our algorithm does not use an initialization and yields the same accuracy in all conditions.

We further compare the computational costs of our algorithm and the local refinement methods. The results are shown in Table 3. Remarkably, our global registration

Table 3. Timing comparison with local algorithms, measured in seconds. Our global algorithm is 2.8 times faster than a state-of-the-art implementation of ICP.

	Average # of points	PCL ICP point-to-point	PCL ICP point-to-plane	Sparse ICP point-to-point [5]	Sparse ICP point-to-plane [5]	Our approach
Bimba	9,416	0.73	0.31	3.1	11.8	0.13
Children	11,148	0.75	0.46	3.9	15.0	0.20
Dragon	11,232	0.99	0.47	3.6	13.8	0.23
Angel	12,072	0.81	1.01	4.9	18.5	0.26
Bunny	13,357	2.10	1.70	9.2	10.3	0.28
Average	11,445	1.08	0.79	4.9	13.9	0.22

algorithm is 2.8 times faster than a state-of-the-art implementation of ICP. The key reason is that our algorithm does not need to recompute correspondences.

UWA benchmark. Next, we evaluate our method on the UWA dataset [27]. This dataset has 50 scenes. Each scene has multiple objects that can be aligned to it. In total, the dataset contains 188 pairwise registration tests. Figure 6(a) shows a scene with objects aligned to it by our approach. This dataset is challenging due to clutter, occlusion, and low overlap. The lowest overlap ratio in the dataset is only 21%. As shown in Figure 6(b), many prior global registration algorithms perform poorly on this dataset. Our algorithm achieves a 0.05-recall of 84%, comparable with PCL and CZK (82% and 78%, respectively). OpenCV achieves 52% and the other algorithms are all below 7%.

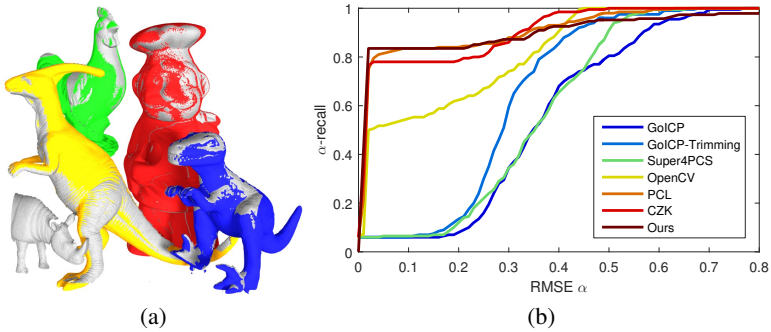


Fig. 6. Global registration results on the UWA benchmark [27]. (a) Our result on one of the 188 tests. The scene is colored white and objects aligned to the scene have distinct colors. (b) α -recall plot comparing our method and prior global registration algorithms. (Higher is better.)

Table 4 compares the speed of our approach with the other global registration methods on the UWA benchmark. Our approach is an order of magnitude faster than the fastest prior methods.

Table 4. Average running times of global registration methods on the 188 tests from the UWA dataset, measured in seconds.

	GoICP [42]	GoICP-Trimming [42]	OpenCV [8]	Super4PCS [26]	PCL [34, 19]	CZK [7]	Our approach
Average time	18.7	18.6	17.6	77.4	8.2	8.7	0.5

Scene benchmark. We now evaluate on the scene benchmark provided by Choi et al. [7]. This benchmark has 4 datasets. Each dataset consists of 47 to 57 fragments of a scene. These fragments contain high-frequency noise and low-frequency distortion that simulate scans created by consumer depth cameras. Global pairwise registration is

Table 5. Evaluation on the scene benchmark of Choi et al. [7]. Our approach has the highest precision and the second highest recall, while being at least an order of magnitude faster.

	OpenCV [8]	Super 4PCS [26]	PCL [34, 19]	CZK [7]	Our approach
Recall (%)	5.3	17.8	44.9	59.2	51.1
Precision (%)	1.6	10.4	14.0	19.6	23.2
Avg. Time (sec)	10	62	3	8	0.2

performed on every pair of fragments from a given scene. Table 5 compares recall and precision (as defined by Choi et al.), and average running times.

5.2 Multi-way registration

We evaluate the multi-way extension of our algorithm on the Augmented ICL-NUIM dataset [7, 17]. The dataset contains four sequences, each of which consists of 47 to 57 scene fragments. We apply the multi-way registration algorithm presented in Section 4 to these fragments. (The same parameter $\lambda = 2$ was used for all experiments.) Our method produces a global alignment of all fragments. We integrate these aligned fragments and report the mean distance of the reconstructed surface to the ground-truth model. Table 6 reports the resulting accuracy and running time. The reconstruction accuracy yielded by our direct multi-way registration algorithm matches the accuracy of the registration approach of Choi et al. [7]. However, our algorithm solves for the joint global alignment directly, without exhaustive intermediate pairwise alignments. It is therefore 60 times faster than the approach of Choi et al.

	Mean error (meters)		Time (seconds)	
	Choi et al. [7]	Ours	Choi et al. [7]	Ours
Living room 1	0.04	0.05	8,940	131
Living room 2	0.07	0.06	3,360	81
Office 1	0.03	0.03	4,500	69
Office 2	0.04	0.05	4,080	48
Average	0.05	0.05	5,220	82

Table 6. Evaluation of multi-way registration on the Augmented ICL-NUIM dataset [7, 17]. Our multi-way registration algorithm matches the accuracy of the state-of-the-art multi-way registration pipeline of Choi et al., but is 60 times faster.

6 Conclusion

We have presented a fast algorithm for global registration of partially overlapping 3D surfaces. Our algorithm is more than an order of magnitude faster than prior global registration algorithms and is much more robust to noise. It matches the accuracy of well-initialized local refinement algorithms such as ICP, without requiring an initialization and at lower computational cost. The algorithm may be broadly applicable in computer vision, computer graphics, and robotics.

References

1. Aiger, D., Mitra, N.J., Cohen-Or, D.: 4-points congruent sets for robust pairwise surface registration. *ACM Transactions on Graphics* 27(3) (2008)
2. Black, M.J., Anandan, P.: The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding* (1996)
3. Black, M.J., Rangarajan, A.: On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *IJCV* 19(1) (1996)
4. Blake, A., Zisserman, A.: *Visual Reconstruction*. MIT Press (1987)
5. Bouaziz, S., Tagliasacchi, A., Pauly, M.: Sparse iterative closest point. In: *Symposium on Geometry Processing* (2013)
6. Bylow, E., Sturm, J., Kerl, C., Kahl, F., Cremers, D.: Real-time camera tracking and 3D reconstruction using signed distance functions. In: *RSS* (2013)
7. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: *CVPR* (2015)
8. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: Efficient and robust 3D object recognition. In: *CVPR* (2010)
9. Eggert, D.W., Lorusso, A., Fisher, R.B.: Estimating 3-D rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications* 9(5/6) (1997)
10. Enqvist, O., Josephson, K., Kahl, F.: Optimal correspondences from pairwise constraints. In: *ICCV* (2009)
11. Fitzgibbon, A.W.: Robust registration of 2D and 3D point sets. *Image and Vision Computing* 21(13-14) (2003)
12. Gelfand, N., Mitra, N.J., Guibas, L.J., Pottmann, H.: Robust global registration. In: *Symposium on Geometry Processing* (2005)
13. Glocker, B., Izadi, S., Shotton, J., Criminisi, A.: Real-time RGB-D camera relocalization. In: *ISMAR* (2013)
14. Granger, S., Pennec, X.: Multi-scale EM-ICP: A fast and robust approach for surface registration. In: *ECCV* (2002)
15. Guo, Y., Bennamoun, M., Soheli, F.A., Lu, M., Wan, J.: 3D object recognition in cluttered scenes with local surface features: A survey. *PAMI* 36(11) (2014)
16. Guo, Y., Bennamoun, M., Soheli, F.A., Lu, M., Wan, J., Kwok, N.M.: A comprehensive performance evaluation of 3D local feature descriptors. *IJCV* 116(1) (2016)
17. Handa, A., Whelan, T., McDonald, J., Davison, A.J.: A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM. In: *ICRA* (2014)
18. Hartley, R.I., Kahl, F.: Global optimization through searching rotation space and optimal estimation of the essential matrix. In: *ICCV* (2007)
19. Holz, D., Ichim, A.E., Tombari, F., Rusu, R.B., Behnke, S.: Registration with the point cloud library: A modular framework for aligning in 3-D. *IEEE Robotics and Automation Magazine* 22(4) (2015)
20. Huber, D.F., Hebert, M.: Fully automatic registration of multiple 3D data sets. *Image and Vision Computing* 21(7) (2003)
21. Jian, B., Vemuri, B.C.: Robust point set registration using Gaussian mixture models. *PAMI* 33(8) (2011)
22. Kolluri, R.K., Shewchuk, J.R., O'Brien, J.F.: Spectral surface reconstruction from noisy point clouds. In: *Symposium on Geometry Processing* (2004)
23. Li, H., Hartley, R.I.: The 3D-3D registration problem revisited. In: *ICCV* (2007)
24. Liu, Y.: A mean field annealing approach to accurate free form shape matching. *Pattern Recognition* (2007)
25. Makadia, A., Patterson, A., Daniilidis, K.: Fully automatic registration of 3D point clouds. In: *CVPR* (2006)

26. Mellado, N., Aiger, D., Mitra, N.J.: Super 4PCS: Fast global pointcloud registration via smart indexing. *Computer Graphics Forum* 33(5) (2014)
27. Mian, A.S., Bennamoun, M., Owens, R.: Three-dimensional model-based object recognition and segmentation in cluttered scenes. *PAMI* 28(10) (2006)
28. Mian, A.S., Bennamoun, M., Owens, R.A.: Automatic correspondence for 3D modeling: an extensive review. *International Journal of Shape Modeling* 11(2) (2005)
29. Papazov, C., Haddadin, S., Parusel, S., Krieger, K., Burschka, D.: Rigid 3D geometry matching for grasping of known objects in cluttered scenes. *International Journal of Robotics Research* 31(4) (2012)
30. Pomerleau, F., Colas, F., Siegwart, R., Magnenat, S.: Comparing ICP variants on real-world data sets – open-source library and experimental protocol. *Autonomous Robots* 34(3) (2013)
31. Raguram, R., Frahm, J., Pollefeys, M.: A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus. In: *ECCV* (2008)
32. Rangarajan, A., Chui, H., Mjolsness, E., Pappu, S., Davachi, L., Goldman-Rakic, P.S., Duncan, J.S.: A robust point-matching algorithm for autoradiograph alignment. *Medical Image Analysis* 1(4) (1997)
33. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: *3DIM* (2001)
34. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (FPFH) for 3D registration. In: *ICRA* (2009)
35. Salas-Moreno, R.F., Newcombe, R.A., Strasdat, H., Kelly, P.H.J., Davison, A.J.: SLAM++: Simultaneous localisation and mapping at the level of objects. In: *CVPR* (2013)
36. Salvi, J., Matabosch, C., Fofi, D., Forest, J.: A review of recent range image registration methods with accuracy evaluation. *Image and Vision Computing* 25(5) (2007)
37. Shin, J., Triebel, R., Siegwart, R.: Unsupervised discovery of repetitive objects. In: *ICRA* (2010)
38. Tam, G.K.L., Cheng, Z., Lai, Y., Langbein, F.C., Liu, Y., Marshall, D., Martin, R.R., Sun, X., Rosin, P.L.: Registration of 3D point clouds and meshes: A survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics* 19(7) (2013)
39. Theiler, P.W., Wegner, J.D., Schindler, K.: Globally consistent registration of terrestrial laser scans via graph optimization. *Journal of Photogrammetry and Remote Sensing* 109 (2015)
40. Tombari, F., Salti, S., di Stefano, L.: Performance evaluation of 3D keypoint detectors. *IJCV* 102(1-3) (2013)
41. Tsin, Y., Kanade, T.: A correlation-based approach to robust point set registration. In: *ECCV* (2004)
42. Yang, J., Li, H., Campbell, D., Jia, Y.: Go-ICP: A globally optimal solution to 3D ICP point-set registration. *PAMI* (2016), to appear