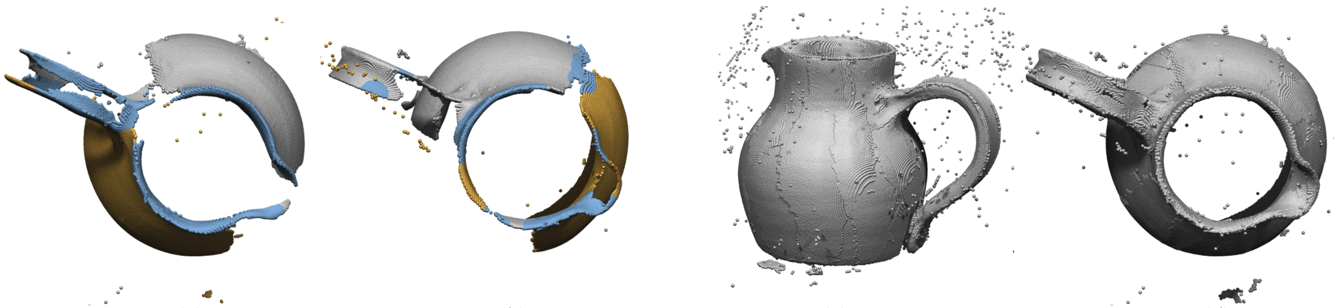
4-Points Congruent Sets for Robust Pairwise Surface Registration

Dror Aiger
Department of Computer Science,
Ben Gurion University

Niloy J. Mitra
IIT Delhi
TU Vienna

Daniel Cohen-Or
School of Computer Science,
Tel Aviv University



(a) (b) (c) (d)

Figure 1: Reconstruction from raw scans using 4-points congruent sets. Reconstruction results from nine input scans of a shinny water jug. Neighboring scans have 40% overlap or less, and required an average of 16 seconds for fully automatic alignment starting from *arbitrary initial poses*. Pairwise alignment results are robust even with low overlap. Typical pairwise alignments are shown in (a) and (b), where for visualization we roughly mark the overlap regions in blue. The final alignment result, (c) and (d), is obtained without any data smoothing, outlier removal, local ICP refinement, global error distribution, or any assumption about starting alignment.

Abstract

We introduce 4PCS, a fast and robust alignment scheme for 3D point sets that uses wide bases, which are known to be resilient to noise and outliers. The algorithm allows registering raw noisy data, possibly contaminated with outliers, without pre-filtering or denoising the data. Further, the method significantly reduces the number of trials required to establish a reliable registration between the underlying surfaces in the presence of noise, without any assumptions about starting alignment. Our method is based on a novel technique to extract all coplanar 4-points sets from a 3D point set that are approximately congruent, under rigid transformation, to a given set of coplanar 4-points. This extraction procedure runs in roughly $O(n^2 + k)$ time, where n is the number of candidate points and k is the number of reported 4-points sets. In practice, when noise level is low and there is sufficient overlap, using local descriptors the time complexity reduces to $O(n + k)$. We also propose an extension to handle similarity and affine transforms. Our technique achieves an order of magnitude asymptotic acceleration compared to common randomized alignment techniques. We demonstrate the robustness of our algorithm on several sets of multiple range scans with varying degree of noise, outliers, and extent of overlap.

Keywords: computational geometry, pairwise surface registration, scan alignment, partial shape matching, largest common pointset (LCP) measure, affine invariant ratio

---

ACM Reference Format
Aiger, D., Niloy, M., Cohen–Or, D. 2008. 4–Points Congruent Sets for Robust Pairwise Surface Registration. *ACM Trans. Graph. 27*, 3, Article 85 (August 2008), 10 pages. DOI = 10.1145/1360612.1360684 http://doi.acm.org/10.1145/1360612.1360684.

1 Introduction

Surface registration is the process of identifying and matching corresponding regions across multiple scans given in arbitrary initial positions, and estimating the corresponding rigid transforms that best align the scans to each other. In recent years, advances in geometry scanning technology have lead to a growing interest in surface acquisition techniques where multiple scans are required to be registered into a coherent coordinate frame using rigid transformations [Callieri et al. 2004, Gelfand et al. 2005, Li and Guskov 2005], or using non-rigid alignment [Pauly et al. 2005, Brown and Rusinkiewicz 2007]. The registration problem is an instance of *partial matching* of 3D point sets, a larger and more general fundamental problem in computational geometry and computer vision.

One popular philosophy behind registration techniques is to use robust local shape descriptors [Li and Guskov 2005] to define an approximate transformation, followed by an Iterated Closest Point (ICP) method [Besl and McKay 1992, Chen and Medioni 1992] to refine the solution. Rigid transformations, being low dimensional entities, can be uniquely recovered if the correct correspondence is known between just three point-pairs. Local descriptors, invariant under rigid transforms, are commonly used to extract such a small set of points with good candidate correspondences. We refer to such a set of points as a *base*.

Given two parts P and Q in arbitrary initial poses, matching pairs of bases, one from P and one from Q, generates a set of candidates aligning transformations between P and Q. A popular technique called geometric hashing can be used to pick a good aligning transformation [Wolfson and Rigoutsos 1997] from such a candidate set. Randomized algorithms, like RANSAC (RANdom SAmple Consensus) [Fischler and Bolles 1981], repeat the voting process for a number of base candidates enough times to ensure that it finds, with high probability, a good solution. Improvements of this basic paradigm have been proposed by Chen and co-authors [1999] with tradeoff between speed and robustness to noise.

Global quantities like object centroids and Principal Component Analysis (PCA) are often used to roughly align objects to transformed copies of the same. However, in case of partial overlap, such

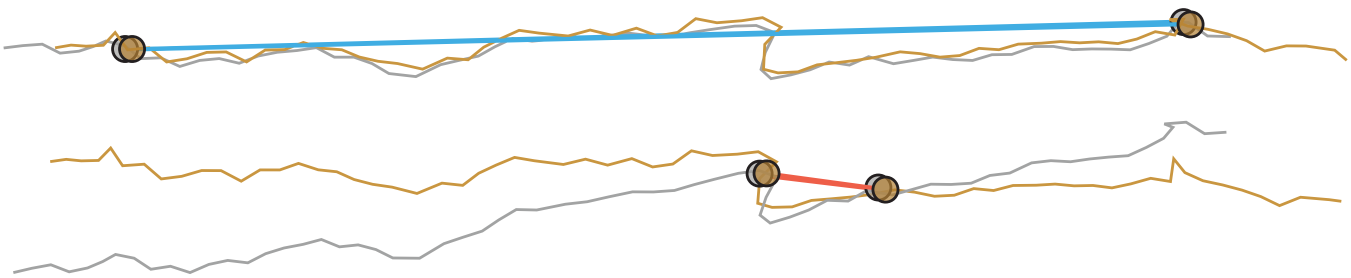


Figure 2: Stability of wide-bases. Alignment with wide-base (top) is more stable than alignment using narrow-base (bottom). The gray and the golden curves were generated from a common curve, but were perturbed using noise. For robustness we prefer the widest possible base [Goodrich et al. 1994], where the maximum width is limited by the extent of overlap between the shapes.

methods quickly break down. Local descriptors [Johnson 1997, Li and Guskov 2005], computed from local surface geometry and invariant to rigid transforms, have been used as an efficient tool for fast partial matching. Although under noisy conditions it is possible to robustly compute such local descriptors [Pottmann et al. 2007], in presence of significant noise and outliers, defining a reliable local descriptor still remains a challenging task. In such scenarios, instead of using local descriptors, an effective alternative is to rely on the principle of large numbers. This approach requires solving the Largest Common Pointset (LCP) problem: Given two point sets P and Q, LCP under δ-congruence solves for a subset P' of P, having the largest possible cardinality, such that the distance between $T(P')$ and Q is less than δ under an appropriate distance measure, T being a rigid transform.

Given 3D point sets P and Q with respective cardinality m and n, a naive alignment scheme has a time complexity of $O(m^3n^3)$: For each triplet of points or *base* from P, take a set of three points from Q, solve for the unique aligning rigid transformation using this correspondence, and evaluate the quality of the current correspondence by aligning all other points using the computed transform. A randomized version of this algorithm tries an appropriate number of random bases from P, thus reducing the complexity down to $O(mn^3 \log n)$ time [Irani and Raghavan 1996]. This complexity can be further reduced to $O(n^3 \log n)$ using randomized verification. However, for large point sets this is still overly expensive.

Contributions and Overview. In this paper, we introduce an alignment scheme for 3D point sets to automatically register a pair of surfaces even with rather small overlap. The method makes no assumption about their starting positions, and brings the two pieces into a good alignment, which can further be refined using ICP algorithm (cf. [Rusinkiewicz and Levoy 2001]). Our method is based on a novel technique to extract all sets of coplanar 4-points from a 3D point set that are approximately congruent i.e., related by rigid transforms, to a given planar 4-points in roughly $O(n^2 + k)$ time, where n is the number of points in Q and k is the number of reported 4-points sets. The algorithm uses the following fact: Certain ratios defined on a planar congruent set remain invariant under affine transformations, and hence under rigid motion. Additionally, when simple reliable local descriptors can be computed, the alignment procedure runs in $O(n + k)$. We also show how to handle similarity and affine transformations using an extension of our algorithm.

A key point of our approach is to use a *wide-base* invariant to quickly and reliably determine rigid transformation in 3D, as shown in Figure 2. Constructed using well spaced samples, wide-bases as opposed to local ones, are in general more stable, and hence desirable. Specifically Goodrich and colleagues [1994] provide bounds on approximation level in term of the diameters of the point sets undergoing rigid transformations in the plane. However, for partial matching, the width of the base is restricted by the extent of overlap.

The combination of a wide-base and the LCP measure make our registration method resilient to noise and outliers, thus allowing direct pairwise registration of raw scanned data. This is advantageous since the reconstruction process accounts for all the sampled data avoiding any early removal, filtering, or other mollification of the scanned data (see Figure 3). Our algorithm makes no assumption about the initial alignment of the given point sets, works directly on data corrupted with noise and outliers, and successfully aligns point

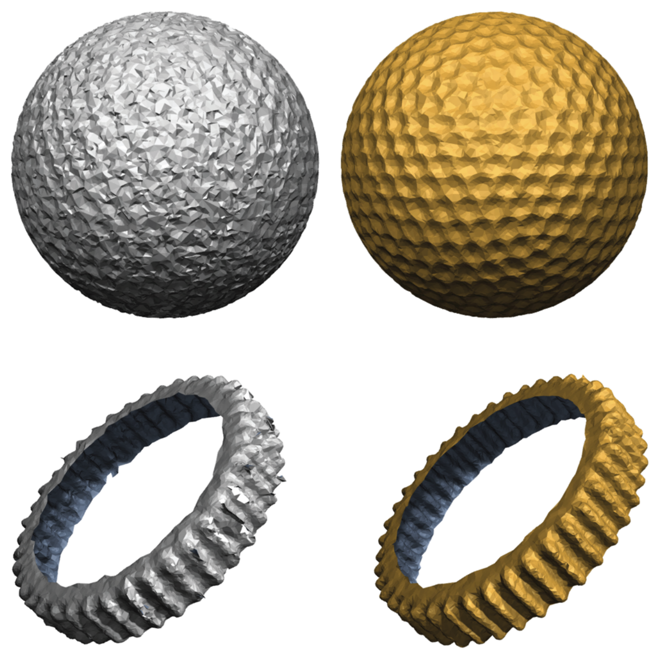


Figure 3: Advantage of directly registering raw noisy data. (Left) Denoising the original scans before registration can be harmful: Given two scans P and Q, we pre-smooth them, use the smoothed versions to compute local descriptors to establish correspondence [Gelfand et al. 2005] and compute an aligning transform. We use this transform to align the original dataset Q to P, and finally smooth the combined models. (Right) Directly aligning the noisy data using 4PCS, and then smoothing the result yields a higher quality surface. In both cases, the same MLS operator is used for surface smoothing. Further reduction of noise from the left column models results in significant loss of high frequency features.

sets even with small overlap (see Figure 1). Experiments on real scans indicate that our method outperforms state-of-the-art algorithms for pairwise registration of surfaces in the presence of noise and outliers.

After reviewing previous work and background in Section 2, we describe the concept of approximate congruent 4-points in Section 3. In Section 4, we present the registration algorithm. We tested the proposed method on a wide variety of range scans obtained using different acquisition devices. In Section 5, we report the algorithm's running time and its resiliency to noise, outliers, and varying extent of overlap.

2 Background

Partial matching is a fundamental task for registration and recognition of 3D objects [Frome et al. 2004]. Typically, the alignment of parts is based on matching local descriptors, such as spin images [Johnson 1997], shape context [Mori et al. 2005], or integral invariants [Pottmann et al. 2007]. Such methods have been adapted to accelerate the registration of 3D scans acquired by range scanners [Li and Guskov 2005].

In the computer vision community, there has been extensive work on pattern matching to seek resemblance between point sets in 2D. Famous algorithms for this problem include Hough transform [Ballard 1987], RANSAC [Fischler and Bolles 1981], image based alignment [Huttenlocher and Ullman 1990], and geometric hashing [Wolfson and Rigoutsos 1997]. These methods are based on the principle of *generate and test*. Hashing based approaches have also been used for shape registration and retrieval from large collection of models [Germain et al. 1997, Gal and Cohen-Or 2006, Mitra et al. 2006].

In this paradigm, one picks a base from P where the number of points in the base is the minimum number required to uniquely define a transformation. Now for each random choice of base from Q, the corresponding aligning transform is verified [Fischler and Bolles 1981, Huttenlocher and Ullman 1990], or voted for in a suitable transformation space [Ballard 1987]. We now briefly describe the basic RANSAC algorithm and related randomized alignment techniques, since our method is motivated by similar principles.

First, let us define the problem: Given two point sets P and Q in arbitrary initial positions, find the best transformation from a prescribed family of transformations, typically rigid transformations, that best aligns regions of P and Q. By *best fit*, we refer to the transformation that brings the maximum number of points from P to within some δ-distance of points in Q (see Figure 10). In the case of rigid motion, a base size of three points is sufficient to uniquely determine the aligning transform.

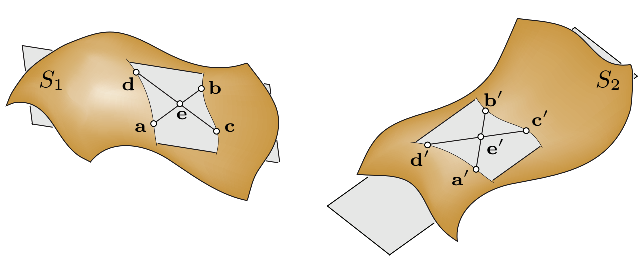


Figure 4: Affine invariant ratio for congruent 4-points. Given points from a surface S_1, let points $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}$ be coplanar, and the lines $\mathbf{ab}$ and $\mathbf{cd}$ meet at point $\mathbf{e}$. The ratios $r_1 = \|\mathbf{a}-\mathbf{e}\|/\|\mathbf{a}-\mathbf{b}\|$ and $r_2 = \|\mathbf{c}-\mathbf{e}\|/\|\mathbf{c}-\mathbf{d}\|$ are preserved under any affine transform. If S_2 is another surface which (partially) matches S_1 and the 4-points coplanar base lies in the overlap region, then the set of corresponding points from S_2 are coplanar, and satisfy the following relations: $\|\mathbf{a}'-\mathbf{e}'\|/\|\mathbf{a}'-\mathbf{b}'\| = r_1$ and $\|\mathbf{c}'-\mathbf{e}'\|/\|\mathbf{c}'-\mathbf{d}'\| = r_2$.

RANSAC. The RANSAC algorithm [Fischler and Bolles 1981] is a widely used general technique for robust fitting of models to data corrupted with noise and outliers. The RANSAC based alignment procedure is simple: Randomly select three different points from P and three from Q to form a pair of bases in correspondence, compute the candidate transformation T_i that aligns the base pairs, and then count the number of points k_i from P that within δ-distance from points in Q. If k_i is sufficiently large, accept T_i as a good solution. Otherwise, the process is repeated by randomly selecting another triplet of points thus deriving different candidate transformations that may improve the current best fit.

The process of selecting base-points randomly from P and Q is repeated L times, and the best solution i.e., the solution with the highest k_i is selected. Depending on the percentage of the data that belongs to the structure being fit (assuming there is only one structure to fit), and the desired success probability, L can be chosen accordingly [Fischler and Bolles 1981].

Randomized Alignment. Irani and Raghavan [1996] proposed a variant of the RANSAC algorithm to define a randomized version of alignment in 2D under similarity transformations. The procedure randomly picks a base from P, computes transformations that align the base to all possible bases from Q, and verifies the resulting registration. As in RANSAC, to achieve a certain probability of success this procedure is repeated for L different choices of bases from P. Further, the verification stage is also randomized: First only a constant number of random points in P are verified, and only if a significant fraction of this subset is well matched, the remaining points are tested. Unlike basic RANSAC, we pick base points randomly from P, and look for corresponding points in Q using some registration algorithm – we propose 4PCS for this step.

Let p_g be the probability that a randomly selected point from P is also present in Q (i.e., is in the overlap region), the size of the base is N, and p_f be the probability that the algorithm exits after L tries *failing* to find a good fit that exists. Since we choose points from P at random, simple reasoning gives us the relation $p_f = (1 - p_g^N)^L$. Thus for a success probability more than p_s we need,

$$L > \log(1 - p_s)/\log(1 - p_g^N) \quad (1)$$

iterations. For rigid transforms, it is sufficient to have $N = 3$.

Our algorithm builds on this randomized alignment approach. However, instead of exhaustively testing all possible bases from Q which quickly becomes infeasible in 3D, we introduce the idea of *planar congruent sets* to select only a small set of bases from Q that can potentially match a given base from P. Next we explain how this is made possible by a special ratio property that is preserved for planar congruent 4-points sets under affine transformations.

3 Approximate Congruent 4-Points

To align two point sets P and Q in arbitrary initial positions, we follow the general alignment approach described in the previous section. For different choices of base pairs from P and Q, we compute the corresponding rigid transformations, evaluate their quality, and choose the best transformation. A pair of triplets, one from P and one from Q, is enough to uniquely define a rigid transformation. For a base from P, randomly select a 3-points base from Q. Naively, there are $O(n^3)$ such candidate triplets from Q, where n is the number of points in Q.

Surprisingly the problem becomes easier if we look for special 4-points bases, instead of 3-points ones. Our approach is to use a set of *4 coplanar* points from P as base B, and to quickly find all subsets of 4-points from Q that are approximately congruent to B. Later we elaborate how to efficiently extract such a set B from P. By approximate *congruence*, we mean that the two 4-points sets can be aligned, up to some allowed tolerance δ, using rigid transformation. We will detail how to extract all such congruent subsets

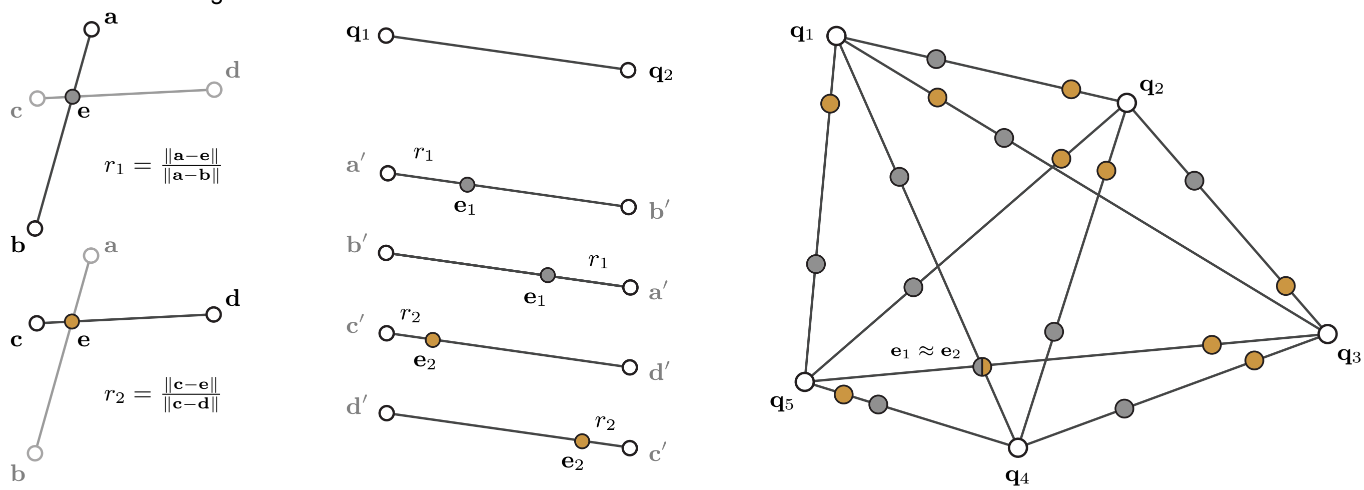


Figure 5: Extracting affine invariant congruent 4-points. (Left) Given a base $B \equiv \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ consisting of four (approximately) coplanar points, we extract the two ratios r_1 and r_2. (Middle) For any point-pair $\{\mathbf{q}_1, \mathbf{q}_2\}$, there can be two assignments corresponding to $\{\mathbf{a}, \mathbf{b}\}$, and another two assignments corresponding to $\{\mathbf{c}, \mathbf{d}\}$ leading to 4 possible *intermediate points*. These points are computed as $\mathbf{e}_1 = \mathbf{q}_1 + r_1(\mathbf{q}_2 - \mathbf{q}_1)$ and $\mathbf{e}_2 = \mathbf{q}_1 + r_2(\mathbf{q}_2 - \mathbf{q}_1)$. (Right) Now, given a set of coplanar points Q, we want to extract a 4-points set which is congruent to the given base B up to affine transforms. For each pair of points $\{\mathbf{q}_1, \mathbf{q}_2\} \in Q$, we compute four intermediate points as described. For simplicity, we just indicate two points per point-pair in the figure. A set of 4 points is approximately congruent to given B, if $\mathbf{e}_1 \approx \mathbf{e}_2$. In this example, $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ is approximately congruent to $\{\mathbf{q}_5, \mathbf{q}_3, \mathbf{q}_4, \mathbf{q}_1\}$.

of Q in roughly $O(n^2 + k)$ runtime, k being the number of such congruent sets in Q. In practice k is small, leading to a significant acceleration with respect to a naive alignment technique which requires $O(n^3)$ time, or one of the best known algorithms [Indyk et al. 1999] which requires $O(n^{2.25}\sqrt{D})$, D being the diameter of the base from P. Chen and colleagues [1999] also presented a strategy for selecting good candidates for Q.

3.1 Overview

Affine transformations exhibit the following property: Given three *collinear* points $\{\mathbf{a}, \mathbf{b}, \mathbf{c}\}$, the ratio $\|\mathbf{a} - \mathbf{b}\|/\|\mathbf{a} - \mathbf{c}\|$ is preserved. Huttenlocher [1991] used this invariant to extract all sets of $2D$ affine invariants of 4-points in the plane that are equivalent under affine transforms. We take a similar approach in $\mathbb{R}^3$. Given a coplanar 4-points base set, we look for other (approximately) affine equivalent 4-points sets in the given point cloud data. The set of all affine-invariant 4-points is a *superset* of the 4-points congruent points in $\mathbb{R}^3$. Subsequently we verify whether such 4-points sets are (approximately) congruent to the chosen base set. In practice, using a distance constraint for rigid transforms this superset can be made quite conservative. First we briefly describe the 2D method of extracting affine invariant 4-points set, and then detail the extraction procedure in 3D.

3.2 Affine Invariants of 4-Points Sets

Huttenlocher [1991] introduced a method to extract the set of 4-points which are affine-invariant in 2D. The strength of the method is that it only requires to examine all pairs of points, leading to a $O(n^2)$ method, excluding the time to report the sets. A set of coplanar points $X \equiv \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$, not all collinear, defines two independent ratios of three collinear points. Let $\mathbf{ab}$ and $\mathbf{cd}$ be the two lines that intersect at an *intermediate* point $\mathbf{e}$. Note that it is always possible to choose the pairs such that the lines intersect. The two ratios,

$$\begin{aligned} r_1 &= \|\mathbf{a} - \mathbf{e}\|/\|\mathbf{a} - \mathbf{b}\| \\ r_2 &= \|\mathbf{c} - \mathbf{e}\|/\|\mathbf{c} - \mathbf{d}\| \end{aligned} \qquad (2)$$

are invariant under affine transformation, and uniquely define 4-points up to affine transformations (see Figure 4). Now given a set Q of n points, and two affine invariant ratios r_1 and r_2, we can efficiently extract all 4-points sets that are defined by these two invariants in $O(n^2 + k)$ time, where k is the number of reported 4-points sets, as follows: For each pair of points $\mathbf{q}_1, \mathbf{q}_2 \in Q$, compute two *intermediate* points:

$$\begin{aligned} \mathbf{e}_1 &= \mathbf{q}_1 + r_1(\mathbf{q}_2 - \mathbf{q}_1) \\ \mathbf{e}_2 &= \mathbf{q}_1 + r_2(\mathbf{q}_2 - \mathbf{q}_1). \end{aligned} \qquad (3)$$

Any two pairs of points whose intermediate points, one resulting from r_1 and one from r_2, are coincident, probably correspond to a 4-points set that is an affine transformed copy of X (see Figure 5). Since the points $\mathbf{e}_1$-s and $\mathbf{e}_2$-s are all in the same coordinate system, it is possible to quickly search for coincident points using a neighborhood search structure.

In practice, due to noise intermediate points instead of being exactly coincident end up being nearby points. To allow quick range queries of proximity points we use a standard data structure in $\mathbb{R}^3$. We use an approximate *range tree* [Arya et al. 1998], which can be built in $O(n \log n)$ for a point set of size n, and supports querying of all points inside any rectangle in $O(\log n + k)$ time, k being the number of points to be reported. Once all the intermediate points are inserted into the range tree, we query for all points associated with r_1 that are in the δ-neighborhood of points associated with r_2. Calculating all the intermediate points takes $O(n^2)$ time, building and querying the neighbors takes $O(n^2 \log n + k)$, where k is the total number of points reported. Later we further improve this complexity.

3.3 Extracting Congruent 4-points in 3D

Given a 4-points base B of (approximately) coplanar points selected from a point set P and another point set $Q \in \mathbb{R}^3$, our goal is to extract the set of all 4-points from Q that are approximately *congruent* to B, up to an approximation level δ. First given B we compute its two affine invariants ratios over this plane, as described previously. Then from the points Q, we extract all point sets that can be related to B by affine transforms, using the method described

Algorithm 1 (4PCS) Given two point sets P and Q in arbitrary initial positions, with high probability, compute the best rigid alignment according to the LCP (Largest Common Pointset) measure within an approximation level $\delta > 0$.

```
h ← 0
for i = 1 to L do                                        ▷ RANSAC loop
    B ← SelectCoplanarBase (P)
    U ← FindCongruent (B, Q, δ)
    for all 4-points coplanar sets U_i ∈ U do
        T_i ← best rigid transform that aligns B to U_i in the least
square sense [Horn 1987].
        Find S_i ⊆ P, such that d(T_i(S_i), Q) ≤ δ
    end for
    k ← arg max_i {|S_i|}
    if |S_k| > h then
        h ← |S_k|
        T_opt ← T_k
    end if
end for
    return T_opt
```

```
procedure FindCongruent (B ≡ {b_1, b_2, b_3, b_4}, Q, δ)
d_1 ← ||b_1 − b_2||
d_2 ← ||b_3 − b_4||
Compute R_1 ≡ {(p_i, p_j) | p_i, p_j ∈ Q}, such that ||p_i − p_j|| ∈
[d_1 − δ, d_1 + δ].
Compute R_2 ≡ {(p_i, p_j) | p_i, p_j ∈ Q}, such that ||p_i − p_j|| ∈
[d_2 − δ, d_2 + δ].
for all pair r_1i ∈ R_1 do
    Compute the 4 points, {e^1_1i, e^2_1i, e^3_1i, e^4_1i} related to the in-
variants r_1 and r_2 (see Figure 5). Let Π denote the originating
point-pair for such intermediate points i.e., Π(e^j_1i) = r_1i.
end for
Build an approximate range tree structure (RS) in ℝ^3 for the point
set {e^j_1i} [Arya et al. 1998].
for all pair r_2i ∈ R_2 do
    Compute the 4 points, {e^1_2i, e^2_2i, e^3_2i, e^4_2i} related to the in-
variants r_1 and r_2. Again Π(e^j_2i) = r_2i.
end for
U' ← ∅.
for all e^j_2i do
    Using RS, retrieve all points in δ-neighborhood of query
point e^j_2i. For each such point q, create candidate 4-points sets
corresponding to B as U' ← {U', (Π(q), Π(e^j_2i))}.
end for
U ← all 4-points sets in U' that are approximately congruent to
B in ℝ^3.
    return R̃
```

in Section 3.2. Though this method generates a superset of the desired 4-points, for rigid transforms we get only a limited number of spurious matches.

In order to remove the non-congruent bases, we look at their original positions in $\mathbb{R}^3$, and verify whether the corresponding sets agree within some threshold to the base set B, up to rigid transformations. Then using base B and each of the potential bases from Q, we compute the best aligning rigid transformation in the least square sense using a closed form solution as proposed by Horn [1987]. For a wide base the typical number of such reported subsets is $O(n)$ with a small constant, since the distance between the base points is close to the diameter of Q. In such a scenario, we can extract all the congruent 4-points sets from point sets of size n in $O(n^2)$. However, with no approximation ($\delta = 0$), the number of reported subsets is upper bounded by $O(n^{5/3})$ [Agarwal and Sharir 2002]. Later we extend our algorithm to handle affine transformations. In Section 5, we report the performance of our technique on a variety of range scans.

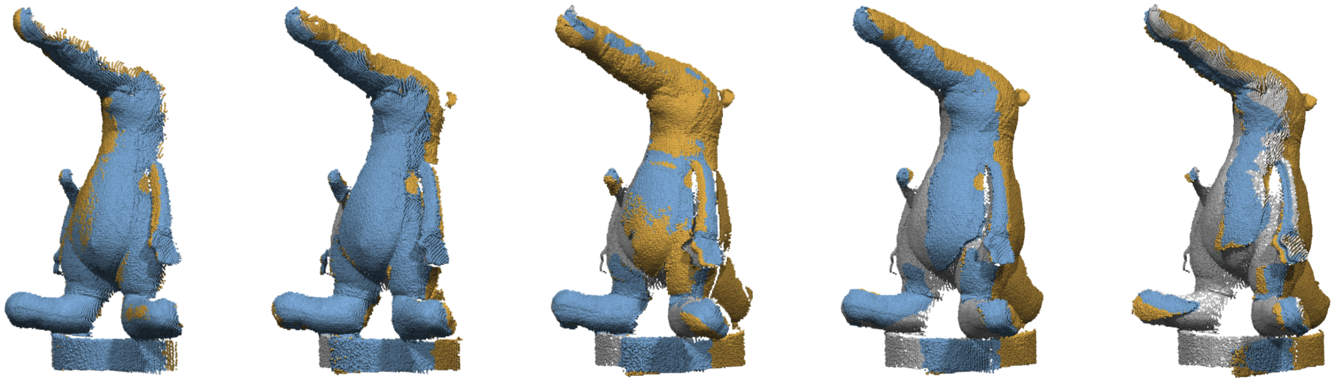
The above procedure requires computation and storing of $O(n^2)$ intermediate points, which is prohibitive for large point sets. However, when looking for a rigid alignment, a conservative subset of size $O(n)$ is sufficient because of the following: Rigid transforms preserve inter-point Euclidean distance. Given a base $B \equiv \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$, we first compute distances $d_1 = \|\mathbf{a} - \mathbf{b}\|$ and $d_2 = \|\mathbf{c} - \mathbf{d}\|$. Now we only consider point pairs from Q which are d_1 or d_2 apart, up to a tolerance δ. For point sets which have roughly uniform sampling, we only need to insert $O(n)$ pairs into our range data structure. Hence the full algorithm runs in $O(n^2)$ including the time for range tree construction. Considering only a linear number of such pairs, instead of a quadratic number of them, leads to significant speedup and a manageable space, enabling rigid alignment even for large point sets.

4 The 4PCS Algorithm

We are given two point sets P and Q, some uncertainty measure ($\delta > 0$) for the positional accuracy of the points, and an estimate of the fraction f of the points in P that can be matched to Q. Our goal is to find a rigid transform that brings maximum number of points from P to distance less than δ from some point in Q. We propose an output sensitive algorithm (see Algorithm 1) whose runtime depends on the maximum number of matching points between the given point sets, and being randomized, discovers the correct solution with high probability.

We first select a base $B \subseteq P$ consisting of 4-coplanar points. In practice, we allow some non-planarity since it is unlikely that there exists 4-coplanar points. We pick three points at random, and select the remaining point such that the four points together form a wide base which is (approximately) coplanar. A *wide base* created by selecting points that are far from each other results in more stable alignments [Goodrich et al. 1994]. However, if we choose the points too far apart, the selected points may not all lie in the overlap area (for partial matching), and hence the desired solution may be missed. We use the overlap fraction f to estimate this maximum distance. If an estimate of f is not provided, we run the algorithm with decreasing guesses as $f = 1, 0.5, 0.25, \ldots$ until we achieve the desired error tolerance. For applications where additional information is available about possible regions of overlap, such knowledge can be used to make a more informed choice of base B. We first select a set of wide three points that are likely to be in the overlap region, and then choose the fourth one as described before.

For a planar base B obtained from the SelectCoplanarBase stage of our algorithm, we can define affine invariant ratios using the base points. However, for approximately planar base, we use the closest points between the lines joining suitable point pairs for defining invariant ratios. Now, we apply the method described in Section 3.3 to extract the set $U \equiv \{U_1, U_2, .., U_s\}$ of all subsets of 4-points from Q that are potentially congruent to B up to an approximation level δ. For each U_i, using the correspondence information between B and U_i, we compute the best aligning rigid transform T_i that brings B close to U_i in a least square sense [Horn 1987]. To verify T_i, we compute $T_i(P)$ and find how many points in $T_i(P)$ are closer than δ to some point in Q. This verification is performed in a randomization fashion. For efficiency, we use ANN (Approximate Nearest Neighbor) [Arya et al. 1998] for neighborhood query in $\mathbb{R}^3$. We first select a constant number of points from P, transform the points using T_i, and for each such point query for close neighbors in Q. If enough points are matched, we perform similar tests for the remaining points in P, and assign a score for T_i



80% 70% 60% 50% 40%

Figure 6: Partial matching with varying overlap. We align scan pairs of the Coati model over varying amount of overlap, from 80% down to 40%. Overlap is being measured with respect to surface area of the smaller input model. The results are shown *without* any ICP refinement to demonstrate how the algorithm degrades (see Figure 8 for corresponding estimation error). These results when refined by a couple ICP iterations, all lead to near perfect alignment. For visualization, we show the overlapping regions in blue, while gray and golden, respectively, denote the remaining parts of the input models. Overlap regions were identified by manually aligning the models followed by ICP refinement.

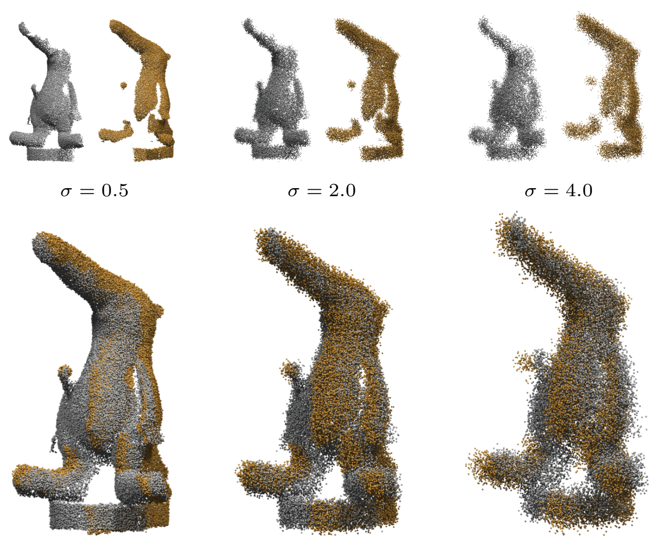
(cf. [Irani and Raghavan 1996]). Let T denote the transformation with the best score.

Given a base B_i, we described how to compute the best transformation T_i corresponding to it. Each aligning transform T_i is assigned a score based on the number of points that are brought into alignment up to a threshold δ. Using this procedure we now perform randomized alignment with RANSAC (see Section 2), and test out L different bases (see Equation 1) depending on the estimate of overlap fraction f. Over all such trials, we select the transform T_{opt} with the best score.

Runtime. Let point sets P and Q have m and n points, respectively. Selecting a random coplanar 4-points base $B \subseteq P$ takes $O(m)$. In the most expensive phase of the algorithm, all point pairs from Q with distance d_1 or d_2 (Section 3.3) are extracted in $O(n^2+k_1)$, where k_1 is the number of reported pairs. Subsequently all the matched bases from Q can be extracted in $O(k_1 + k_2)$, k_2 being the number of 4-points sets in Q affine invariant to B. When the overlap margin is not arbitrarily small, a randomized verification takes $O(k_3)$ where k_3 is the number of actual 4-points bases from Q that are approximately congruent to B, assuming that P does not have many copies in Q. Worst case bounds for such k_i-s is a topic of interest in combinatorial geometry, and such bounds exist for both the exact ($\delta = 0$) and approximate cases [Indyk et al. 1999, Agarwal and Sharir 2002]. However, for point sets with samples approximately evenly distributed on the scanned surface, all k_i-s are $O(n)$. If the overlap is more than a constant fraction of m, we need to try only a constant number of random bases B. Hence the proposed algorithm runs in $O(n^2)$ requiring $O(n)$ space.

Next we describe how local descriptors, if they can be reliably computed, lead to improved efficiency without compromising accuracy of the result.

Further Enhancements. In the context of surface registration, local descriptors, invariant under rigid transforms, are often used to reduce the search space for aligning objects in arbitrary initial poses [Johnson 1997, Gelfand et al. 2005, Li and Guskov 2005, Pottmann et al. 2007]. Similarly, our algorithm benefits from local descriptors when they can be robustly computed. In the following, we show that even using a simple *non-invariant* descriptor, such as surface normal, significant speedup is possible, while still preserving the accuracy of computed transforms using wide bases. Given a base B of 4-coplanar points and a set $Q \in \mathbb{R}^3$, we want to find all 4-points subsets of Q that are approximately congruent up to rigid transforms to B, within an approximation level δ. Let $B \equiv \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ be four coplanar points in P (chosen as in Section 4), and $O \equiv \{N_{\mathbf{a}}, N_{\mathbf{b}}, N_{\mathbf{c}}, N_{\mathbf{d}}\}$ be their associated normals, with $d_1 = \|\mathbf{a} - \mathbf{b}\|$ and $d_2 = \|\mathbf{c} - \mathbf{d}\|$. Let $N(\mathbf{x}, \mathbf{y})$ denote the dihedral angle between the two local normal lines at points $\mathbf{x}$ and $\mathbf{y}$, respectively. Notice we do not require global consistency for normal orientations.



$\sigma = 0.5$ $\sigma = 2.0$ $\sigma = 4.0$

Figure 7: Partial registration with noisy data. (Top row) Partial scans of the Coati model under zero-mean additive Gaussian noise with variance σ^2. One unit roughly equals 1% of the bounding box diagonal. (Bottom row) Alignment results using 4PCS algorithm. All results were obtained without any ICP refinement, or assumptions about starting orientation of the input models. For noisy data, specially when sampling density is low, it is better to perform smoothing only *after* alignment as shown in Figure 3.

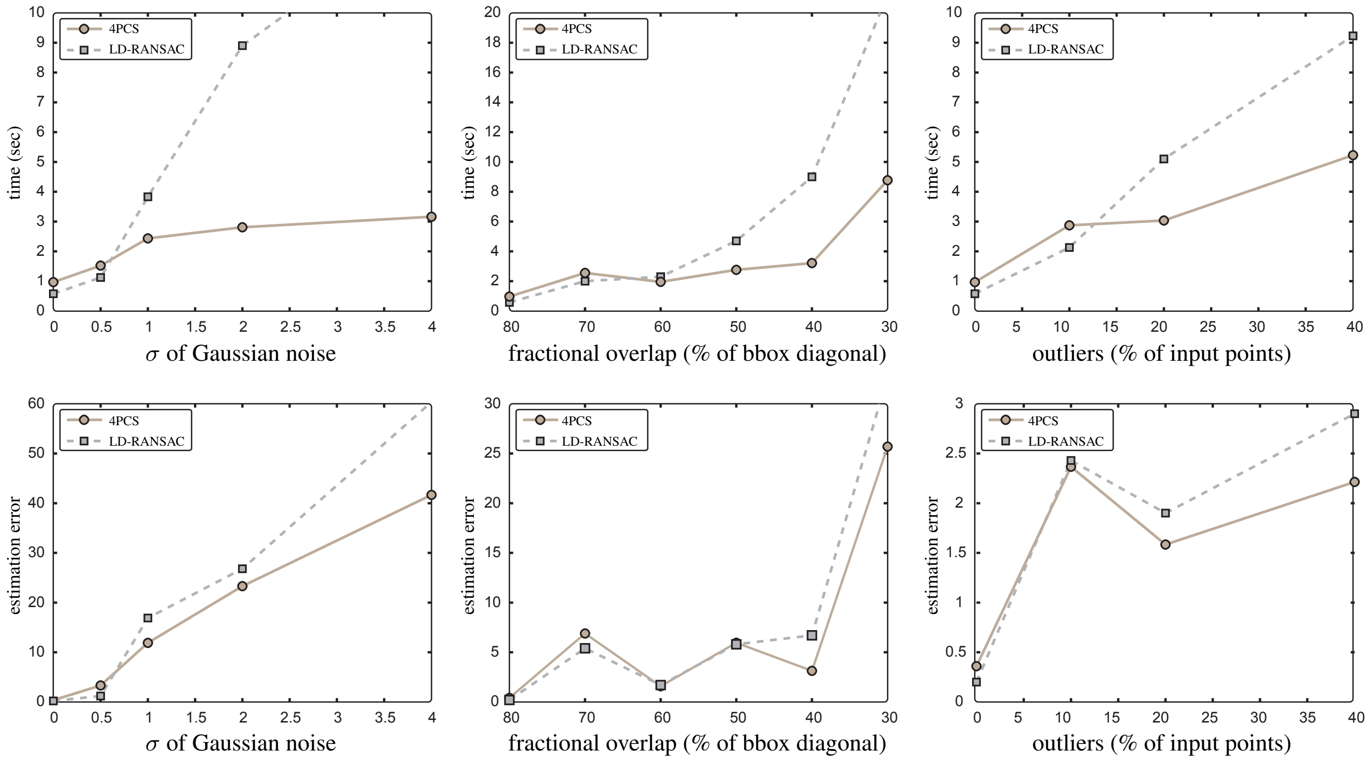


Figure 8: Performance and comparison. We compare the performance of our algorithm with a local descriptor based RANSAC algorithm. We use a combination of spin-image based descriptor [Li and Guskov 2005] and integral invariants [Pottmann et al. 2007] as robust local descriptors (LD-RANSAC). Parameters for local descriptors are manually adjusted to achieve good performance. Number of RANSAC steps for local method are chosen such that the error values (measured respect to ground truth) are comparable to those of our method. We observe that under low overlap, large noise, or high outliers, our algorithm outperforms LD-RANSAC method even when its parameters are manually tuned for best results. Corresponding alignments by our algorithm can be seen in Figure 6, 7, and 9, respectively. Estimation error is measured using standard RMS error between the final surface pairs. Bounding box diagonal length is taken as 100 units.

Equipped with approximate normals, all point pairs $\{\mathbf{q}_1, \mathbf{q}_2\} \in Q$ satisfying either $\|\mathbf{q}_1 - \mathbf{q}_2\| \approx d_1$, or $\|\mathbf{q}_1 - \mathbf{q}_2\| \approx d_2$ can be further pruned by the following computation: Select an arbitrary point $\mathbf{q} \in Q$, compute its approximate normal $N_{\mathbf{q}}$, and then extract all points $\mathbf{q}_i \in Q$, such that either $N(\mathbf{q}, \mathbf{q}_i) \approx N(\mathbf{a}, \mathbf{b})$, or $N(\mathbf{q}, \mathbf{q}_i) \approx N(\mathbf{c}, \mathbf{d})$. Using a neighborhood data structure over approximate normals (represented as points on the Gauss sphere) the above computation is done efficiently. Thus instead of storing all n^2 pairs, in practice, such normal and distance based constraints allow us to work with space linear in the number of input points.

Even using undirected approximate surface normals, which vary under rigid transforms, we observe a speedup by a factor of two. Further speedup is achieved when reliable rigid-invariant descriptors are available.

5 Results

We tested our 4PCS algorithm on a variety of input data with varying amount of noise, outliers, and extent of overlap. We now report performance regarding both the accuracy and robustness. For comparison, we describe how a local descriptor based RANSAC algorithm performs on the same inputs.

We aligned scans of the Coati model under various conditions: Figure 7 shows the robustness of the registration procedure under various amount of zero-mean Gaussian noise without any ICP refinement. In Figure 9 we show the alignment results in presence of varying amount of outliers without ICP refinement. In a related experiment, we varied the amount of overlap to evaluate performance degradation (Figure 6). Observe that varying the amount of outliers or the extent of overlap has similar effects on the run time of 4PCS since both results in fewer congruent bases between the two objects.

In Figure 8 we list the various performance characteristics of the algorithm. For comparison we used a RANSAC based algorithm with state-of-the-art local descriptors. We used a combination of multi-scale spin images [Li and Guskov 2005], and robust inte-

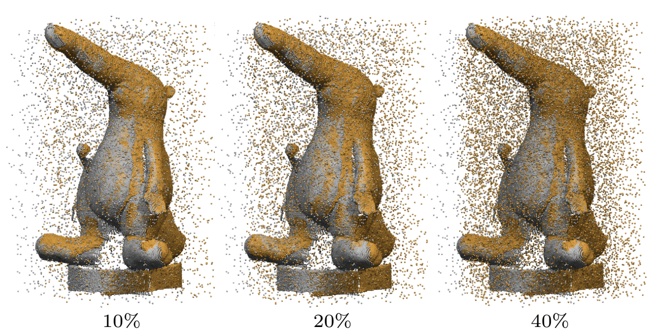


Figure 9: Partial matching with outliers. Given input scans of the Coati model in arbitrary initial poses, corrupted with outliers, the registered scans are shown. The number of outliers as percentage of the original number of input points are indicated in the corresponding figures. Reliable local descriptors such as spin images or integral invariants are very challenging to compute on such data without pre-processing the input scans.

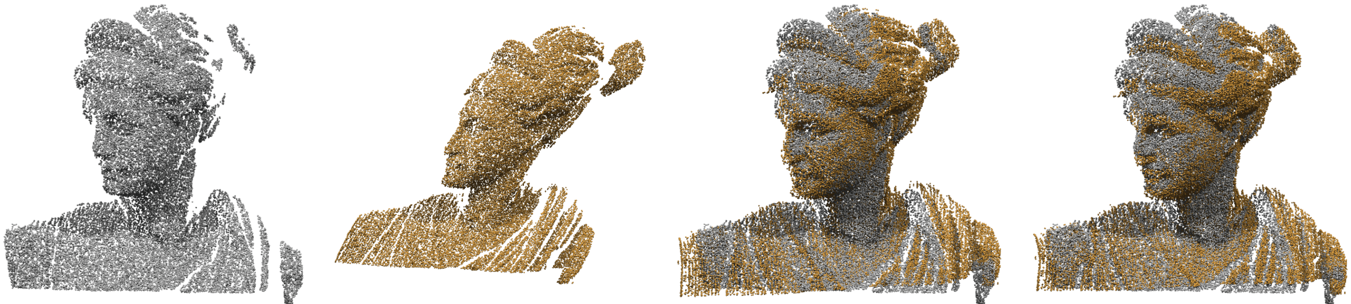


Figure 10: Affine surface registration using 4-points congruent set. Given two point sets in arbitrary initial poses, we solve for the best aligning affine transform according to the LCP measure. The alignment produced by the affine version of 4PCS method is refined with a few steps of a TPS (Thin Plate Spline) based refinement (see [Brown and Rusinkiewicz 2007]) to produce the final alignment. In this example, the initial models differed by a scale factor of 1.2 and a strong shear component. The final result is irrespective of the starting model poses.

gral invariants [Pottmann et al. 2007] as local descriptors. With increasing amount of noise and outliers, the RANSAC based approach slows down as the local descriptors become less reliable, and finally degenerates into a brute force search. In the local signature space, we manually selected minimum search radius such that the estimation error value remains comparable to that of our proposed algorithm. Under extreme conditions, local descriptor based results were too far for ICP refinement to converge to correct solution [Mitra et al. 2004], while 4PCS still performed satisfactorily. To summarize, our algorithm always outperforms LD-RANSAC – in presence of high amount of outliers, low overlap, or high noise. In scenarios when local descriptors can be reliably computed, 4PCS works even faster. Notice that LD and 4PCS address complementary issues arising during registration.

For noisy scans when local descriptor based methods fail, an alternate pipeline is as follows: denoise the inputs, align the smoothed scans, use the computed transform to align and merge the original noisy inputs, and finally smooth the merged scans. However, in some cases this turns out to be sub-optimal as crucial information gets lost in the process. Our algorithm has the advantage of working directly on the raw data as shown in Figure 3. This scenario is specially relevant when the sampling density is small, and the amount of redundancy in data is low.

In Figure 1, we demonstrate the stability of the algorithm under low extent of overlap. The final reconstruction is from nine scans. The result is obtained without any outlier removal, denoising, global error distribution, or ICP refinement. In Figures 11 and 12, we show how 4PCS gives near perfect alignment even when the scans are very flat, featureless, and noisy – a difficult class of alignment inputs for many local descriptor based methods. For large dense data sets it is sufficient to use a small fraction of the points, sampled uniformly, for computing the alignment, and the full data set is only used for verification. For the case of Jerusalem and building facade examples we used 0.05 as this fraction. Since we use LCP measure, our method is resilient to such uniform sampling. Residual alignment error is easily removed by a few ICP iterations.

In Table 1, we report the performance of 4PCS on various test datasets. As shown in the comparison to local descriptor methods

model	# points (in 1000)	4PCS (in sec)	4PCS w/normals (in sec)
water jug	450	159.5	78.4
Coati	54	2.3	1.7
mug	146	9	6.4
bust	70	4.1	2.4
Jerusalem	593	6.4	1.1
building facade	400	20.6	15.7

Table 1: Time taken by 4PCS for aligning various input sets, without and with normal information, as measured on a 1.8GHz Pentium M laptop with 1GB RAM. In all the examples, the models start in arbitrary positions, preprocessing time if any are included, and the reported times are averages over a few runs of the algorithm.

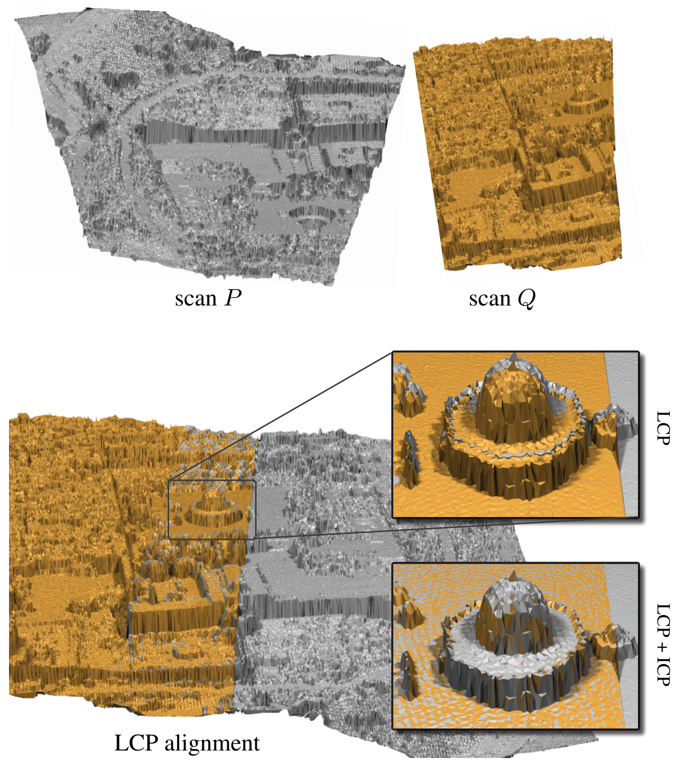


Figure 11: Aligning aerial scans of the old city of Jerusalem. Given two aerial scans P and Q, in arbitrary initial poses, we align them using 4PCS algorithm. Use of wide base for alignment results in stable alignment even for such flat aerial scans (see Figure 2). The small overlap between the scans makes this a challenging example. In the zoom-inset, we show the improvement in alignment after three steps of ICP refinement, a step orthogonal to our algorithm.

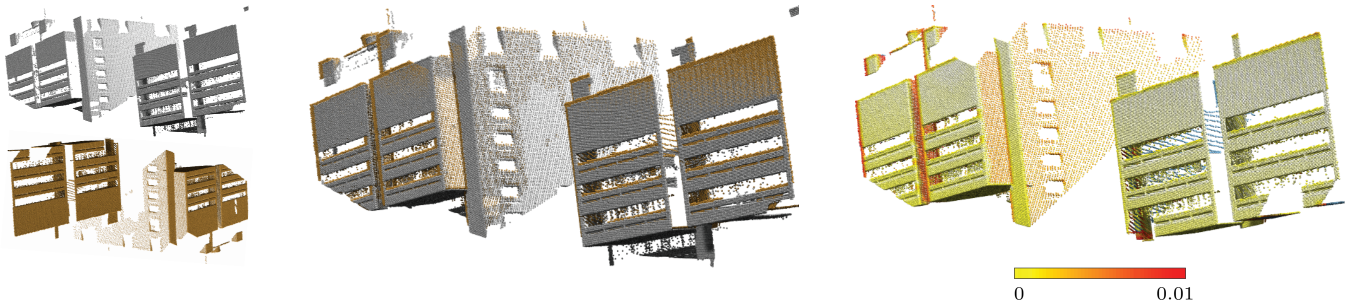


Figure 12: Aligning building facades. Given two building facades in arbitrary starting poses, 4PCS successfully aligns the scans. This is a challenging case for automatic registration since the scans comprise of noisy data with large flat featureless regions. This example has very few distinct features that can be reliably detected using any local descriptors. (Middle) The result is without any ICP refinement. (Right) We color the error in 4PCS alignment when compared to the final position after ICP refinement. In our scale, we set the length of the bounding box diagonal of the model to unity. Points with error more than 0.01 are marked in blue. Notice that even without ICP refinement our algorithm aligns the scans very reliably.

(Figure 8), the noise and small overlaps prevent us from achieving a linear time. We used a high threshold for the normals, whose estimates are unreliable for noisy data, thus achieving only factor of two speedup. Precomputation time for normal estimation is also included in the reported timings.

It is easy to extend our algorithm to undo small shear as shown in Figure 10. However for such affine transforms between P and Q, which cannot be uniquely determined using a single 4-points base in P, we have to construct a pair of 4-points bases with two points in common: the rest of the algorithm stays the same. Although this procedure can handle small shears, for general affine transforms the algorithm may store intermediate points quadratic in number of input points, making the procedure impractical.

Figure 13: Alignment of slippable scans using 4PCS. The congruent 4-points bases are marked, colors indicating detected correspondences. With slippable bodies, 4PCS gives *sub-optimal* solution in the direction of slippage, like along the axis of the mug in this example. Here local descriptor based methods [Gelfand et al. 2005] degenerate to brute force search due to lack of distinctive features.

Limitations. In Figure 13, we observe how 4PCS performs if the underlying object being scanned is slippable – the resulting alignment may be sub-optimal in the direction of slippage (cf. [Gelfand and Guibas 2004]). Since we operate at a point level, this problem is unavoidable. However, removal of such ambiguities might require more semantic information about the objects, and not just point positions as now being used.

In extreme scenarios of scans with very low overlap, choosing a 4-points wide base might not be possible. In such cases, we have to compromise stability of registration by selecting narrower bases. However, any alignment technique faces similar problems on such inputs.

Although 4PCS performs *affine* registration (see Figure 10), the space requirement being no longer linear in the size of point-sets, it can be impractical for very large point sets. Finally, when the input data is clean and local-descriptors can be reliably computed, our algorithm is somewhat redundant since there are simpler ways to register such scans [Gelfand et al. 2005, Li and Guskov 2005].

6 Conclusions

In this paper, we have presented 4PCS, a wide-based pairwise alignment approach. Unlike a local descriptor, a wide base provides resiliency to noise. Typically, a wide base approach requires extensive number of trials to match congruent sets of points. Introducing a coplanar 4-points base, rather than the minimum of 3-points base, allows us to employ a technique that efficiently matches pairs of affine invariant ratios in 3D.

As we showed in our analysis, the asymptotic behavior of the approach is an order of magnitude faster than prevalent alignment techniques. The speedup was further supported by our experiments on a large variety of data sets and applications.

It is worthwhile to note the distinction between noise and outliers. For partial matching, outliers and non-overlapping portions have similar effects. Although outliers hinder the effectiveness of the local descriptors, they do not affect the performance of wide bases. The 4PCS algorithm achieves resiliency to noise and outliers by using wide bases and the LCP measure.

It is important to emphasize here that typical range scanners generate noise that is not additive or Gaussian, and any pre-filtering can be harmful. Thus, we believe that many reverse engineering applications will benefit from such a robust fundamental building block in the reconstruction pipeline.

Acknowledgements. This work was supported in part by grants from the Frankel Foundation, Austrian Science Fund FWF through research network S92, the Israeli Ministry of Science, and the Israel Science Foundation. Niloy is also supported by a Microsoft Outstanding Young Faculty Fellowship.

We thank Filip Sadlo for kindly providing us the water jug data set (Figure 1), Thibaut Weise for the Coati dataset (Figure 6), Ran Gal for the Jerusalem data set (Figure 11), Baoquan Chen for the building facade dataset (Figure 12), and Alexander Bogomjakov for the mug dataset (Figure 13). We are grateful to Ran Gal, Adi Levin, Yaron Lipman, Andrei Sharf, and the anonymous reviewers for their suggestions and critical comments.

References

AGARWAL, P. K., AND SHARIR, M. 2002. The number of congruent simplices in a point set. In *Discrete Comput. Geometry*, vol. 28, 123–150.

ARYA, S., MOUNT, D. M., NETANYAHU, N. S., SILVERMAN, R., AND WU, A. Y. 1998. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *ACM 45*, 6, 891–923.

BALLARD, D. H. 1987. Generalizing the hough transform to detect arbitrary shapes. *Readings in computer vision: issues, problems, principles, and paradigms*, 714–725.

BESL, P. J., AND MCKAY, N. D. 1992. A method for registration of 3-d shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence 14*, 2, 239–256.

BROWN, B. J., AND RUSINKIEWICZ, S. 2007. Global non-rigid alignment of 3-d scans. *ACM Trans. Graph. 26*, 3, # 21.

CALLIERI, M., FASANO, A., IMPOCO, G., CIGNONI, P., SCOPIGNO, R., PARRINI, G., AND BIAGINI, G. 2004. Roboscan: An automatic system for accurate and unattended 3d scanning. In *3DPVT*, 805–812.

CHEN, Y., AND MEDIONI, G. G. 1992. Object modelling by registration of multiple range images. In *Image and Vis. Compt.*, vol. 10, 145–155.

CHEN, C.-S., HUNG, Y.-P., AND CHENG, J.-B. 1999. RANSAC-based DARCES: A new approach to fast automatic registration of partially overlapping range images. *IEEE Trans. on Pattern Analysis and Machine Intelligence 21*, 11, 1229–1234.

FISCHLER, M. A., AND BOLLES, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM 24*, 6, 381–395.

FROME, A., HUBER, D., KOLLURI, R., BULOW, T., AND MALIK, J. 2004. Recognizing objects in range data using regional point descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

GAL, R., AND COHEN-OR, D. 2006. Salient geometric features for partial shape matching and similarity. *ACM Trans. Gr. 25*, 1, 130–150.

GELFAND, N., AND GUIBAS, L. J. 2004. Shape segmentation using local slippage analysis. In *Proc. Symp. Geometry Processing*, 214–223.

GELFAND, N., MITRA, N. J., GUIBAS, L. J., AND POTTMANN, H. 2005. Robust global registration. In *Proc. Symp. Geometry Processing*, 197–206.

GERMAIN, R. S., CALIFANO, A., AND COLVILLE, S. 1997. Fingerprint matching using transformation parameter clustering. *IEEE Comput. Sci. Eng. 4*, 4, 42–49.

GOODRICH, M. T., MITCHELL, J. S. B., AND ORLETSKY, M. W. 1994. Practical methods for approximate geometric pattern matching under rigid motions. In *Symp. on Computational Geometry*, 103–112.

HORN, B. K. 1987. Closed-form solution of absolute orientation using unit quaternions. In *Journal of the Optical Society of America*, vol. 4, 629–642.

HUTTENLOCHER, D. P., AND ULLMAN, S. 1990. Recognizing solid objects by alignment with an image. *Int. J. Comput. Vision 5*, 2, 195–212.

HUTTENLOCHER, D. P. 1991. Fast affine point matching: An output-sensitive method. Tech. rep., Cornell, CS department.

INDYK, P., MOTWANI, R., AND VENKATASUBRAMANIAN, S. 1999. Geometric matching under noise: combinatorial bounds and algorithms. In *Symposium on Discrete algorithms*, 457–465.

IRANI, S., AND RAGHAVAN, P. 1996. Combinatorial and experimental results for randomized point matching algorithms. In *Symp. on Computational Geometry*, 68–77.

JOHNSON, A. 1997. *Spin-Images: A Representation for 3-D Surface Matching*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

LI, X., AND GUSKOV, I. 2005. Multi-scale features for approximate alignment of point-based surfaces. In *Proc. Symp. Geometry Processing*, 217–226.

MITRA, N. J., GELFAND, N., POTTMANN, H., AND GUIBAS, L. 2004. Registration of point cloud data from a geometric optimization perspective. In *Proc. Symp. Geometry Processing*, 23–31.

MITRA, N. J., GUIBAS, L., GIESEN, J., AND PAULY, M. 2006. Probabilistic fingerprints for shapes. In *Proc. Symp. Geometry Processing*, 121–130.

MORI, M.-G., BELONGIE, M.-S., AND MALIK, S. M.-J. 2005. Efficient shape matching using shape contexts. *IEEE Trans. on Pattern Analysis and Machine Intelligence 27*, 11, 1832–1837.

PAULY, M., MITRA, N. J., GIESEN, J., GROSS, M., AND GUIBAS, L. 2005. Example-based 3d scan completion. In *Proc. Symp. Geometry Processing*, 23–32.

POTTMANN, H., WALLNER, J., YANG, Y.-L., LAI, Y.-K., AND HU, S.-M. 2007. Principal curvatures from the integral invariant viewpoint. *Comput. Aided Geom. Des. 24*, 8-9, 428–442.

RUSINKIEWICZ, S., AND LEVOY, M. 2001. Efficient variants of the ICP algorithm. In *Proc. of 3DIM*, 145–152.

WOLFSON, H. J., AND RIGOUTSOS, I. 1997. Geometric hashing: An overview. *IEEE Comput. Sci. Eng. 4*, 4, 10–21.