

Hierarchical Multiview Rigid Registration

Yizhi Tang¹ and Jieqing Feng^{†1}¹State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou, 310058, P.R. China

Abstract

Registration is a key step in the 3D reconstruction of real-world objects. In this paper, we propose a hierarchical method for the rigid registration of multiple views. The multiview registration problem is solved via hierarchical optimization defined on an undirected graph. Each node or edge in this graph represents a single view or a connection between two overlapped views, respectively. The optimizations are performed hierarchically on the edges, the loops, and the entire graph. First, each overlapped pair of views is locally aligned. Then, a loop-based incremental registration algorithm is introduced to refine the initial pairwise alignments. After a loop is registered, the views in the loop are merged into a metaview in the graph. Finally, global error diffusion is applied to the entire graph to evenly distribute the accumulated errors to all views. In addition, a new objective function is defined to describe the loop closure problem; it improves the accuracy and robustness of registration by simultaneously considering transformation and registration errors. The experimental results show that the proposed hierarchical approach is accurate, efficient and robust for initial view states that are not well posed.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

1. Introduction

The 3D reconstruction of real-world objects using modern acquisition facilities is of great concern in many areas such as computer graphics, computer-aided design and computer vision. As a result of limited fields of view and self-occlusion, a 3D acquisition device can capture only partial information about an object from a single viewpoint. To obtain a complete model, multiple partial views of the model must be acquired. These views are independently captured in different local coordinate systems; therefore, they must be registered into a common global coordinate system. Hence, multiview registration is a necessary and important step in 3D reconstruction.

In general, multiview registration can be formulated as an optimization problem. The goal of optimization is to find a transformation for each view such that the overlapping regions of the transformed views are aligned with each other as closely as possible. If the transformations are restricted to be rigid, the problem is characterized as multiview rigid registration. Pairwise registration (alignment) is a special case of multiview registration in which only one pair of overlapped

views is considered. The problem of pairwise registration has already been well studied [CB12], and its solution can serve as an initial approach to the multiview case. For example, each overlapped pair of views is first locally aligned; then, all views can be sequentially integrated into a common coordinate system using the transitivity of the transformations. This method is termed *Sequential Integration (SI)*. However, this approach will encounter the well-known loop closure problem [GCRN09], where the transformations are not consistent in the loops and the registration result exhibits visible misalignments due to accumulated errors.

Many global algorithms have been proposed to solve the loop closure problem. Some of them diffuse the transformation errors of local pairwise alignments to achieve consistency of transformations [SLW04, GP14]. These methods are generally efficient because only the transformations need be considered in the global optimization. However, the registration errors may not be minimized following the diffusion of the transformation errors. Other global methods globally minimize the registration errors in all overlapped view pairs [WB01, LZY*14]. These methods are not as efficient as those based on the diffusion of transformation errors because the global optimization must include all data in the overlapping regions. Moreover, the choice of optimization

[†] Author for correspondence: jqfeng@cad.zju.edu.cn

strategy also has a significant impact on the registration results. For example, incremental optimization can avoid local minima but tends to produce accumulated errors, whereas simultaneous optimization can eliminate accumulated errors but may lead to local minima. The robustness with respect to initial view states that are not well-posed is another concern in the multiview registration problem.

In this paper, we present a hierarchical approach to multiview rigid registration. The multiple views and their overlapping relations are represented using an undirected graph. The nodes and edges in the graph represent the views and the overlapping relations of the view pairs, respectively. Three-level hierarchical optimization is performed on the graph. In this optimization procedure, the edges, the loops and the graph itself are the basic processing units of each level from bottom to top. The optimizations at the three levels are termed *Initial Pairwise Alignments (IPA)*, *Loop-based Incremental Registration (LIR)* and *Global Errors Diffusion (GED)*. In addition, to achieve more accurate and robust registration, a new objective function is proposed to solve the loop closure problem. It is designed to simultaneously minimize the transformation and registration errors. By combining the hierarchical optimization mechanism with the new objective function, the proposed approach is shown to be accurate, efficient and robust for views with poor position initialization. The main contributions of our work can be summarized as follows:

- A hierarchical approach to the problem of multiview rigid registration is proposed.
- A new objective function that improves the accuracy and robustness of registration by considering both transformation errors and registration errors is defined to solve the loop closure problem.

2. Related Work

Over the past two decades, the registration of 3D shapes has been an intensively studied problem. In recent works, Tam et al. [TCL*13] present a survey that classifies various techniques for both rigid and non-rigid registration, and Díez et al. [DRLS15] offer a qualitative review of several 3D coarse registration methods. In this section, we simply classify rigid registration problems as either pairwise registration or multiview registration.

Pairwise registration. The simplest case in 3D registration is the registration of two overlapping views. If point correspondences between the two views are known a priori, then the problem can be formulated as the minimization of the sum of the squared distances between all corresponding point pairs. Closed-form solutions of this minimization problem can be derived using the *Singular Value Decomposition (SVD)* method [AHB87]. Thus, correspondence is the most important factor in pairwise registration.

Distinguished by their approaches to correspondence

matching, the solutions to the pairwise registration problem can be categorized into two classes. One is the *Iterative Closest Point (ICP)* class [BM92], in which pairs of closest points are matched to serve as correspondences and the sum of the squared distances between these pairs is iteratively minimized. The ICP algorithm has been widely used, and many variants have been developed by modifying either the alignment metric, the rejection of point pairs, or the method of optimization [RL01]. For example, the point-to-point distance metric may be replaced by a more accurate point-to-plane distance metric [CM92]. Ignoring point pairs near the boundary [TL94] and rejecting point pairs for which the distance between the points is greater than a given threshold [Zha94] can eliminate incorrect correspondences and improve registration accuracy. Recently, several new optimization models, such as the Sparse ICP [BTP13] and the stochastic optimization method [AGV*14], which can enhance the robustness of registration with respect to noises and outliers, have been proposed. Given good initial positional states and sufficient overlapping regions, the ICP method and its variants are generally accurate and efficient. Hence, we adopt the ICP method to perform pairwise alignment in our proposed registration method.

Solutions of the second class are known as voting methods [GMGP05, AMC08]. In an algorithm of this type, sparse feature points are extracted for correspondence matching. Transformations are calculated based on these sparse correspondences and recorded for voting. The alignment that receives the most votes is selected as the optimal alignment. Thus, this class of approaches is insensitive to the initial positional states, and the accuracy of the registration result depends on the accuracy of feature location and matching.

Multi-view registration. Multiview registration is more complicated than pairwise registration. The simple integration of local pairwise alignments will lead to the loop closure problem. Thus, a global method should be adopted. The process of solving the loop closure problem is also called loop-closing.

Certain global methods take advantage of local pairwise alignments and perform loop-closing by diffusing the transformation errors in the initially aligned view pairs. Sharp et al. [SLW04] evenly distributed the deviations of rotation and translation among all overlapped view pairs to obtain rigid consistency in the transformations. Govindu et al. and Torsello et al. represented 3D rigid transformations using Lie algebra parameters [GP14] and dual quaternions [TRA11], respectively. They diffused the transformation errors using these new representations. An explicit loop-closing technique was proposed by Sprickerhof et al. [SNLH09]. These authors detected loops along the scanning path. Once the loops were detected, the transformation errors in the loops were recursively propagated to other adjacent loops. The approaches described above are very efficient because their optimization calculations involve only a small number of ma-

trix or algebraic operations. However, the registration errors cannot be guaranteed to be minimized because the data in the overlapping regions are not considered in the loop-closing procedure.

Other global methods attempt to minimize the sum of the registration errors in all overlapped view pairs to achieve the direct diffusion of the registration errors. Chen et al. [CM92] proposed an incremental registration algorithm wherein the first two views are registered into a *metaview*, into which the other views are incrementally merged. Another incremental algorithm that evenly diffuses the pairwise registration errors under the assumption of perfect pairwise initialization was proposed by Pulli [Pul99]. Incremental registration can avoid local minima, but it may produce accumulated deviations. To overcome this problem, Bergevin et al. [BSGL96] introduced a simultaneous registration strategy. They calculated the transformation of each view separately and then applied all transformations simultaneously before the next round of correspondence matching. Given correspondences in all overlapping regions, the simultaneous calculation of all transformations was achieved by Williams et al. [WB01]. To make the rigid transformations consistent in the global registration procedure, Liu et al. [LZY*14] proposed a globally consistent formulation by choosing a single view as an fixed anchor. In addition to the above methods, certain global methods have adopted special optimization models. Stoddart et al. [SH96] simulated the registration process using a spring model, in which the optimal registration corresponded to a stable state of the physical system. Krishnan et al. considered optimization on a manifold [KLMV05]. Pottmann et al. [PLH02] studied the dynamic registration problem and accelerated the convergence speed of the optimization by linearizing the motion between successive scanned views [BTP14]. Notably, Huang et al. [HZG*12, HG13] recently developed optimization approaches to extract consistent maps from a collection of related shapes, directly solving the loop closure problem by establishing cycle-consistent point correspondences.

3. Hierarchical Registration

Figure 1 shows an overview of the proposed hierarchical registration algorithm. The registration procedure is performed from the bottom up. First, each view pair connected by an edge in the graph is aligned locally. Then, the initial pairwise alignments are refined via loop-based incremental registration. The views in each registered loop are merged into a *metaview*, and the graph is updated. Finally, global error diffusion is applied to the entire graph.

The registration procedure begins with the initially posed views. The initialization can be achieved manually or using various voting methods [GMGP05, AMC08]. It is worth noting that the proposed registration method is not completely insensitive to the initial positions of the views, although a coarse initialization is sufficient.

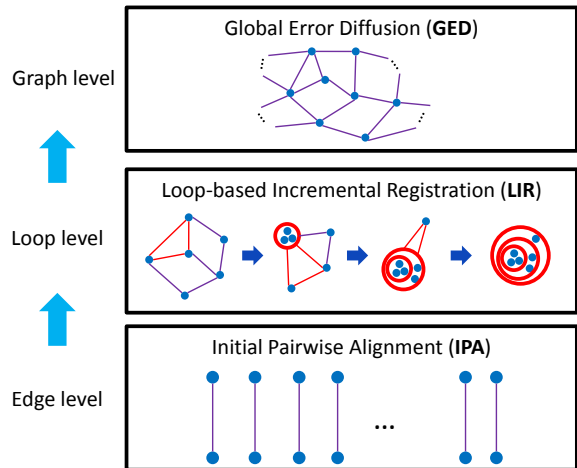


Figure 1: Overview of the hierarchical multiview rigid registration algorithm: Each blue point represents a view. Each line segment indicates the overlapping relation between a pair of views. Loops that consist of red edges are merged into metaviews (red circles).

Figure 2 offers a preview of various registration results obtained using various combinations of **SI** method and the operations that compose our algorithm. The results of the **SI** method exhibits visible misalignments as a result of the accumulated errors, as shown in the first column. The effects of **LIR** and **GED** are illustrated by the registration error histograms presented in the second column and third columns, respectively: **LIR** is designed to refine the initial pairwise alignments, and **GED** is used to evenly diffuse the registration errors of all overlapped view pairs.

3.1. Initial Pairwise Alignment

Pairwise alignment is the fundamental operation used in our approach. We employ an ICP method to perform this task. In this method, a point-to-plane distance is adopted to define the alignment metric [CM92]. We implement an approximation of this metric with a projection method. Each point equipped with a normal defines a plane. Each source point is projected on the plane defined by its closest target point. Together, the projection point on the target plane and the source point then constitute a correspondence. Although various local fitting methods could be used to obtain a high-order approximation (instead of a linear plane), this would be time consuming. We choose the first-order approximation (plane) as a trade-off between accuracy and efficiency.

When a pair of overlapped views is aligned, the resulting rigid transformation is not invertible in general, which means that the inverse transformation can not be generated by swapping the source view and the target view. This is undesirable because the presence of inconsistent direct-

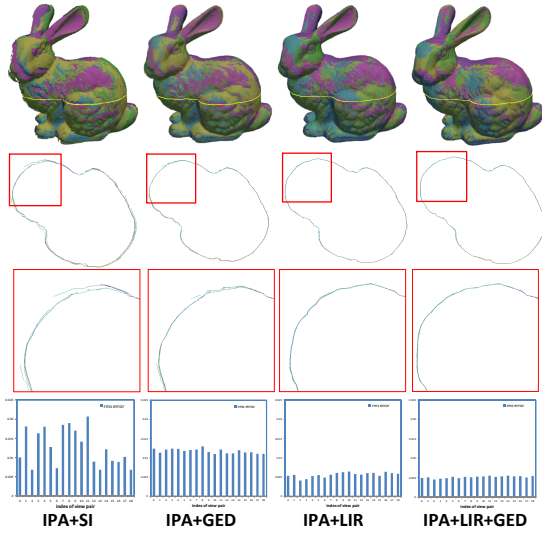


Figure 2: Illustrations of various registration results obtained using different combinations of **SI** method and the operations in our algorithm. In the first row, different colors represent different views. The second and third rows show slices and zoomed slices, respectively, of the registered models. The last row shows the registration error of each overlapped view pair in terms of the root mean square (RMS) error.

ed edges will greatly increase the complexity of addressing the loop closure problem. Therefore, we wish to represent the registration problem using an undirected graph. We address the direction issue by adopting a bi-directional projection approach [LZY*14], in which we project the points in the source view onto the target view and vice versa. Thus, the same numbers of sampling points can be collected for both the source view and the target view. In this way, a one-to-one correspondence can be established. Among the corresponding points, some will be the original sampling points, and the others are the projection points. The point correspondence will remain unchanged when the alignment direction is inverted. Thus, invertible transformations can be obtained in our initial pairwise alignment.

3.2. Loop Closure Problem

Via the bi-directional ICP method introduced in Section 3.1, a set of invertible rigid transformations is obtained from the initial pairwise alignments. Before describing the details of **LIR** at the loop level, we will first introduce a new objective function for the loop closure problem and its solution. This objective function is fundamental to both **LIR** and **GED**.

3.2.1. Loop Closure for One Loop

Let \mathbf{L} be a loop that contains L views $\{\mathbf{V}_t\}_{t=1}^L$ with $\mathbf{V}_{L+1} = \mathbf{V}_1$, and let $\Psi = \{\psi_t\}_{t=1}^L$ be the set of rigid transformations

obtained from the initial pairwise alignments, where

$$\psi_t : \mathbf{V}_t \rightarrow \mathbf{V}_{t+1}, \quad t = 1, \dots, L. \quad (1)$$

Then, the transformations in Ψ should, in theory, satisfy the following consistency constraint:

$$\psi_1 \circ \psi_2 \circ \dots \circ \psi_L = I, \quad (2)$$

where I is the identity transformation. Because all of these transformations are calculated individually from a set of mutually independent pairwise alignments, the constraint specified in Equation (2) is not fulfilled in general. There are several ways to optimize the rigid transformations in Ψ such that the optimized transformations satisfy the loop closure constraints.

Minimization of transformation errors. In this method, a set of new rigid transformations $\hat{\Psi} = \{\hat{\psi}_t\}_{t=1}^L$ is identified such that the error between the corresponding transformations in Ψ and $\hat{\Psi}$ is minimized subject to the loop closure constraint, i.e.,

$$\begin{aligned} & \underset{\hat{\Psi}}{\text{minimize}} \quad \sum_{t=1}^L \|\hat{\psi}_t - \psi_t\|^2 \\ & \text{subject to} \quad \hat{\psi}_1 \circ \hat{\psi}_2 \circ \dots \circ \hat{\psi}_L = I \end{aligned} \quad (3)$$

where the norm $\|\hat{\psi}_t - \psi_t\|$ is a metric that measures the distance between $\hat{\psi}_t$ and ψ_t , which is also called the transformation error of the pairwise alignment $\mathbf{V}_t \rightarrow \mathbf{V}_{t+1}$. The norm could be a Frobenius norm of the matrix or a metric defined in terms of the angle of rotation and the length of translation [SLW04], or other forms of the norm defined in space of Lie algebra [GP14] or in a quaternion formulation [TRA11]. The purpose of this minimization is to diffuse the transformation errors such that the variation between $\hat{\psi}_t$ and ψ_t is minimized. In theory, if the initial pairwise alignments are well-solved, then the induced variations in registration errors should also be minimized. However, the relationship between the two types of errors is not intuitive in practical applications. For example, depending on the position of the center of rotation, a small rotation may lead to a large deviation of a data set.

Minimization of registration errors. In the initial pairwise alignments, pairs of corresponding points are generated during each ICP iteration. The point pairs generated in the final iteration are called concrete point pairs [Pul99]. Another loop-closing approach for **L** concerns the registration errors of these concrete point pairs. The objective function is to used to minimize the sum of the squared distances between all of the concrete point pairs. It can be formulated as

$$\begin{aligned} & \underset{\hat{\Psi}}{\text{minimize}} \quad \sum_{t=1}^L \sum_{i=1}^{N_t} \|\hat{\psi}_t(\mathbf{x}_i^t) - \mathbf{y}_i^t\|^2 \\ & \text{subject to} \quad \hat{\psi}_1 \circ \hat{\psi}_2 \circ \dots \circ \hat{\psi}_L = I \end{aligned} \quad (4)$$

where $(\mathbf{x}_i^t, \mathbf{y}_i^t)$ is a concrete point pair as shown in Figure 3; N_t is the number of concrete point pairs between \mathbf{V}_t and

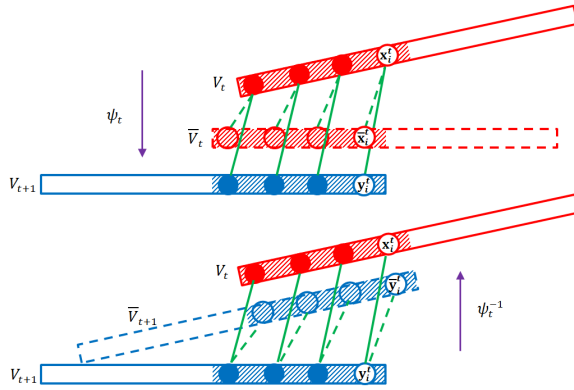


Figure 3: Illustrations of concrete pairings (solid green lines) and virtual pairings (dotted green lines): $(\mathbf{x}_i^t, \mathbf{y}_i^t)$ is a concrete point pair, $(\mathbf{x}_i^t, \bar{\mathbf{x}}_i^t)$ and $(\mathbf{y}_i^t, \bar{\mathbf{y}}_i^t)$ are virtual point pairs, and $\bar{\mathbf{V}}_t$ and $\bar{\mathbf{V}}_{t+1}$ are the virtual views transformed from \mathbf{V}_t and \mathbf{V}_{t+1} by ψ_t and ψ_t^{-1} , respectively. Note that in each concrete point pair considered, only one point is an original sample and the other is its projected point generated in the final ICP iteration, whereas the concrete point pairs considered in [Pul99] consist entirely of original sample points.

\mathbf{V}_{t+1} ; and $\sum_{i=1}^{N_t} \|\hat{\psi}_t(\mathbf{x}_i^t) - \mathbf{y}_i^t\|^2$ is called the registration error of the pairwise alignment $\mathbf{V}_t \rightarrow \mathbf{V}_{t+1}$. This formulation globally minimizes the registration errors of all pairwise alignments.

Equation (4) is an attractive formulation for solving the loop closure problem, but it may fail to address a lack of salient geometric features in certain overlapping regions, e.g. planar or spherical regions. In these regions, the results of the initial pairwise alignments often exhibit drift because their registration errors may remain very small even in the case of large drifts. The drift problem is also difficult to detect and overcome when using Equation (4) to address registration errors. However, we find that if there are large disparities in the overlapping regions of aligned view pairs that are subject to drift, then the corresponding transformations $\psi_1 \circ \psi_2 \circ \dots \circ \psi_L$ should be far from I because the drift-affected pairwise alignments often contain large transformation errors. Thus, minimizing the transformation errors using Equation (3) can be regarded as a heuristic solution to the drift problem.

Minimization of both transformation errors and registration errors. As seen from the analysis given above, a heuristic and reasonable objective function for loop closure should simultaneously minimize transformation errors and registration errors. Thus, both the concrete point pairs and the transformations in Ψ should be considered. This can be accomplished using the concept of *virtual mates* proposed by Pulli [Pul99]. Consider the concrete point pair $(\mathbf{x}_i^t, \mathbf{y}_i^t)$ in Equation (4); the virtual mates of these points, $\bar{\mathbf{x}}_i^t$ and $\bar{\mathbf{y}}_i^t$, are

defined as

$$\bar{\mathbf{x}}_i^t = \psi_t(\mathbf{x}_i^t) \quad \text{and} \quad \bar{\mathbf{y}}_i^t = \psi_t^{-1}(\mathbf{y}_i^t) \quad (5)$$

respectively, and the resulting pairs $(\mathbf{x}_i^t, \bar{\mathbf{x}}_i^t)$ and $(\mathbf{y}_i^t, \bar{\mathbf{y}}_i^t)$ are called virtual point pairs (see Figure 3). By replacing the registration errors in Equation (4) with the sum of the squared distances between virtual point pairs, the new objective function can be defined as follows:

$$\begin{aligned} \text{minimize}_{\Psi} \quad & \sum_{t=1}^L \sum_{i=1}^{N_t} (\|\hat{\psi}_t(\mathbf{x}_i^t) - \bar{\mathbf{x}}_i^t\|^2 + \|\hat{\psi}_t^{-1}(\mathbf{y}_i^t) - \bar{\mathbf{y}}_i^t\|^2) \\ = \quad & \sum_{t=1}^L \sum_{i=1}^{N_t} (\|(\hat{\psi}_t - \psi_t)(\mathbf{x}_i^t)\|^2 + \|(\hat{\psi}_t^{-1} - \psi_t^{-1})(\mathbf{y}_i^t)\|^2) \end{aligned}$$

subject to $\hat{\psi}_1 \circ \hat{\psi}_2 \circ \dots \circ \hat{\psi}_L = I$

(6)

Let us consider the point \mathbf{x}_i^t and its virtual mate $\bar{\mathbf{x}}_i^t$ in Equation (6). On the one hand, according to the definition of a virtual mate given in Equation (5) and the second line of Equation (6), the minimization of $\|(\hat{\psi}_t - \psi_t)(\mathbf{x}_i^t)\|^2$ will indirectly lead to the minimization of $\|\hat{\psi}_t - \psi_t\|$, which is equivalent to the minimization of the transformation errors. On the other hand, the virtual mate $\bar{\mathbf{x}}_i^t$ already lies in the position to be convergent, which minimizes the registration error between \mathbf{V}_t and \mathbf{V}_{t+1} . Obviously, $\bar{\mathbf{x}}_i^t$ is very close to the target point \mathbf{y}_i^t . Therefore, minimizing $\|\hat{\psi}_t(\mathbf{x}_i^t) - \bar{\mathbf{x}}_i^t\|^2$ can be regarded as an indirect method of minimizing the registration error.

Virtual pairing has another advantage over concrete pairing, i.e., its robustness to noise and outliers. For quadratic minimization, the solution will bias the answer in favor of minimizing the distances between widely separated point pairs. If we adopt concrete pairing, any noise and outliers present in the overlapping regions are likely to result in point pairs separated by longer distances compared with those that lie in well-sampled regions. This will cause the registration result to deviate from the correct one. However, in the virtual pairing approach, the geometrical shapes of the source point set and the corresponding target point set (virtual mates) are the same. Regardless of the presence of noise or outliers, the distances between the virtual point pairs are determined solely by the relative transformation between the two point sets. Thus, the optimization will be less seriously affected by noise or outliers than that in the concrete approach.

3.2.2. Loop Closure for Multiple Loops

Having defined the new objective function for the closure of one loop, we now extend it to the closure of multiple loops. Consider M views $\{\mathbf{V}_m\}_{m=1}^M$, which form K loops, $\{\mathbf{L}_k\}_{k=1}^K$. The loop \mathbf{L}_k consists of L_k views $\mathbf{V}_{t_1^k}, \mathbf{V}_{t_2^k}, \dots, \mathbf{V}_{t_{L_k}^k}$. In these views, there are P pairs of views that have overlapping regions, which were initially pairwise aligned. Let $\psi_{\alpha,\beta}$ denote the rigid transformation that aligns \mathbf{V}_α to \mathbf{V}_β , where $(\alpha, \beta) \leftrightarrow \mu$, for $\mu = 1, \dots, P$. A straightforward generalization of Equation (6) is to simultaneously close all K loops,

where the objective function remains defined as the sum of the squared distances between all virtual point pairs:

$$\begin{aligned} & \text{minimize} && \sum_{\mu=1}^P \sum_{i=1}^{N_{\mu}} (\|\hat{\Psi}_{\alpha,\beta}(\mathbf{x}_i^{\mu}) - \bar{\mathbf{x}}_i^{\mu}\|^2 + \|\hat{\Psi}_{\alpha,\beta}^{-1}(\mathbf{y}_i^{\mu}) - \bar{\mathbf{y}}_i^{\mu}\|^2) \\ & \text{subject to} && \hat{\Psi}_{t_1^k, t_2^k} \circ \hat{\Psi}_{t_2^k, t_3^k} \circ \cdots \circ \hat{\Psi}_{t_k^k, t_1^k} = I \\ & \text{for} && k = 1, \dots, K \end{aligned} \quad (7)$$

where $\hat{\Psi}_{\alpha,\beta}$ is the rigid transformation to be optimized; $(\mathbf{x}_i^{\mu}, \mathbf{y}_i^{\mu})$ is a concrete point pair in the pairwise alignment $V_{\alpha} \rightarrow V_{\beta}$; $\bar{\mathbf{x}}_i^{\mu}$ and $\bar{\mathbf{y}}_i^{\mu}$ are the virtual mates of these points, i.e., $\bar{\mathbf{x}}_i^{\mu} = \Psi_{\alpha,\beta}(\mathbf{x}_i^{\mu})$ and $\bar{\mathbf{y}}_i^{\mu} = \Psi_{\alpha,\beta}^{-1}(\mathbf{y}_i^{\mu})$; and N_{μ} is the number of concrete point pairs between V_{α} and V_{β} .

However, a constrained optimization problem is, in general, difficult to solve. Inspired by the method described in [LZY*14], we propose an elaborate approach in which Equation (7) is converted into an unconstrained optimization problem. Let $\hat{\Phi}_m$ be the transformation to be optimized for V_m , for $m = 1, \dots, M$. Then, $\hat{\Psi}_{\alpha,\beta}$ can be rewritten as

$$\hat{\Psi}_{\alpha,\beta} = \hat{\Phi}_{\beta}^{-1} \circ \hat{\Phi}_{\alpha} \quad (8)$$

By substituting Equation (8) to Equation (7), the objective function can be reformulated as

$$\begin{aligned} \mathbf{E} &= \sum_{\mu=1}^P \sum_{i=1}^{N_{\mu}} (\|\hat{\Phi}_{\beta}^{-1} \circ \hat{\Phi}_{\alpha}(\mathbf{x}_i^{\mu}) - \bar{\mathbf{x}}_i^{\mu}\|^2 + \|\hat{\Phi}_{\alpha}^{-1} \circ \hat{\Phi}_{\beta}(\mathbf{y}_i^{\mu}) - \bar{\mathbf{y}}_i^{\mu}\|^2) \\ &= \sum_{\mu=1}^P \sum_{i=1}^{N_{\mu}} (\|\hat{\Phi}_{\alpha}(\mathbf{x}_i^{\mu}) - \hat{\Phi}_{\beta}(\bar{\mathbf{x}}_i^{\mu})\|^2 + \|\hat{\Phi}_{\beta}(\mathbf{y}_i^{\mu}) - \hat{\Phi}_{\alpha}(\bar{\mathbf{y}}_i^{\mu})\|^2) \end{aligned} \quad (9)$$

In addition, the loop closure constraints given in Equation (7) will be automatically satisfied. Because the transformations in $\hat{\Phi} = \{\hat{\Phi}_m\}_{m=1}^M$ are mutually independent, the minimization of \mathbf{E} in Equation (9) becomes an unconstrained optimization problem.

Recall the single-loop case; Equation (6) can be converted into an unconstrained optimization problem in a similar manner. Let $\hat{\Phi}_t$ denote the transformation to be optimized for V_t , for $t = 1, \dots, L$. Then, $\hat{\Psi}_t = \hat{\Phi}_{t+1}^{-1} \circ \hat{\Phi}_t$, and the objective function given in Equation (6) can be reformulated as

$$\begin{aligned} \mathbf{E}(\mathbf{L}) &= \sum_{t=1}^L \sum_{i=1}^{N_t} (\|\hat{\Phi}_{t+1}^{-1} \circ \hat{\Phi}_t(\mathbf{x}_i^t) - \bar{\mathbf{x}}_i^t\|^2 + \|\hat{\Phi}_t^{-1} \circ \hat{\Phi}_{t+1}(\mathbf{y}_i^t) - \bar{\mathbf{y}}_i^t\|^2) \\ &= \sum_{t=1}^L \sum_{i=1}^{N_t} (\|\hat{\Phi}_t(\mathbf{x}_i^t) - \hat{\Phi}_{t+1}(\bar{\mathbf{x}}_i^t)\|^2 + \|\hat{\Phi}_{t+1}(\mathbf{y}_i^t) - \hat{\Phi}_t(\bar{\mathbf{y}}_i^t)\|^2) \end{aligned} \quad (10)$$

where $\mathbf{E}(\mathbf{L})$ can be regarded as a special case of \mathbf{E} in Equation (9) that is to close only one loop \mathbf{L} .

3.2.3. Optimization

The minimization of \mathbf{E} in Equation (9) can be achieved using the simultaneous registration method developed by

Williams and Bennamoun [WB01]. Unlike methods that solve each transformation separately [BSGL96] or incrementally [Pu199], the simultaneous method simultaneously solves all transformations as long as the correspondences in all overlapping regions are given. Thus, the seesaw effect can be substantially alleviated, and the amount of accumulated error will decrease.

We briefly introduce the simultaneous method as follows. For details of the proofs, readers can refer to the original paper [WB01]. In Equation (9), each unknown rigid transformation $\hat{\Phi}_m$ consists of a 3×3 rotation matrix R_m and a 3×1 translation vector T_m . We concatenate all of these matrices and vectors into a $(3 \times 3M)$ matrix \mathbf{R} and a $(3M \times 1)$ vector \mathbf{T} :

$$\mathbf{R} = [R_1 \dots R_M] \quad \text{and} \quad \mathbf{T} = \begin{bmatrix} T_1 \\ \vdots \\ T_M \end{bmatrix} \quad (11)$$

Then, \mathbf{E} becomes a function of \mathbf{R} and \mathbf{T} . First, under the assumption that \mathbf{R} is known, \mathbf{E} can be minimized by computing the partial derivative with respect to each component of \mathbf{T} and equating these partial derivatives to zero. The result is a linear system in terms of \mathbf{T} because \mathbf{E} is a quadratic function of \mathbf{T} . Thus, the optimal \mathbf{T}_{min} can be obtained by solving the linear system, which is written in terms of the given \mathbf{R} . Upon substituting \mathbf{T}_{min} back into \mathbf{E} , \mathbf{E} becomes the function only of \mathbf{R} . The optimal \mathbf{R}_{min} is determined using an iterative algorithm as follows. First, each rotation matrix R_m is initialized as an identity matrix. Then, by fixing the other rotation matrices, a temporary optimum of R_m can be obtained using the SVD method. This temporary optimization is iteratively applied to each rotation matrix R_m , for $m = 1, \dots, M$, and the optimized rotation matrix is immediately updated before the next iteration. Because the SVD method causes \mathbf{E} to decrease rapidly during each iteration, the optimization of \mathbf{R} typically converges within 10 iterations.

In this manner, both \mathbf{R}_{min} and \mathbf{T}_{min} can be obtained. The largest size of the matrices involved in the computation is $3M \times 3M$ or $3P \times 3P$, and the matrices are generally sparse, with at most $3 \times 3 \times P$ nonzero elements. Furthermore, we do not need to perform any additional correspondence matching to generate virtual point pairs because these point pairs can be determined using the transformations resulting from the initial pairwise alignments and the concrete point pairs generated in the final ICP iteration. The virtual point pairs between each overlapped view pair are transformed into a 3×3 covariance matrix for use in the optimization computation, which can be pre-calculated and recorded to avoid repetitive calculations.

3.3. Loop-based Incremental Registration

Given a set of views that contain multiple loops and have been initially pairwise aligned, the multiple loops can be simultaneously closed by solving Equation (9). However, this

Table 1: Performance of our method for the examples shown in Figure 5.

Model name	Data size (10 ⁶ Points)	Num of views	Num of edges	Num of loops	IPA time (min.)	LIR time (min.)	GED time (min.)	Total time (min.)
Buste	0.32	16	30	15	2.2	1.3	0.02	3.5
Buddha	1.09	15	31	17	3.3	2.4	0.03	5.7
Dragon	1.23	45	88	44	5.6	5.2	0.09	10.9
Neptune	4.71	39	75	37	13.4	12.2	0.3	25.9

may generate undesirable results if the view pairs are not initially well aligned. For example, when two views are initially posed far apart and their overlapping region is relatively small, the pairwise alignment is likely to be imperfect or even incorrect. Then the loops that contain the imperfectly or incorrectly aligned view pairs may suffer large transformation errors and registration errors. If these loops are directly closed, these large errors will be diffused to other perfectly aligned view pairs. This will eventually lead to the local minima problem, as shown in the second column of Figure 2. To overcome this problem, we develop a loop-based incremental registration algorithm that closes each loop individually and incrementally to refine the initial pairwise alignments.

3.3.1. Priority

The order in which the alignments are performed plays an important role in many incremental registration algorithms. A suitable order typically accelerates the convergence of optimization and reduces the risk of falling into local minima. In general the order can be determined based on pre-defined priorities. For example, the view pair with the largest overlapping region is always aligned first.

Rather than consider the priority of the view pairs, we define the priority for the view loops, which are the basic processing units of the middle level. Because pairwise alignments have been initially performed in each loop, the loop registration task is that of loop-closing. To alleviate the negative influences caused by imperfect initial pairwise alignments, the loops that contain well aligned view pairs will be closed first. Thus, the definition of the priority should reflect the quality of the initial pairwise alignments in each loop. Intuitively, the registration errors can be used to assess the quality of the pairwise alignments in a loop. As seen from the analysis presented in Section 3.2.1, a more reliable and reasonable assessment criterion should also include the transformation errors. Therefore, we use the objective function $\mathbf{E}(\mathbf{L})$ given in Equation (10) to define the loop priority.

We again consider the loop \mathbf{L} defined in Section 3.2.1. After applying loop-closing to \mathbf{L} by minimizing $\mathbf{E}(\mathbf{L})$, we can obtain the residual error in $\mathbf{E}(\mathbf{L})$, i.e., $\mathbf{E}_{min}(\mathbf{L})$, which is a reflection of both the transformation errors and the registration errors. Thus, the priority of \mathbf{L} is defined as

$$p = \frac{\mathbf{E}_{min}(\mathbf{L})}{\sum_{t=1}^L N_t} \quad (12)$$

where p is simply the mean residual error of $\mathbf{E}(\mathbf{L})$. Generally, the smaller are the registration errors and transformation errors in \mathbf{L} , the smaller p will be, then the higher is the priority that should be given to \mathbf{L} . Note that the loop-closing is tentative, which means that the transformations of $\{\mathbf{V}_t\}_{t=1}^L$ are not updated to satisfy the consistency constraints. Thus, it serves merely as a pre-computation of the priority used to guide the order of the real loop-closing operation.

3.3.2. Algorithm Pipeline

We now introduce the pipeline of our loop-based incremental registration algorithm. The methods of loop detection and selection that are used to ensure the integrity of the algorithm will be introduced in Section 3.3.3. Under the assumption of a view graph that contains a set of loops, the pipeline of this algorithm is as follows:

1. Compute the initial priorities of all loops using Equation (12).
2. Select the loop in the graph with the highest priority and apply loop-closing to it to update the transformations.
3. Merge the views in this closed loop into a metaview V_{meta} and then generate a new graph.
4. Re-perform the pairwise alignments related to V_{meta} in the new graph and update the virtual point pairs.
5. Compute the priorities for the newly generated loops.
6. Return to 2 until there are no loops left in the graph.

The loop level in Figure 1 illustrates the evolution of a graph as the above algorithm is performed. The key components of the algorithm are pairwise alignment and loop-closing, which have been introduced in detail in previous sections. The merging of a loop into the metaview should be performed after the loop is closed to ensure that transformation consistency is satisfied. The refinement of the pairwise alignment occurs in Step 4. During the alignment of a certain view V_m with the metaview V_{meta} , the overlapping region between these two views is larger than that produced in the alignment of V_m with each single view in V_{meta} . A larger overlapped region typically results in more reliable and more accurate registration result, especially when the views are provided with initial states that are not well posed.

3.3.3. Loop Detection and Selection

The above algorithm requires a set of pre-detected loops as input. Thus, we should design an efficient and effective al-

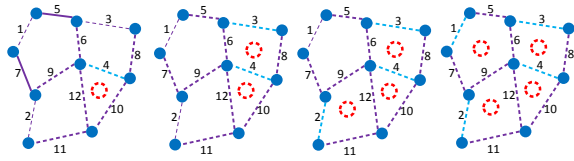


Figure 4: Loop detection and selection. The edges of the maximum spanning tree are drawn as solid purple lines. The solid blue lines are the edges that have been added back into the graph. The loops marked with red circles are the currently selected loops.

gorithm for loop detection and selection, especially for complex graphs that contain large numbers of nodes and edges. During loop selection, there are three factors that should be considered. First, it is preferable to select loops that contain large overlapping regions. Second, the path of the selected loops should be as short as possible because a long path tends to produce a large accumulated error, which cause the loop to have a low priority. Third, some loops may be redundant, i.e., they are composed of other loops.

Here, an algorithm that considers the three factors stated above is proposed as follows:

1. Compute the size of the overlapping region in each overlapped view pair after the initial pairwise alignment and set this value as the weight of the edge.
2. Extract the maximum spanning tree from the graph; the edges that contained in this tree remain in the graph, and the other edges are pushed into a priority queue.
3. The edge with the highest weight is taken from the queue and added back to the graph.
4. Select the newly generated loop with the shortest path.
5. Return to 3 until no edges remain.

Figure 4 illustrates an example of loop detection and selection. In Step 1, the size of the overlapping region can be estimated based on the number of concrete point pairs. The generation of loops begins with the maximum spanning tree extracted in Step 2 because we must guarantee the connectivity of the graph. In Step 3, once an edge is added to the graph, at least one loop will be generated, but only one of these loops will be independent. Finally, a set of independent loops that includes all of the edges is selected. If the numbers of nodes and edges are M and P respectively, then the number of selected independent loops will be $P - M + 1$.

3.4. Global Error Diffusion

The global error diffusion is performed at the top level of our approach and is applied to the entire graph. The purpose is to diffuse the accumulated deviations possibly produced in the loop-based incremental registration. This step simply consists of simultaneously applying the loop-closing to all loops by solving Equation (9). Note that the pairwise alignments



Figure 5: Registrations of real scanning range data. First column: initial states. Second column: graphs. Last two columns: the registration results from two viewpoints. Rows from top to bottom: Buste, Buddha, Dragon, and Neptune.

considered here have already been refined in the loop-based incremental registration, i.e., the virtual point pairs that are used are generated from these refined pairwise alignments.

4. Results and Discussion

The proposed algorithm was implemented on a PC with a 3.4 GHz*8 CPU (Intel i7-3770), 16GB of RAM and the Windows 7 OS. The data in our experiments are all real scanning data, except for one dataset with synthetic noise and outliers and another used for benchmarking.

4.1. Registration Results

Figure 5 shows our registration results for four classic models, with different colors indicating different views. In the first column, we present the results of a quite coarse initialization obtained using the method of Aiger et al. [AMC08]. Although we could obtain a better initialization with more careful parameter selection, we declined to do so to facilitate the investigation of the robustness of our method to with respect to poor initial view states. In the first and second rows, the number of views is small; therefore, we can clearly see

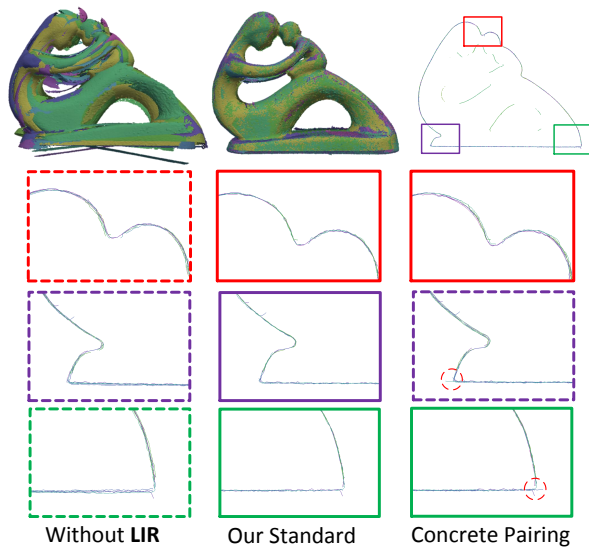


Figure 6: Registration of an example with smooth overlapping regions. The first row shows the initial view states, the registered model generated by our standard algorithm and a slice of this model. The three rows below show zoomed slices of the results of our algorithm without **LIR**, our standard algorithm, and our algorithm using concrete pairing.

the structures of the graphs. Most loops in these two graphs are simple triangles or quadrangles, and they are connected to each other in a manner similar to a strip. For example, the graph of Buddha is simply a circular strip of triangles. The third and fourth rows show more complex examples, in which the topology of the graph is quite complicated and the number of loops would be difficult to count manually. Some automatically detected and selected loops in these two graphs may have a long path, but they will eventually degenerate into short-path loops because of the merging operation performed during the loop-based incremental registration. In the third row, although the initial states is very messy, all the views of the Dragon model remain well aligned.

The performances of our method when applied to these examples are summarized in Table 1. We find that the most time-consuming step of our algorithm is the **IPA** step. This is because many closest point queries are performed via KD-tree. Because these closet point queries are mutually independent during each ICP iteration, we implemented them in a multi-thread manner, which provides an acceleration by a factor of approximately 4. The **LIR** step requires an amount of time comparable to that of the **IPA** step. During **LIR**, more than 95% of the time is spent on the re-performance of the pairwise alignments relative to the metaviews, whereas the time required for the priority calculations and loop-closing is less than 5%. As expected, the **GED** step requires the least amount of time because of the high efficiency of our adopted optimization approach.

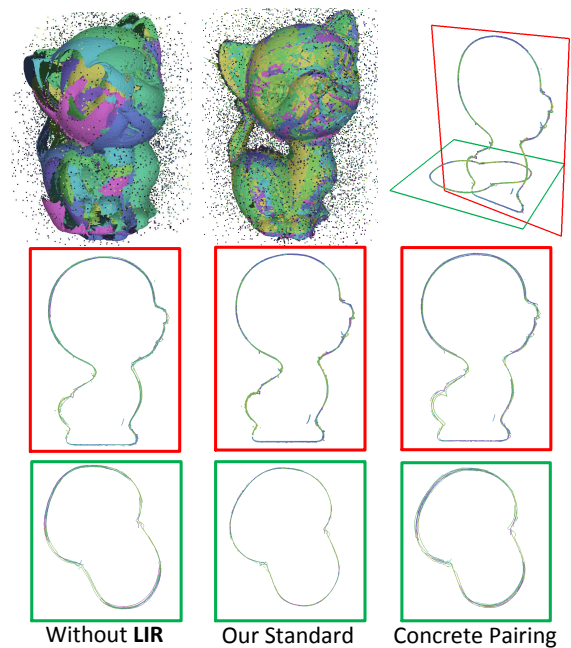


Figure 7: Registration of an example with synthetic noise and outliers. The number of synthetic points is 10% of the number of original sample points. The first row shows the initial state, the registration result generated by our standard algorithm and two slices of this result. The two rows below show clearer views of these slices, with each column corresponding to the result produced by our standard algorithm or one of its two variants, as in Figure 6.

In addition to the above examples, other data sets were also tested. Figure 6 shows a unique example in which the overlapping regions are very smooth. The bottom portion of the model is planar, and thus, the pairwise alignment in this region is likely to exhibit drift. To demonstrate the robustness of our method, two variants of our standard algorithm were implemented for comparison. One is performed without the **LIR** step and the other was performed using concrete pairing. Through carefully observation and comparison of the zoomed slices, we find that the concrete pairing approach cannot successfully address the drift problem and that direct global optimization without the **LIR** step tends become trapped in a local minimum. However, both these problems are overcome in our standard algorithm. Figure 7 presents an example that includes synthetic noise and outliers. To eliminate the negative effects caused by noise and outliers in the **IPA** step at the bottom level of our approach, the Sparse ICP method [BTP13] is adopted to execute the pairwise registration task. This example was also tested using the same two variants of our standard algorithm that were used to generate the results presented in Figure 6. The registration results demonstrate the robustness of our standard algorithm to noise and outliers.

Table 2: Comparison of registration error and performance

Test	ELCH			Liu's method			Pulli's method			Ours		
	error (10^{-4})		time (min.)	error (10^{-4})		time (min.)	error (10^{-4})		time (min.)	error (10^{-4})		time (min.)
	rms	max		rms	max		rms	max		rms	max	
1	6.38	20.18	15.36	2.47	8.54	113.27	2.55	10.74	16.14	2.32	5.06	23.79
2	20.2	258.6	23.68	33.1	573.1	134.75	3.92	15.57	24.48	2.32	5.15	32.88

4.2. Evaluation and Comparison

We evaluated our approach on a set of synthetic scanning data. Simulating a virtual scanner using the Z-buffer technique, we obtained 42 scans of the Armadillo mesh model normalized in a unit cube from 42 evenly distributed directions. Thus, each point in the scans has a corresponding ground truth. Using the synthetic scanning data, we compared our approach with three other approaches. The first approach considered for comparison was the ELCH [SNLH09], which is a representative loop-closing method that diffuses transformation errors. The second was Liu's method [LZY*14], which is a state-of-art global rigid registration method. The third was Pulli's method [Pul99], which also uses virtual pairing and shares many similarities with our method.

To evaluate the accuracy of our algorithm, we first tested the data sets with good initial position states, called Test 1 (see the first two columns in Figure 8). As shown in Table 2, our method produces a smaller error. The ELCH produces the largest error because it only distributes transformation errors to adjacent views. Pulli's method and Liu's method demonstrate a comparable accuracy to ours but with a slightly larger error. This is primarily because Pulli's optimization is incremental and Liu's optimization is achieved through a sequence of one-to-many alignments, whereas, the global optimization in our approach is simultaneously performed for all views; therefore, the errors are more evenly diffused.

To demonstrate the robustness of our algorithm, we also tested the data sets with a poor initialization, called Test 2 (see the last two columns in Figure 8). In this case, both ELCH and Liu's method fall into a local minimum that is a significant distance from the correct alignment, whereas our method works well and achieves satisfactory results as in Test 1. Pulli's method performs better than ELCH and Liu's method, but the poor initial states may lead to some imperfect initial pairwise alignments and the virtual point pairs used by Pulli are not updated as ours are; thus, its result eventually falls into a local minimum near the correct one.

The runtimes of ELCH, Pulli's method and our algorithm are shorter than that of Liu's method. This is primarily because except for Liu's method, they are all implemented in a multi-thread manner for the closest point queries. Moreover, Liu's method requires a non-linear optimization to find the optimal correspondences; therefore, it is slower. ELCH is faster than Pulli's method and our method, because except for the pairwise alignments, its optimization involves only

a small number of transformations. Pulli's method and our method optimize the the transformations and the data sets in the overlapped regions together to achieve a higher accuracy. However, Pulli's method is faster than ours because our method includes an additional LIR step to refine the initial pairwise alignments.

5. Conclusion and Future Work

We have presented a hierarchical method that addresses the multiview registration problem. Initial pairwise alignments, loop-based incremental registration and global error diffusion constitute the three levels of our method. With this hierarchical scheme, our approach is found to be efficient and robust to initial view states that are not well-posed. A new objective function that can provide high accuracy and robustness by simultaneously minimizing transformation and registration errors has also been defined for loop-closing.

Although we have provided a multi-thread implementation of our algorithm, a more efficient implementation can be achieved if we consider GPU acceleration. The bottleneck of our algorithm lies in the correspondence matching procedure, during which numerous closest point queries can be executed in parallel. Moreover, the initial pairwise alignments performed in our algorithm are mutually independent, which can provide offer greater parallelism.

Another future direction is that of nonrigid registration. Intuitively, our approach can be easily extended to a nonrigid version. We need only modify the relevant rigid transformation to be nonrigid. Our loop-closing algorithm still operates, with little modification, as long as the virtual point pairs produced by the nonrigid pairwise alignment are provided.

Acknowledgments

We would like to thank Yuan Liu and his colleagues for help with comparison experiments. We thank the Aim@Shape project and the Stanford Computer Graphics Laboratory for providing the data. We also thank the anonymous reviewers for their insightful and detailed comments. This work was supported by the National Natural Science Foundation of China under Grant Nos. 61170138 and 61472349.

References

[AGV*14] AGUS M., GOBBETTI E., VILLANUEVA A. J., MURACA C., PAJAROLA R.: SOAR: stochastic optimization for affine

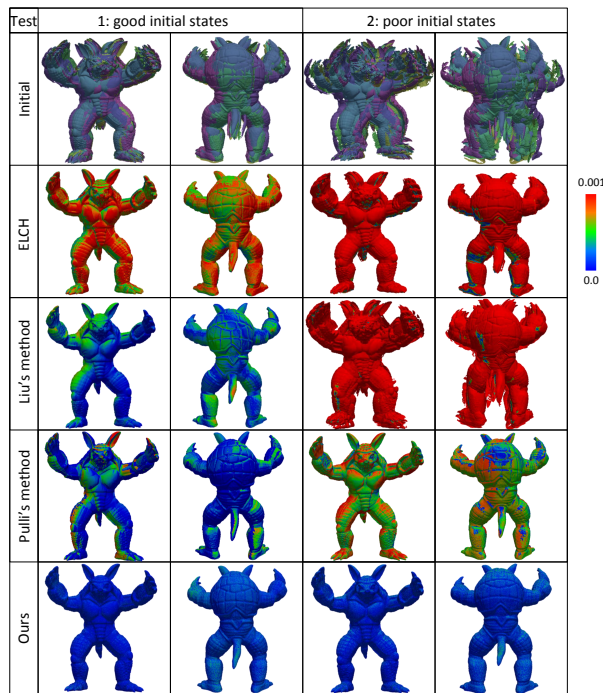


Figure 8: Comparison of different algorithms applied to good (Test 1) and poor (Test 2) initializations. The initial states are rendered with different colors representing different views. The results are rendered with error mapping colors.

- global point set registration. In *Proc. VMV '14* (Oct. 2014), p. 103–110. [2](#)
- [AHB87] ARUN K. S., HUANG T. S., BLOSTEIN S. D.: Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 9, 5 (Sept. 1987), 698–700. [2](#)
- [AMC08] AIGER D., MITRA N. J., COHEN-OR D.: 4-points congruent sets for robust pairwise surface registration. *ACM TOG* 27, 3 (Aug. 2008). [2, 3, 8](#)
- [BM92] BESL P. J., MCKAY N. D.: A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14, 2 (Feb. 1992), 239–256. [2](#)
- [BSGL96] BERGEVIN R., SOUCY M., GAGNON H., LAUREN-DEAU D.: Towards a general multi-view registration technique. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18, 5 (May 1996), 540–547. [3, 6](#)
- [BTP13] BOUAZIZ S., TAGLIASACCHI A., PAULY M.: Sparse iterative closest point. *Computer Graphics Forum* 32, 5 (Aug. 2013), 113–123. [2, 9](#)
- [BTP14] BOUAZIZ S., TAGLIASACCHI A., PAULY M.: Dynamic 2d/3d registration. In *Eurographics 2014 - Tutorials, Strasbourg, France, 2014* (Aug. 2014), p. 7. [3](#)
- [CB12] CASTELLANI U., BARTOLI A.: *3D Shape Registration*. Springer, 2012. [1](#)
- [CM92] CHEN Y., MEDIONI G. G.: Object modelling by registration of multiple range images. *Image and Vision Computing* 10, 3 (Apr. 1992), 145–155. [2, 3](#)

- [DRLS15] DÍEZ Y., ROURE F., LLADÓ X., SALVI J.: A qualitative review on 3d coarse registration methods. *ACM Comput. Surv.* 47, 3 (Feb. 2015), 45:1–45:36. [2](#)
- [GCRN09] GRANSTRÖM K., CALLMER J., RAMOS F. T., NIETO J. I.: Learning to detect loop closure from range data. In *Proc. ICRA '09* (May 2009), pp. 15–22. [1](#)
- [GMGP05] GELFAND N., MITRA N. J., GUIBAS L. J., POTTMANN H.: Robust global registration. In *Proc. SGP '05* (July 2005), pp. 197–206. [2, 3](#)
- [GP14] GOVINDU V. M., POOJA A.: On averaging multiview relations for 3d scan registration. *IEEE Transactions on Image Processing* 23, 3 (Mar. 2014), 1289–1302. [1, 2, 4](#)
- [HG13] HUANG Q., GUIBAS L. J.: Consistent shape maps via a semidefinite programming. *Computer Graphics Forum* 32, 5 (2013), 177–186. [3](#)
- [HZG*12] HUANG Q., ZHANG G., GAO L., HU S., BUTSCHER A., GUIBAS L. J.: An optimization approach for extracting and encoding consistent maps in a shape collection. *ACM TOG* 31, 6 (2012), 167. [3](#)
- [KLMV05] KRISHNAN S., LEE P. Y., MOORE J. B., VENKATASUBRAMANIAN S.: Global registration of multiple 3d point sets via optimization-on-a-manifold. In *Proc. SGP '05* (July 2005), pp. 187–196. [3](#)
- [LZY*14] LIU Y., ZHOU W., YANG Z., DENG J., LIU L.: Globally consistent rigid registration. *Graphical Models* 76, 5 (Sept. 2014), 542–553. [1, 3, 4, 6, 10](#)
- [PLH02] POTTMANN H., LEOPOLDSEDER S., HOFER M.: Simultaneous registration of multiple views of a 3d object. In *Intl. Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2002), pp. 265–270. [3](#)
- [Pu199] PULLI K.: Multiview registration for large data sets. In *Proc. 3DIM '99* (Oct. 1999), pp. 160–168. [3, 4, 5, 6, 10](#)
- [RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the ICP algorithm. In *Proc. 3DIM '01* (May 2001), pp. 145–152. [2](#)
- [SH96] STODDART A. J., HILTON A.: Registration of multiple point sets. In *Proc. ICPR '96* (Aug. 1996), pp. 40–44. [3](#)
- [SLW04] SHARP G. C., LEE S. W., WEHE D. K.: Multiview registration of 3d scenes by minimizing error between coordinate frames. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 8 (Aug. 2004), 1037–1050. [1, 2, 4](#)
- [SNLH09] SPRICKERHOF J., NÜCHTER A., LINGEMANN K., HERTZBERG J.: An explicit loop closing technique for 6d S-LAM. In *Proc. ECMR '09* (Sept. 2009), pp. 229–234. [2, 10](#)
- [TCL*13] TAM G., CHENG Z.-Q., LAI Y.-K., LANGBEIN F., LIU Y., MARSHALL D., MARTIN R., SUN X.-F., ROSIN P.: Registration of 3d point clouds and meshes: A survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics* 19, 7 (July 2013), 1199–1217. [2](#)
- [TL94] TURK G., LEVOY M.: Zippered polygon meshes from range images. In *Proc. SIGGRAPH '94* (July 1994), pp. 311–318. [2](#)
- [TRA11] TORSELLO A., RODOLÀ E., ALBARELLI A.: Multi-view registration via graph diffusion of dual quaternions. In *Proc. CVPR '11* (June 2011), pp. 2441–2448. [2, 4](#)
- [WB01] WILLIAMS J. A., BENNAMOUN M.: Simultaneous registration of multiple corresponding point sets. *Computer Vision and Image Understanding* 81, 1 (Jan. 2001), 117–142. [1, 3, 6](#)
- [Zha94] ZHANG Z.: Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision* 13, 2 (oct 1994), 119–152. [2](#)