Intrinsic Second-Order Geometric Optimization for Robust Point Set Registration Without Correspondence

Dirk Breitenreicher and Christoph Schnörr

University of Heidelberg
Image & Pattern Analysis Group (IPA)
Heidelberg Collaboratory for Image Processing (HCI)
Speyerer Strasse 6, 69115 Heidelberg, Germany

Abstract. Determining Euclidean transformations for the robust registration of noisy unstructured point sets is a key problem of model-based computer vision and numerous industrial applications. Key issues include accuracy of the registration, robustness with respect to outliers and initialization, and computational speed.
In this paper, we consider objective functions for robust point registration without correspondence. We devise a numerical algorithm that fully exploits the intrinsic manifold geometry of the underlying Special Euclidean Group $\mathsf{SE}(3)$ in order to efficiently determine a local optimum. This leads to a quadratic convergence rate that compensates the moderately increased computational costs per iteration. Exhaustive numerical experiments demonstrate that our approach exhibits significantly enlarged domains of attraction to the correct registration. Accordingly, our approach outperforms a range of state-of-the-art methods in terms of robustness against initialization while being comparable with respect to registration accuracy and speed.

1 Introduction

1.1 Overview and Motivation

Registration of point sets is an important task in many 3D vision applications including quality inspection [1, 2], reverse engineering [3], object recognition and detection [4–7]. In each case, robustness against noise, imprecise initialization and accuracy of registration are important as well as sufficiently short runtimes. Additionally, large pose variations between model and shape together with outliers and unstructured point measurements often render this problem quite challenging.

Each approach amounts to the design of an objective function and a numerical algorithm for computing an optimal registration. We review related work in Sec. 1.2. Generally speaking, since the correspondence between model and measurements is assumed to be unknown, the overall optimization problem is inherently nonconvex. Hence, robustness against poor initializations is a crucial issue.

In this context, we focus on an objective criterion that does not require to determine point correspondences explicitly. The domain of definition is therefore just the entire set of Euclidean transformations of 3D space, i.e. the Special Euclidean Group $\mathsf{SE}(3)$.

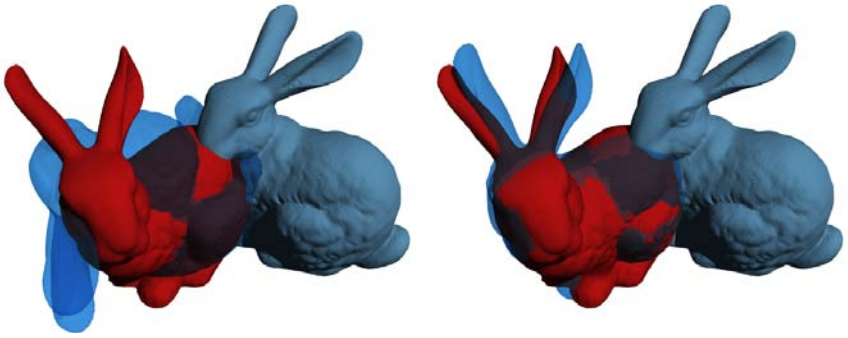

Fig. 1. Left: State-of-the-art methods that do not sufficiently take into account the manifold structure of Euclidean transformations are susceptible to imprecise initializations of the model (blue) and the scene (red) to be registered, and may reach a poor local optimum. Right: Taking the intrinsic geometry of the underlying manifold into account significantly increases robustness with respect to poor initialization.

Regarded as a matrix group, this set is a smooth manifold embedded in the corresponding ambient matrix space. We devise a Newton-like optimization algorithm that fully exploits the intrinsic manifold geometry (up to second order) to efficiently determine a locally optimal transform representing the registration.

Comparison with a range of state-of-the-art methods (see next section) reveals a significantly enlarged domain of attraction to the correct registration, thus alleviating the problem of poor initializations. Figure 1 illustrates this point, confirmed by exhaustive numerical experiments reported in section 5.

1.2 Related Work

Registration with Point Correspondence The problem to register two point sets amounts to the chicken-and-egg problem of determining simultaneously both point correspondences and a rigid transformation. Having solved either problem, the other one becomes trivial. Consequently, most approaches proceed in an alternating fashion: given an estimate of the transformation, correspondence can be determined followed by improving the estimated transformation, and so forth. The prototypical representant is the Iterative Closest Point (ICP) algorithm [8] that due to its simplicity still is a state-of-the-art algorithm [1, 2].

It is well known that this two-step iteration is susceptible to noise and poor initialization. While numerous robust variants including [9, 10] have been suggested, a major drawback concerning the *representation* of the problem remains, in particular when dealing with unstructured point sets: explicit correspondences increase both the nonconvexity and nonsmoothness of the objective function, and gaining insight into the optimization problem is hampered by complicated structure of the domain of optimization comprising *both* Euclidean transformations and correspondence variables.

Registration without Explicit Correspondence In order to obtain an optimization criteria that avoids computing corresponding points in each iterate, Mitra et al. [11] as well as Pottmann et al. [6] approximate the objective distance by local quadratic functions that represent the distance of certain points to the scene.

Another way to avoid the explicit determination of correspondence has been suggested by Tsin and Kanade [12], Jian and Vemuri [13] and Wang et al. [14]. By representing point clouds of both the scene and the model by mixture distributions, registration can be achieved by minimizing the squared L_2 distance [12, 13] or the Jensen-Shannon divergence [14] between two distributions. Compared to [11, 6] this avoids exhaustive pre-computation of the local distance approximation at the cost of more expensive function evaluations.

Because we consider this class of approaches as advantageous in connection with *unstructured* noisy point sets, we adopt mixture models of model and scene points in this paper.

Geometric Optimization However, in this case, the optimal rigid body transformation cannot be determined in closed form. In order to minimize distance measures between mixture distributions representing unstructured point sets, methods of continuous optimization like gradient descent or Newton-like schemes have to be applied. This task differs from standard applications because the underlying domain where an optimum has to be computed is a curved space (manifold).

Concerning manifolds related to the orthogonal group (Grassmann and Stiefel manifolds) continuous optimization methods are considered in [15]. Adler et al. [4], for instance, proposed a corresponding Newton-like algorithm for human spine alignment. Concerning Euclidean transformations, Li and Hartley [16] presented a Branch and Bound algorithm that determines the optimal registration of two 3D point sets together with the correspondence in terms of a permutation matrix.

A decisive advantage of this approach is its independence from initialization because the global optimum is always found. On the other hand, the runtime scales badly, e.g. nearly 20 min for 200 points, which excludes industrial applications with hundreds of points. Furthermore, point sets of *equal* cardinality are required as input which is not the case in the standard scenario of matching a model (small point set) with a scene (large point set).

Pottmann et al. [6] as well as Taylor and Kriegman [17] suggested an iterative registration algorithm based on successive local approximations of the manifold Euclidean transformations in terms of the tangent space at the current iterate.

In a similar way, Krishnan et al. [3] proposed an algorithm for multiple point set alignment. We consider this approximation in more detail below (Section 4) and work out differences to our approach (Sections 4, 5).

1.3 Contribution

In this paper we devise and study a second-order optimization method that fully exploits the geometry of the manifold $\mathsf{SE}(3)$ of Euclidean transformations in order to minimize a distance measure between two mixture distributions representing two unstructured point sets. Additionally, we show that our novel algorithm

- outperforms state of the art algorithms including ICP and Softassign [8, 9, 1, 2] in terms of speed of convergence,
- and has a significantly larger basin of quadratic convergence to the correct registration than previous work based on local approximations of $\mathsf{SE}(3)$ [6].

1.4 Organization

In Sect. 2 we recall objective criteria used for point set registration with and without explicit representation of point correspondences. Section 3 collects elements of differential geometry needed to detail our optimization approach in Sect. 4, and to point out differences to related work based on approximate Newton methods.

In Sect. 5, we compare our approach to state-of-the-art point set registration algorithms with respect to runtime and robustness against poor initializations, i.e. the size of the region of convergence to the correct registration. We conclude and point out further work in Sect. 6.

2 Objective Functions

Let $\{\boldsymbol{u}_i, i = 1, \ldots, N\} \subset \mathbb{R}^3$ denote a set of scene points obtained by a scanning device and, let $\{\boldsymbol{v}_j, j = 1, \ldots, M\} \subset \mathbb{R}^3$ be a point set specified by a given model description, i.e. a CAD file or a sample scan.

The objective of registration is to find a rigid body transformation $\boldsymbol{Y} \in \mathsf{SE}(3)$ such that model and scene points are aligned best. Here, $\mathsf{SE}(3)$ denotes the special Euclidean group parametrized by a proper rotation matrix $\boldsymbol{R} \in \mathsf{SO}(3)$ and a translation vector $\boldsymbol{t} \in \mathbb{R}^3$. There are multiple ways to parametrize rotations $\boldsymbol{R}$ like Euler angles, quaternions etc. For optimization and numerical algorithm design, however, working with the matrix representation of the group $\mathsf{SE}(3)$ Euclidean transforms is most appropriate.

2.1 Explicit Point Correspondences

The most common criterion for point registration is the sum of squared Euclidean distances of corresponding points given by

$$\min_{\boldsymbol{Y}=\{\boldsymbol{R},\boldsymbol{t}\}\in\mathsf{SE}(3)} \sum_{i=1}^{N} \|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_{\eta(i)} - \boldsymbol{t}\|_2^2 \,, \tag{1}$$

where $\eta(i) : \{1, \ldots, N\} \to \{1, \ldots M\}$ denotes a correspondence function that assigns a scene point to its model counterparts. As this function is assumed to be unknown, apart from the optimization with respect to $\boldsymbol{Y}$, we have to determine the optimal η as well. To do so, the correspondence function is replaced by weights w_{ij} assigning closest point pairs to each other, that is

$$\min_{\boldsymbol{Y}\in\mathsf{SE}(3)} \sum_{i=1}^{N}\sum_{j=1}^{M} w_{ij}\|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2 \,, \tag{2}$$

where $w_{ij} = 1$ if $j = \arg\min_k \|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_k - \boldsymbol{t}\|_2^2$ and 0 otherwise. Note, that $w_{ij} = w_{ij}(\boldsymbol{Y})$ depends on the current estimate of the transformation which complicates the optimization of (2) considerably.

The common approach is to solve alternately for the transformation parameters $\boldsymbol{R}, \boldsymbol{t}$ and correspondences w_{ij}. Drawbacks of related work in connection with unstructured point sets are discussed in Sect. 1.2.

2.2 Implicit Point Correspondences

An alternative class of approaches [13, 12] for registration utilizes kernel estimates of functions in terms of given points sets,

$$s(\boldsymbol{x}) := \sum_{i=1}^{N} \mu_i K\big(\frac{1}{2\sigma_s^2}\|\boldsymbol{x} - \boldsymbol{u}_i\|_2^2\big) \;, \tag{3a}$$

$$m(\boldsymbol{x}) := \sum_{j=1}^{M} \nu_j K\big(\frac{1}{2\sigma_m^2}\|\boldsymbol{x} - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2\big) \;, \tag{3b}$$

where $K(\cdot)$ denotes a smoothing kernel integrating to 1, and σ_s, σ_m are parameters related to the noise levels. The values $\mu_i, \nu_j \geq 0, \sum_i \mu_i = \sum_j \nu_j = 1$ signal the importance of related samples if such information is available and otherwise are set to be constant, $1/N, 1/M$, as in this paper. Thus, $s(\boldsymbol{x}), m(\boldsymbol{x})$ can be regarded as probability density estimates with respect to the assignment of points $\boldsymbol{x}$ to the scene and the model, respectively. We henceforth assume that all user parameters have been fixed beforehand.

Following [12], registrations of model and scene can now be evaluated by probabilistic distance measures of the respective distributions (3) including the Kullback-Leibler divergence

$$\begin{aligned} D(s\|m) &= \int_{\boldsymbol{x}} s(\boldsymbol{x}) \log \frac{s(\boldsymbol{x})}{m(\boldsymbol{x})} \\ &= \int_{\boldsymbol{x}} s(\boldsymbol{x}) \log s(\boldsymbol{x}) - \int_{\boldsymbol{x}} s(\boldsymbol{x}) \log m(\boldsymbol{x}) \;. \end{aligned} \tag{4}$$

We ignore the first term of (4) in the following because it does not depend on the transformation to be determined.

A further and reasonable simplification results from taking into account the noise level only in terms of a *single* smoothing parameter σ_m in (3). Correspondingly, choosing the Gaussian kernel for K in (3) and considering $\sigma_s \to 0$, function $s(\boldsymbol{x})$ becomes a sum of Dirac distributions. Insertion into the second term of (4) yields

$$\sum_{i=1}^{N} \log\Big(\frac{1}{M}\sum_{j=1}^{M} \exp\big(-\frac{1}{2\sigma_m^2}\|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2\big)\Big) \;, \tag{5}$$

where we dropped the constant $1/N$ and the factor normalizing the Gaussian because it does not depend on the transformation to be determined.

We point out that in contrast to the objective function (2), (5) only depends on the rigid body transformation and not on further variables representing point correspondences.

Furthermore, (5) parallels smoothed objective functions for prototypical clustering [18] in terms of the log of a sum of Gaussians. A corresponding optimization scheme, therefore, is given by the fixed point iteration

$$\operatorname*{argmin}_{\boldsymbol{Y}\in\mathsf{SE}(3)} \sum_{i=1}^{N}\sum_{j=1}^{M} \rho_{ij}\left(\boldsymbol{Y}^{(k)}\right) \|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2 \,, \tag{6}$$

where

$$\rho_{ij}\left(\boldsymbol{Y}\right) = \frac{\exp\left(-\frac{1}{2\sigma_m^2}\|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2\right)}{\sum_{l=1}^{M}\exp\left(-\frac{1}{2\sigma_m^2}\|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_l - \boldsymbol{t}\|_2^2\right)} \,. \tag{7}$$

This procedure is a variant of the Softassign procedure [9] (without annealing) that is significantly more robust than procedures based on hard assignments as in (2).

On the other hand, due to the structure of (6) only a linear convergence rate is achieved as will be confirmed in Sect. 5. This motivates the study of Newton algorithms that exhibit quadratic convergence rates in general. Furthermore, by fully exploiting the geometry of the underlying manifold, we increase robustness against poor initializations.

3 The Manifold of Euclidean Transformations

We collect in this section few basic concepts needed to specify and discuss our optimization approach in Sect. 4. For the mathematical background, we refer to e.g. [19, 20].

3.1 The Lie Group $\mathsf{SE}\,(3)$

Euclidean transformations $\boldsymbol{Y} = \{\boldsymbol{R}, \boldsymbol{t}\} \in \mathsf{SE}\,(3)$ map a point $\boldsymbol{x}$ to $\boldsymbol{Y}\boldsymbol{x} = \boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}$ and form a group via concatenation: $\boldsymbol{Y}_1\boldsymbol{Y}_2 = \{\boldsymbol{R}_1, \boldsymbol{t}_1\}\{\boldsymbol{R}_2, \boldsymbol{t}_2\} = \{\boldsymbol{R}_1\boldsymbol{R}_2, \boldsymbol{t}_1 + \boldsymbol{R}_1\boldsymbol{t}_2\}$. The inverse element is $\boldsymbol{Y}^{-1} = \{\boldsymbol{R}^{-1}, -\boldsymbol{R}^{-1}\boldsymbol{t}\}$.

For the purpose of optimization and numerical analysis, it is convenient to identify $\mathsf{SE}\,(3) \subset \mathsf{GL}(4)$ with a subgroup of all 4×4 regular matrices with respect to matrix multiplication. Keeping symbols for simplicity, this representation reads

$$\boldsymbol{Y} = \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{pmatrix} \,, \quad \boldsymbol{Y}^{-1} = \begin{pmatrix} \boldsymbol{R}^\top & -\boldsymbol{R}^\top\boldsymbol{t} \\ \boldsymbol{0}^\top & 1 \end{pmatrix} \,. \tag{8}$$

In this way $\mathsf{SE}\,(3)$ becomes a differentiable manifold embedded into $\mathsf{GL}(4)$, hence a Lie group.

3.2 Tangents

With each Lie group is associated its Lie algebra, the vector space tangent to the manifold at $\boldsymbol{I}$. In case of $\mathsf{SE}\,(3)$ it reads

$$\mathfrak{se}\,(3) = \left\{ \begin{pmatrix} \boldsymbol{\Phi_R} & \boldsymbol{\Phi_t} \\ \mathbf{0}^\top & 0 \end{pmatrix} \;\middle|\; \boldsymbol{\Phi_R}^\top = -\boldsymbol{\Phi_R}\,,\ \boldsymbol{\Phi_t} \in \mathbb{R}^3 \right\}\,, \tag{9}$$

which is easily deduced from the fact that $\mathfrak{se}\,(3)$ contains all matrices $\boldsymbol{\Phi}$ such that for all $t \in \mathbb{R}$, the matrix exponential $\exp(t\boldsymbol{\Phi}) \in \mathsf{SE}\,(3)$ is a Euclidean transformation, and $\boldsymbol{R} = \exp(\boldsymbol{\Phi_R})$ for some skew-symmetric $\boldsymbol{\Phi_R}$. The latter is just Rodrigues' formula for rotations in 3D.

In the following, we denote the vector space (9) equipped with the canonical inner product $\langle \boldsymbol{\Phi}_1, \boldsymbol{\Phi}_2 \rangle = \mathrm{tr}(\boldsymbol{\Phi}_1{}^\top \boldsymbol{\Phi}_2)$ with $\mathcal{T} := \mathfrak{se}\,(3)$. Furthermore, functions F and its derivatives defined on $\mathsf{SE}\,(3)$ are evaluated at $\boldsymbol{Y} = \boldsymbol{I}$ without loss of generality, because during iterative optimization the current iterate $\boldsymbol{Y}$ can be regarded as offset redefining the model's original pose.

3.3 Gradients

The gradient ∇F of a function $F\colon \mathsf{SE}\,(3) \to \mathbb{R}$ is defined by the relation [19]

$$\langle \nabla F, \boldsymbol{\Phi} \rangle = \langle \partial F, \boldsymbol{\Phi} \rangle\ ,\ \forall \boldsymbol{\Phi} \in \mathcal{T}\ , \tag{10}$$

where ∂F is the usual matrix derivative of F given by $(\partial F)_{ij} = \frac{\partial}{\partial \boldsymbol{Y}_{ij}} F$. Because $\mathsf{SE}\,(3)$ is embedded into $\mathsf{GL}(4)$, eqn. (10) shows that $\nabla F - \partial F$ is orthogonal to all $\boldsymbol{\Phi} \in \mathcal{T}$. Thus, $\nabla F \in \mathcal{T}$ is the orthogonal projection π mapping ∂F to $\mathcal{T}$. Using the same block-factorization as in (9),

$$\partial F = \begin{pmatrix} \partial F_{11} & \partial F_{12} \\ \partial F_{21} & \partial F_{22} \end{pmatrix}\,, \tag{11}$$

this projection can be computed in closed form:

$$\nabla F = \pi(\partial F) = \begin{pmatrix} \frac{1}{2}\left(\partial F_{11} - \partial F_{11}{}^\top\right) & \partial F_{1,2} \\ \mathbf{0}^\top & 0 \end{pmatrix}\,. \tag{12}$$

3.4 Hessians

In addition to the gradient, optimization with the Newton method requires to compute the Hessian of a given objective function $F(\boldsymbol{Y})$ defined on $\mathsf{SE}\,(3)$. Similar to determining the gradient in the previous section, the usual definition valid for Euclidean spaces has to be adapted to the manifold $\mathsf{SE}\,(3)$.

The Hessian of a function $F\colon \mathsf{SE}\,(3) \to \mathbb{R}$, evaluated at $\boldsymbol{Y} = \boldsymbol{I}$, is a linear mapping from $\mathcal{T}$ onto itself [20] given by $\overline{\nabla}_{\boldsymbol{\Phi}}(\nabla F)\,,\ \forall \boldsymbol{\Phi} \in \mathcal{T}$, where the gradient ∇F is given by (12) and $\overline{\nabla}$ is the Levi-Civita connection defining the covariant derivative $\overline{\nabla}_{\boldsymbol{\Phi}}$ of the vector field ∇F.

To obtain a more explicit expression in terms of the ordinary first- and second-order derivatives, we denote by $\{\boldsymbol{\mathcal{L}}_k\}_{k=1,\dots,6}$ the canonical basis spanning the translational and skew-symmetric components of tangents $\boldsymbol{\Phi} = \sum_k \phi_k \boldsymbol{\mathcal{L}}_k \in \mathcal{T}$ defined by eqn. (9). Then the quadratic form of the Hessian with respect to any $\boldsymbol{\Phi}$ is given by [15]

$$\langle \overline{\nabla}_{\boldsymbol{\Phi}} (\nabla F), \boldsymbol{\Phi} \rangle = \partial^2 F(\boldsymbol{\Phi}, \boldsymbol{\Phi}) - \langle \partial F, \Gamma(\boldsymbol{\Phi}, \boldsymbol{\Phi}) \rangle \tag{13}$$

with $\partial^2 F(\boldsymbol{\Phi}, \boldsymbol{\Phi})$ denoting the bilinear form $\sum_{ij,kl} \frac{\partial^2 F}{\partial \boldsymbol{Y}_{ij} \partial \boldsymbol{Y}_{kl}} \boldsymbol{\Phi}_{ij} \boldsymbol{\Psi}_{kl}$ and

$$\Gamma(\boldsymbol{\Psi}, \boldsymbol{\Phi}) = \sum_{i,j,k} \psi_i \phi_j \Gamma_{ij}^k \boldsymbol{\mathcal{L}}_k \ . \tag{14}$$

We list the so-called Christoffel symbols Γ_{ij}^k defining the connection $\overline{\nabla}$ in the Appendix.

4 Second Order Optimization on $\mathsf{SE}(3)$

In the usual Euclidean space $\mathbb{R}^n$, second-order optimization of some objective function $F : \mathbb{R}^n \to \mathbb{R}$ using the Newton method is based on the local quadratic model

$$F(\boldsymbol{x}^k + \boldsymbol{x}) \approx F(\boldsymbol{x}^k) + \boldsymbol{x}^\top \partial F + \frac{1}{2} \boldsymbol{x}^\top \boldsymbol{H} \boldsymbol{x}, \tag{15}$$

where $\partial F, \boldsymbol{H}$ denote the (ordinary) gradient and Hessian of F evaluated at $\boldsymbol{x}^k$, respectively. The gradient of (15) vanishes if $\boldsymbol{x}$ solves the linear system

$$\boldsymbol{H} \boldsymbol{x} = -\partial F \ , \tag{16}$$

leading to the update $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \boldsymbol{x}$.

In this section, we discuss two ways to generalize this iteration to the case of objective functions $F(\boldsymbol{Y}) \colon \mathsf{SE}(3) \to \mathbb{R}$.

4.1 Truncating the Exponential Map

It is well known that a geodesic path $\boldsymbol{Y}(t) \in \mathsf{SE}(3)$ with $\boldsymbol{Y}(0) = \boldsymbol{I}$ and tangent $\dot{\boldsymbol{Y}}(0) = \boldsymbol{\Phi}$ is locally given by the exponential mapping $\exp \colon \mathcal{T} \to \mathsf{SE}(3)$,

$$\exp(t\boldsymbol{\Phi}) = \sum_{k=0}^{\infty} \frac{(t\boldsymbol{\Phi})^k}{k!} \ . \tag{17}$$

Accordingly, it makes sense to consider local approximations

$$\boldsymbol{Y}_{lin}(t) \approx \boldsymbol{I} + t\,\boldsymbol{\Phi} \tag{18a}$$

$$\boldsymbol{Y}_{quad}(t) \approx \boldsymbol{I} + t\,\boldsymbol{\Phi} + \frac{t^2}{2} \boldsymbol{\Phi}^2 \ , \tag{18b}$$

respectively, as suggested by Pottmann et al. [6], and to determine the optimal tangent vector $t\,\boldsymbol{\Phi}$. By inserting the approximations (18a) and (18b) into $F(\boldsymbol{Y})$, and by expanding $\boldsymbol{\Phi}$ with respect to the basis $\{\boldsymbol{\mathcal{L}}_k\}_{k=1,\dots,6}$ introduced above, the objective function

$F(\boldsymbol{Y})$ is restricted to the 6-dimensional vector space $\mathcal{T}$ in terms of the coefficients $t\boldsymbol{\phi} = t(\phi_1, \ldots, \phi_6)^\top$ as variables.

As a result, the linear system (16) defining the Newton iteration is replaced by (we keep the symbols $\boldsymbol{H}$ and ∂F for simplicity)

$$\boldsymbol{H}(t\,\boldsymbol{\phi}) = -\partial F\ , \tag{19}$$

where $(\partial F)_i = \frac{\partial}{\partial \phi_i} F$ and $H_{ij} = \frac{\partial^2}{\partial \phi_i \partial \phi_j} F$ evaluated at $\boldsymbol{\phi} = \mathbf{0}$.

However, because (18a) and (18b) are only local approximations of the Euclidean group, inserting the solution $t\,\boldsymbol{\Phi} = \sum_k (t\,\phi_k)\boldsymbol{\mathcal{L}}_k$ of the linear system (19) does not give an element of $\mathsf{SE}\,(3)$ in general. Rather, the Newton update $\boldsymbol{Y} \in \mathsf{SE}\,(3)$ is determined by inserting $t\,\boldsymbol{\Phi}$ into the exponential map (17).

4.2 Intrinsic Newton Updates

Instead of restricting the objective function F to the tangent space $\mathcal{T}$ through the local manifold approximations (18) first, and then computing Newton updates by solving (19), we may base the Newton iteration directly on the intrinsic gradient and Hessian of the manifold $\mathsf{SE}\,(3)$.

This means that the linear system (16) in the Euclidean case is replaced by the linear system defined by the variational equation

$$\langle \overline{\nabla}_{\boldsymbol{\Phi}}(\nabla F), \boldsymbol{\Psi} \rangle = -\langle \nabla F, \boldsymbol{\Psi} \rangle\ , \quad \forall \boldsymbol{\Psi} \in \mathcal{T}\ , \tag{20}$$

with the gradient ∇F given by (12) and the Hessian defined in (13).

As in the previous section, the tangent vector $\boldsymbol{\Phi}$ solving (20) does not directly result in a Euclidean transformation $\boldsymbol{Y}$ as Newton update. Rather, here we use the exponential mapping

$$\boldsymbol{Y} = \exp(\boldsymbol{\Phi}) \tag{21}$$

defined by (17).

4.3 Local vs. Intrinsic Approximation

While both schemes require to solve linear systems (19) and (20) in each iteration, respectively, there are major differences in terms of convergence properties. We address this issue in this section and take it up again in connection with discussing experimental results in Sect. 5.

Recall that the objective function to be studied in this paper reads

$$F(\boldsymbol{Y}) = -\sum_{i=1}^{N} \log\Big(\frac{1}{M}\sum_{j=1}^{M} \exp\big(-h_{ij}(\boldsymbol{Y})\big)\Big)\ , \tag{22}$$

where $h_{ij}(\boldsymbol{Y}) = \frac{1}{\sigma^2}\|\boldsymbol{u}_i - \boldsymbol{R}\boldsymbol{v}_j - \boldsymbol{t}\|_2^2$ and $\boldsymbol{Y} \in \mathsf{SE}\,(3)$.

Approximating the rigid body transformation by truncating (17) after the linear term (18a) yields a redefinition of h_{ij} such that optimization of F is restricted to the tangent

space $\mathcal{T}$. Because this approach provides an accurate approximation only within a small neighborhood around the current iterate, however, convergence to the correct local optimum is unlikely if it lies outside this neighborhood [6].

In contrast, second order truncation (18b) provides a more accurate approximation of the manifold $\mathsf{SE}\,(3)$ locally. On the other hand, inserting the quadratic approximation into h_{ij} maps $\boldsymbol{R}\boldsymbol{v}_j + \boldsymbol{t}$ to

$$\left(\boldsymbol{v}_j + \boldsymbol{\Phi}_{\boldsymbol{t}} + \boldsymbol{\Phi}_{\boldsymbol{R}}\boldsymbol{v}_j + \frac{1}{2}\boldsymbol{\Phi}_{\boldsymbol{R}}\big(\boldsymbol{\Phi}_{\boldsymbol{t}} + \boldsymbol{\Phi}_{\boldsymbol{R}}\boldsymbol{v}_j\big)\right) \ . \tag{23}$$

Using the fact that $\boldsymbol{\Phi}_{\boldsymbol{R}}$ is skew symmetric, the latter part rewrites as

$$\frac{1}{2}\big(\boldsymbol{\Phi}_{\boldsymbol{R}}\boldsymbol{\Phi}_{\boldsymbol{t}} + (\phi^\top \boldsymbol{v}_j)\phi - (\phi^\top \phi)\boldsymbol{v}_j\big) \ , \tag{24}$$

where ϕ are the coefficients of the expansion $\boldsymbol{\Phi}_{\boldsymbol{R}} = \sum_k \phi_k \boldsymbol{\mathcal{L}}_k$.

As a consequence, when the rotation components of Newton updates happen to become large in magnitude, the nonconvexity of the objective function due to the quadratic terms in (24) may cause Newton updates to step into wrong directions. This will be confirmed by numerical experiments in the following section.

Finally, the intrinsic second-order approximation (20) computes update directions within the tangent space, as do the approaches discussed above based on (18). A notable difference, however, is that in this case the Hessian and the gradient utilize information of the *embedding of* $\mathsf{SE}\,(3)$ *into the ambient space* in terms of the connection and covariant derivatives, respectively, moving nearby tangents along the manifold.

We will show in the next section that this difference is relevant in practice, too.

5 Numerical Evaluation and Comparison

In 3D vision computer vision applications both robustness against poor initializations and sufficiently short processing times are of utmost importance. In this section, we apply the proposed Newton algorithm to rigid point set registration and experimentally demonstrate the major benefits and drawbacks of our scheme: *fast* convergence in a *large* region of quadratic attraction at the cost of slightly more expensive function evaluations.

Moreover, we compare the proposed scheme to a range of state-of-the-art algorithms including Iterative Closest Point [8], the fix-point iteration (6) as a special case of Softassign [9], and the Newton procedure based on local approximation of the Euclidean group [6].

In our experiments we only considered the case of perfect 3D point measurements in this paper, i.e. no noise or occlusion, in order to clearly separate for each scheme the effect of poor initializations from noise sensitivity. We point out, however, that by adjusting the kernel parameters of (5) or introducing background kernels to handle occlusion, extensions to noisy scenarios are straightforward.

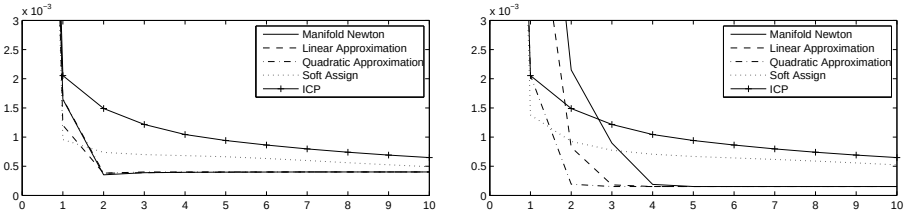

Fig. 2. Evaluation of the performance of Newton algorithms based on linear and quadratic motion approximation [6], and based on manifold structure (this paper) as well as ICP [8] and Softassign [9] for different values of σ (left: 0.3, right: 0.15). Each plot visualizes the value of the cost function (2) in the corresponding iterate. ICP and Softassign converge linearly while the remaining approaches converge quadratically.

5.1 Speed of Convergence

Algorithms like ICP [8] or Softassign [9] return less accurate registrations in cases where the underlying point set has no salient regions. This often occurs in industrial applications where smooth surfaces have to be registered accurately. To compare the ability of the approaches to cope with such scenarios, we generated 2500 data points by randomly sampling points from the smooth function $3(x-1)^2 + 3\sin(2y)$ on the unit interval.

Next, we transformed a copy of the model slightly according to a rigid body transformation (about 4 degree in each rotation and by a total of 0.12 in translation), such that all approaches ICP [8], Softassign [9], the Newton schemes based on local approximation [6] and the approach proposed in this paper always converge. Figure 2 reveals that the convergence rates differ significantly.

While for varying σ, the Newton procedures based on approximation of the Euclidean group converge slightly faster than the approach presented in this paper, all of them exhibit quadratic convergence properties. In contrast, ICP and Softassign only converge linearly to the optimal configuration. As a result, they return less accurate registrations under tight runtime constraints (fixed number of iterations).

This superior performance of the Newton schemes require more expensive computations of the Hessians in each iteration. While ICP requires $O(M \log N)$ computations in each iteration using K-D trees, the evaluation of the gradient and the Hessian of (5) causes costs of $O(MN)$. Putting this into numbers, one round of ICP requires about 1 second. In contrast, the computation of the derivatives, using MatLab research code, needs between 8 (linear and quadratic approximation [6]) and 12 seconds (our approach). This difference is primarily due to the higher dimension of the space in which the gradient and the Hessian are computed. We expect that using a more careful C-tuned implementation will return more accurate registrations if the maximum runtime is fixed beforehand, as is required in industrial applications.

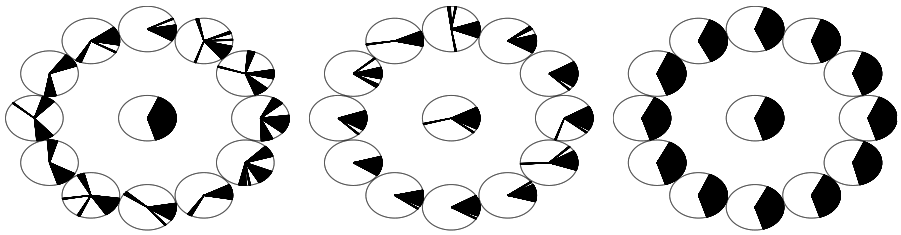

Fig. 3. Evaluation of the region of quadratic convergence for Newton algorithms based on linear (left) and quadratic (middle) local approximation [6], and based on the intrinsic local approximation (right) (this paper) for fixed $\sigma = 0.1$. Each circle center corresponds to the initial translation offset in the x-y plane, where the middle circle center is the origin. The slices in each circle refer to the initial rotation around the z-axis. These slices are colored black if the model converges to the scene within the first few iterations and otherwise remains white. The results illustrate that the approach proposed in this paper is significantly more robust against poor initializations.

5.2 Basin of Convergence

Additionally to fast convergence, robustness to poor initializations is important in many applications. The region of attraction for ICP [8] has already been analyzed in [11]. Thus we only consider Newton procedures here.

For comparison, we used the same initial setup as [11], i.e. a model of the Stanford Bunny, visualized in Fig. 1, rotated around the z-axis and shifted in the x-y plane by the size of the model. As scene we used a copy of the model placed in the origin. Moreover, since we are primarily interested in quadratic and fast convergence and the resulting accuracy after a fixed runtime, we terminated the algorithms after 25 iterations.

We observe that especially for transformations with rotational initialization error, the Newton approach proposed in this work has a significantly larger domain of attraction to the correct solution than the procedures based on local approximations of the Euclidean group, as visualized in Fig. 3 and explained in more detail in the corresponding caption.

In a related experiment we examined the update direction of a single iterate of each scheme, cf. Fig. 4. By only translating the Stanford Bunny in $\mathbb{R}^3$ we found that quadratic local motion approximation as well as our approach exhibit a lower angular error then the scheme based on linear local approximation. We point out that the angular error of all approaches near the origin is primarily due the nature of the objective function (5), that is a slight detrimental effect of the smoothed objective function discussed in Sect. 2.2. Decreasing the value of σ after few iteration steps would fix this minor issue.

6 Conclusion and Discussion

We presented a second-order optimization method that fully exploits the geometry of the manifold $\mathsf{SE}\,(3)$ of Euclidean transformations in order to minimize a distance measure between two mixture distributions representing two unstructured point sets. We

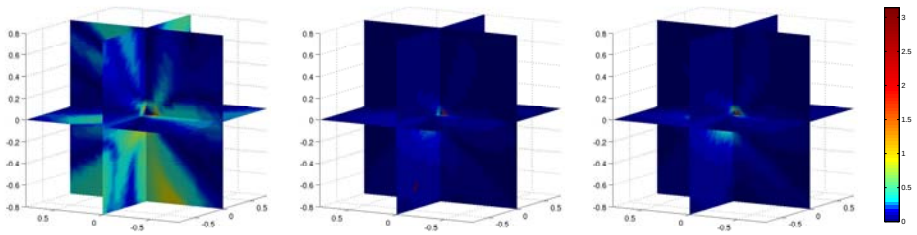

Fig. 4. Visualization of the angular error of the translational update computed with the linear (left) and quadratic (middle) local approximation approach [6], and with the intrinsic local approximation (right) (this paper), as a function of the translational offset (ground truth) model $\leftrightarrow$ scene in 3D-space. No rotation was applied. Each graphics depicts slices through the three-dimensional "error fields". While the linear local approximation fails again in this simpler scenario, both quadratic approximations are more robust against this type of initialization error. Figure 3 shows however that only the intrinsic approximation (this paper) remains stable if rotational initialization errors additionally occur.

experimentally compared this approach to state-of-the-art algorithms including ICP and Softassign [8, 9] and showed that it has a significantly larger basin of convergence to the correct registration than recent work based on local approximations of $\mathsf{SE}(3)$ [6].

This better performance comes at the cost of slightly more expensive computations required in each iteration. Thus, in further work we want to address this issue by considering approximating schemes of the objective that allow faster evaluation of the objective function.

Additionally we want to analyze the region of quadratic attraction more carefully in order to derive bounds that guarantee convergence to the desired local optimum.

Acknowledgements The authors would like to thank the VMT Vision Machine Technic Bildverarbeitungssysteme GmbH, a company of the Pepperl+Fuchs Group, for supporting this reseach work.

A Christoffel Symbols Defining the Connection $\overline{\nabla}$

The non-zero Christoffel symbols of (14) are

$$\Gamma_{12}^{3} = \Gamma_{23}^{1} = \Gamma_{31}^{2} = \frac{1}{2}, \tag{25a}$$

$$\Gamma_{13}^{2} = \Gamma_{21}^{3} = \Gamma_{32}^{1} = -\frac{1}{2}, \tag{25b}$$

$$\Gamma_{15}^{6} = \Gamma_{26}^{4} = \Gamma_{34}^{5} = 1, \tag{25c}$$

$$\Gamma_{16}^{5} = \Gamma_{24}^{6} = \Gamma_{35}^{4} = -1. \tag{25d}$$

References

1. Shi, Q., Xi, N., Chen, Y., Sheng, W.: Registration of Point Clouds for 3D Shape Inspection. In: Int. Conf. Intelligent Robots and Systems. (2006)
2. Zhu, L., Barhak, J., Shrivatsan, V., Katz, R.: Efficient Registration for Precision Inspection of Free-Form Surfaces. Int. J. Adv. Manuf. Technol. **32** (2007) 505–515
3. Krishnan, S., Lee, P.Y., Moore, J.B., Venkatasubramanian, S.: Optimisation-on-a-Manifold for Global Registration of Multiple 3D Point Sets. Int. J. Intell. Syst. Technol. Appl. **3**(3/4) (2007) 319 – 340
4. Adler, R.L., Dedieu, J.P., Margulies, J.Y., Martens, M., Shub, M.: Newton's Method on Riemannian Manifolds and a Geometric Model for the Human Spine. IMA J. Numer. Anal. **22**(3) (2002) 359 – 390
5. Frome, A., Huber, D., Kolluri, R., Bülow, T.: Recognizing Objects in Range Data using Regional Point Descriptors. In: Proc. Europ. Conf. Comp. Vision. (2004)
6. Pottmann, H., Huang, Q.X., Yang, Y.L., Hu, S.M.: Geometry and Convergence Analysis of Algorithms for Registration of 3D Shapes. Int. J. Computer Vision **67**(3) (2006) 277–296
7. Rodgers, J., Anguelov, D., Pang, H.C., Koller, D.: Object Pose Detection in Range Scan Data. In: Proc. Conf. Comp. Vision Pattern Recogn. (2006)
8. Besl, P.J., McKay, N.D.: A Method for Registration of 3-D Shapes. IEEE Trans. Pattern Anal. Mach. Intell. **14**(2) (1992) 239–256
9. Rangarajan, A., Chui, H., Bookstein, F.L.: The Softassign Procrustes Matching Algorithm. In: Proc. Int. Conf. Inf. Process. Med. Imaging. (1997)
10. Rusinkiewicz, S., Levoy, M.: Efficient Variants of the ICP Algorithm. In: Proc. Int. Conf. 3D Digital Imaging and Modeling. (2001)
11. Mitra, N.J., Gelfand, N., Pottmann, H., Guibas, L.: Registration of Point Cloud Data from a Geometric Optimization Perspective. In: Proc. Sym. Geom. Process. (2004)
12. Tsin, Y., Kanade, T.: A Correlation-Based Approach to Robust Point Set Registration. In: Proc. Europ. Conf. Comp.Vision. (2004)
13. Jian, B., Vemuri, B.C.: A Robust Algorithm for Point Set Registration Using Mixture of Gaussians. In: Proc. Int. Conf. Comp. Vision. (2005)
14. Wang, F., Vemuri, B.C., Rangarajan, A., Schmalfuss, I.M., Eisenschenk, S.J.: Simultaneous Nonrigid Registration of Multiple Point Sets and Atlas Construction. In: Proc. Europ. Conf. Comp. Vision. (2006)
15. Edelman, A., Arias, T.A., Smith, S.T.: The Geometry of Algorithms with Orthogonality Constraints. SIAM J. Matrix Anal. Appl. **20** (1999) 303–353
16. Li, H., Hartley, R.: The 3D-3D Registration Problem Revisited. In: Proc. Int. Conf. Comp. Vision. (2007)
17. Taylor, C.J., Kriegman, D.J.: Minimization on the Lie Group SO(3) and Related Manifolds. Technical Report 9405, Center for Systems Sciene, Dept. of Electrical Engineering, Yale University (1994)
18. Teboulle, M.: A Unified Continuous Optimization Framework for Center-Based Clustering Methods. J. Mach. Learn. Res. **8** (2007) 65–102
19. Matsushima, Y.: Differentiable Manifolds. Marcel Dekker, INC. New York (1972)
20. do Carmo, M.P.: Riemannian Geometry. Birkhäuser Boston (1992)