Sparse Iterative Closest Point

Sofien Bouaziz Andrea Tagliasacchi Mark Pauly
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Abstract

Rigid registration of two geometric data sets is essential in many applications, including robot navigation, surface reconstruction, and shape matching. Most commonly, variants of the Iterative Closest Point (ICP) algorithm are employed for this task. These methods alternate between closest point computations to establish correspondences between two data sets, and solving for the optimal transformation that brings these correspondences into alignment. A major difficulty for this approach is the sensitivity to outliers and missing data often observed in 3D scans. Most practical implementations of the ICP algorithm address this issue with a number of heuristics to prune or reweight correspondences. However, these heuristics can be unreliable and difficult to tune, which often requires substantial manual assistance. We propose a new formulation of the ICP algorithm that avoids these difficulties by formulating the registration optimization using sparsity inducing norms. Our new algorithm retains the simple structure of the ICP algorithm, while achieving superior registration results when dealing with outliers and incomplete data. The complete source code of our implementation is provided at http://lgg.epfl.ch/sparseicp.

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

(c) 2013 The Author(s)
Computer Graphics Forum © 2013 The Eurographics Association and Blackwell Publishing Ltd. Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.
1. Introduction

The registration of digital geometry is a fundamental task in computer graphics and geometry processing. In this paper we focus on pairwise registration, where we aim to compute the optimal alignment of a source onto a target model. We additionally assume that the aligning transformation is rigid, that is, decomposable into a rotation and a translation.

Registration applications. Pairwise rigid alignment is widely employed as a sub-routine in a number of applications. In the acquisition of digital models, self-occlusion and limited sensor range create the need to scan the object from multiple directions; these scans have to be aligned into a common frame of reference to enable further processing [Pul99, GMGP05]. In digital quality inspection, a scan of a physical model needs to be registered with its ground truth CAD model, so to be able to measure and classify possible manufacturing errors [LG04]. In mobile robotics, matching a scan of range data to a digital representation of the scene is used to refine coarse satellite based localization [SHT09]. Even in non-rigid registration, a rigid alignment step is often used to bring a template of the deforming geometry into coarse alignment with the input data [PMG*05, LSP08, WBLP11]. Recently, consumer level 3D scanning devices (e.g., Microsoft Kinect) have lead to a growing interest in robust rigid alignment algorithms. The low cost of these acquisition devices comes at the expense of severely degraded data, which necessitates registration algorithms that can deal with large amounts of noise and outliers; see Figure 3.

Iterative closest point. Given two sets of points related by a correspondence relationship, there exist several ways of computing the optimal rigid transformation that aligns them with each other [ELF97]. The practical rigid registration problem consequently simplifies to finding a suitable set of corresponding points on source and target. The Iterative Closest Point (ICP) algorithm [BM92] addresses this problem by assuming the input data to be in coarse alignment. Under this assumption, a set of correspondences can be obtained by querying closest points on the target geometry. With a convergence guarantee, ICP computes a locally optimal registration by alternately solving for closest correspondences and optimal rigid alignment.

Correspondence outliers. As ICP is effectively performing local optimization, the quality of the solution is related to the quality of correspondences provided as input to the rigid transformation sub-routine. However, incorrect closest-point correspondences are particularly common in the registration of acquired geometry. Closest point queries are not only corrupted by measurement noise, but also by partial overlap of the source and target – many samples on the source simply do not have an ideal corresponding point on the target shape. To address this problem, various techniques rely on a set of heuristics to either prune or downweight low quality correspondences. Typical criteria include discarding correspondences that are too far from each other, have dissimilar normals, or involve points on the boundary of the geometry; see [RL01] for details. These heuristics are often difficult to tune. For example, classifying boundary points in point cloud data is ill-posed and typically obtained by heuristic methods [BL04]. Furthermore, fixing a single parameter for distance thresholds is typically not sufficient, often leading to sub-optimal alignment; see Figure 4 for an example.

Sparsity inducing norms. In this paper, we propose a solution that implicitly models outliers using sparsity. Our work is based on recent advances in sparsity-inducing penalties [BJMO12, MS12] that have been successfully applied in compressive sensing [CW08]. We formulate the local alignment problem as recovering a rigid transformation that maximizes the number of zero distances between correspondences. This can be achieved by minimizing the $\ell_p$ norm of

![Figure 2: (left) A plot of the penalty functions used to induce sparsity in our optimization; for small values of $p$, large outliers do not incur a large penalty in the optimization; this allows the optimization to effectively discard correspondence outliers when computing the optimal rigid transformation. (right) The alignment of the “coatie” dataset from [AMCO08] with several values of $p$ matching the ones in the plot. By decreasing the value of $p$, the quality of the registration improves. The distribution of alignment residuals in the histograms highlights the sparse characteristics of our optimization.](image-url)
the vector of error residuals. In our work, we show how the ICP algorithm can be reformulated using \( \ell_p \) norms, where \( p \in [0,1] \), instead of the classical squared \( \ell_2 \) norm (see Figure 2). Having the possibility to choose \( p \) between zero and one allows a tradeoff between efficiency and robustness.

**Contributions.** We propose a new technique for rigid alignment that robustly deals with significant amounts of noise and outliers, while not requiring any heuristic for correspondence pruning. We propose a sparse \( \ell_p \) \((p \leq 1)\) optimization problem that automatically learns the separation of inliers and outliers, instead of explicitly pruning them. A notable side effect of this approach is an increase in the sparsity of the alignment residuals vector. This observation motivates the recent trend to model inliers/outliers by explicit approaches to registration.

### 2. Background and related works

Over the last two decades, the registration of digital geometry has been an intensively studied problem. The recent survey by Tam et al. [TCL*12] provides a classification of several techniques for both rigid and non-rigid registration. In this section, we provide a simpler classification of rigid registration methods, where we distinguish solutions according to whether they seek a global optimum, or whether they attempt to perform a local refinement of an initial, possibly coarse, alignment. Roughly speaking, these optimizations attempt to minimize an alignment energy that measures the proximity of source and target models according to a given metric; see Section 3 for a detailed discussion.

#### Global optimization.

Global approaches to registration exploit the small number of degrees of freedom of rigid transformations (6DoF). These transformations can be fully specified by two sets of matching points having cardinality of at least three. Global methods transform the continuous alignment problem into a discrete one, where the objective is to seek one of these pairs; see [vKZHC011, Section 5.1]. A notable contribution in this domain is the paper by Aiger et al. [AMCO08] that exploits geometric invariants of rigid transformations. They proposed an \( O(n^2) \) RANSAC algorithm capable of computing registration of geometry containing up to 40% outliers and having just 40% overlap. It is critical to note that global methods typically require local refinement in post processing in order to achieve high quality alignment. This creates a need for algorithms capable of performing robust local alignment – the problem we address in this paper.

#### Local optimization.

Local ICP approaches refine the alignment assuming an initial coarse registration of source and target models is provided. Although commonly used in static registration, these algorithms are especially useful in real-time registration; see [RL01]. Indeed, assuming a sufficiently dense temporal sampling, the alignment achieved in the previous time-step can be used as initialization for the registration refinement. Registration based on local optimization effectively performs a descent optimization of the alignment energy [PHYH06]. Techniques in this class can be distinguished according to the way in which they approximate the alignment metric. When the distance to the closest point on the target is used, we obtain the point-to-point method of the original ICP algorithm by Besl and McKay [BM92]. Unfortunately, closest point distances only provide a good approximation of the distance function of the target geometry in far-field conditions [PH03]; a first-order Taylor expansion of the distance field can be used to improve this approximation in the near-field, resulting in the well known point-to-plane ICP variant [CM91]. Second-order approximations of the squared distance function [PH03] are also possible, resulting in schemes able to achieve quadratic convergence [PHYH06]. Note that although iteratively fetching closest points is the most common way of approaching local registration, there also exists the possibility of caching an approximation of the distance function at a desired precision [CLSB92, PLH04, MGPG04], or even directly aligning two distance functions [TK04, JHVM11].

#### Outliers and partial overlap.

Various methods and heuristics have been proposed to improve the robustness of ICP at its different stages to cope with noisy and incomplete data [RL01]. Another prominent approach uses robust functions [Zha94, MY95, TFR99, Fit03] to reduce the importance of outliers, instead of explicitly pruning them. A notable side effect of this approach is an increase in the sparsity of the alignment residuals vector. This observation motivates the recent trend to model inliers/outliers by explic-
Our regularizer is based on the \( \ell_1 \) regularization [BMJO12]. Unlike the above-mentioned approaches, our regularizer is based on the \( \ell_p \) norm with \( p \in [0, 1] \). It has been shown recently that \( \ell_p \) norms with \( p < 1 \) outperform the \( \ell_1 \) norm in inducing sparsity [Cha07], which makes our optimization more resilient to a large number of outliers. However, when \( p < 1 \), the resulting optimization problem is non-smooth and non-convex. It is therefore necessary to carefully design the optimization problem in order to obtain a robust and efficient solution [MS12,CW13]. We present a reformulation of ICP that takes advantage of \( p \)-norms, while still leading to a simple implementation that uses the core components of the original algorithm: optimized \( \ell_2 \) closest point search [FBF77] and least-squares optimization for rigid transformation [ELF97].

3. Pairwise rigid registration

Given two surfaces \( X, Y \) embedded in a \( k \)-dimensional space, we formulate the pairwise registration problem as

\[
\begin{align*}
\arg\min_{R,t} \ & \int_X \phi(Rx + t, Y) dx + I_{SO(k)}(R), \\
\text{where } R \in \mathbb{R}^{k \times k} & \text{ is a rotation matrix, } t \in \mathbb{R}^k \text{ is a translation vector, and } x \in \mathbb{R}^k \text{ is a point on the source geometry. The rigidity of the transformation is enforced by constraining } R \text{ to the special orthogonal group } SO(k) \text{ using the indicator function } I_A(b) \text{ that evaluates to 0 if } b \in A \text{ and to } +\infty \text{ otherwise.} \\
\end{align*}
\]

This objective can be achieved by attempting to find a \( \ell_2 \) norm counts the number of non-zero entries, thus mitigating the number of outliers. Conversely, we approach the problem by identifying an appropriate robust metric \( \phi \), and present an algorithm to efficiently solve the associated optimization problem.

4. Robust alignment by sparsity enforcement

Given a generic optimization residual \( z \in \mathbb{R}^k \), an outlier-robust scheme attempts to automatically categorize a vector of residuals \( z = [||z_1||, \ldots, ||z_n||]^T \) into a large set of inliers having \( ||z_i||_2 \approx 0 \) and a small set of outliers having \( ||z_i||_2 \gg 0 \). This objective can be achieved by attempting to find a sparse vector \( z \). As the \( \ell_0 \) norm counts the number of non-zero entries in a vector, optimizing for sparsity can be reformulated as minimizing \( ||z||_0 \). However, due to the problems caused by the high non-convexity of the \( \ell_0 \)-norm, a popular choice is to optimize for sparsity by employing the \( \ell_1 \)-norm [CRT06]. The \( \ell_1 \) norm penalizes the number of non-zero entries, thus inducing sparsity: it is the closest convex relaxation of the \( \ell_0 \)-norm.

Non-convex relaxation. In this paper, we employ non-convex \( \ell_p, p < 1 \) relaxations of the \( \ell_0 \) norm and optimize it with the Alternating Direction Method of Multipliers

\[
\begin{align*}
\text{Step 1: } \ & \arg\min_{Y} \sum_{i=1}^n \phi(Rx_i + t, y_i) + I_Y(y_i) \\
\text{Step 2: } \ & \arg\min_{R,t} \sum_{i=1}^n \phi(Rx_i + t, y_i) + I_{SO(k)}(R)
\end{align*}
\]
(ADMM). This approach has recently been shown to outperform its $\ell_1$ counterpart in quality of results and performance [Cha07,MS12]. As we will demonstrate in Section 7, choosing $p < 1$ significantly improves the resilience of the method to large amounts of outliers.

Robust distance function. In our rigid alignment problem, given a pair of corresponding points $(x_i, y_i)$, the alignment residual is given by the vector $z_i = R x_i + t - y_i$. Consequently, to optimize for alignment residual sparsity, we choose $\phi$ in Equation 4 to be $\phi(x,y) = \phi(\|x - y\|_2)$, where $\phi(r) = |r|^p$ and $p \in [0, 1]$. Note that $\phi$ constructed in this fashion is still a metric [BK08, Chapter 1]. The curves of these functions, illustrated in Figure 2 for the simple scalar problem, can be interpreted as penalty curves; an outlier, having $\|z\|_2 > 0$, will not be strongly penalized when a small value for $p$ is chosen. This implies that the optimization will not skew the solution in order to reduce the large penalty associated with an outlier (see Figure 1). Please note that in this paper we are not solving for simple element-wise sparsity, but instead for group sparsity [EKB10], where each dimension of a residual vector $z_i$ should vanish simultaneously (see Figure 5).

5. Numerical optimization

The introduction of sparse $p$-norms to increase robustness in the rigid alignment optimization results in a localized change to the classic ICP formulation. Hence, we still adopt the well-known two-step optimization:

$$\text{Step 1:} \quad \arg \min_{Y} \sum_{i=1}^{n} \| R x_i + t - y_i \|_2^p + I_Y(y_i) \quad (8)$$

$$\text{Step 2:} \quad \arg \min_{R,t} \sum_{i=1}^{n} \| R x_i + t - y_i \|_2^p + I_{SO(3)}(R) \quad (9)$$

However, for $p \in [0,1]$ these two problems are non-convex and non-smooth. We explain below how we can nevertheless obtain an efficient optimization algorithm.

5.1. Step 1 - Correspondences

$\phi(r) = |r|^p$ is a non-decreasing function on $\mathbb{R}^+$, $\phi(\|\cdot\|_2)$ and thus achieves its minimum value at the same points as $\|\cdot\|_2$. Consequently, the optimization in Equation 8 is equivalent

Figure 4: (a) The alignment of the virtually scanned “owl” model is evaluated by the root mean square error (RMSE) $\epsilon$ w.r.t the ground truth alignment. (b,c,d) Traditional ICP registration is combined with correspondence pruning where correspondences with a distance above $\delta_h\%$ of the diagonal bounding box are rejected. As illustrated, it is difficult to find an appropriate threshold, which leads to sub-optimal alignment results. (e) $\ell_1$-ICP without explicit outlier management converges to a better minimum, but the alignment is still poor. (f) Our $\ell_p$-ICP outperforms all the previous methods.

Figure 5: Illustration of the difference between group sparsity and element-wise sparsity. For rigid alignment it is important to consider group sparsity, as a good corresponding pair will have small values in $x,y,z$ simultaneously.

Figure 6: Our shrinkage operator for $p \approx 0$ acts as a binary outlier classifier. Upon convergence, this classification simply identifies overlapping regions in the aligned geometry (marked in black).
(a) $p$-norm weights  
(b) Tukey weights  
(c) fair weights

Figure 7: A qualitative comparison of whole-in-part registration using point-to-point ICP. We compare two “robust weight functions” with the $p$-norm weight function presented in Equation 11. In (b) and (c) the weight functions fail to produce the correct registration because of the small amount of overlap, while our method succeeds in computing a perfect registration (a). It is interesting to note that the $p$-norm weight functions tend to infinity when approaching zero, strongly enforcing sparsity. We refer the reader to [MB93] for the definition of the robust weight functions.

\[
\text{Step 2.1: } \arg\min_{Z} \sum_{i} \left| |x_i|^{p} - z_i |^{p} + I_{SO(3)}(R) \right|
\]

\[
\text{Step 2.2: } \arg\min_{R, t} \sum_{i} \left| |x_i|^{p} - (t - y_i) |^{p} + I_{SO(3)}(R) \right|
\]

\[
\text{Step 2.3: } \lambda_i = \lambda_i + \mu \tilde{\delta}_i
\]

where $e_i = y_i + t - y_i - z_i$ is introduced solely for compactness of notation. As detailed in Appendix A, augmented Lagrangian methods are effective tools to approach the constrained optimization problem above. The augmented Lagrangian function for Equation 12 is defined as

\[
L_{A}(R, t, Z, \Lambda) = \sum_{i=1}^{n} |x_i|^{p} + \lambda_i \delta_i + \frac{\mu}{2} \|\delta_i\|_2^2 + I_{SO(3)}(R).
\]

where $\Lambda = \{\lambda_i \in \mathbb{R}^n, i = 1 \ldots n\}$ is a set of Lagrange multipliers and $\mu > 0$ is a penalty weight. We optimize this function by employing the Alternating Direction Method of Multipliers (ADMM); see Appendix B. ADMM effectively decomposes our problem into three simple steps:

\[
\text{Step 2.1: } \arg\min_{Z} \sum_{i} \left| |x_i|^{p} - z_i |^{p} + I_{SO(3)}(R) \right|
\]

\[
\text{Step 2.2: } \arg\min_{R, t} \sum_{i} \left| |x_i|^{p} - (t - y_i) |^{p} + I_{SO(3)}(R) \right|
\]

\[
\text{Step 2.3: } \lambda_i = \lambda_i + \mu \tilde{\delta}_i
\]

The values of $\beta$ and $\tilde{h}_i$ are detailed in Appendix C. The shrink operator can be interpreted as a classifier acting on residual vectors. For example, when $p = 0$, $\beta$ from Equation 16 will always evaluate to one; this results in a binary classification: the operator either rejects the value $h_i$ or accepts it fully. In Figure 6, we highlight correspondences classified as inliers which are naturally located in regions of overlap between source and target. Since the only free variables in Step 2.2 are $R, t$, this least-squares problem can be approached by classical rigid transformation estimation techniques.

6. Higher order metric approximants

Rusinkiewicz and Levoy [RL01] experimentally observed how the point-to-plane variant of ICP has better convergence speed than its point-to-point variant. Subsequently, Pottmann et al. [PHYH06] formally demonstrated how under appropriate conditions, these classical algorithms possess linear resp. quadratic convergence properties. These observations were made possible by expressing rigid alignment as an optimization problem involving the Taylor expansion of the $\ell_2$ metric $\varphi(x, y)$ for $y \in \gamma$, the “foot point” of $x$, i.e. the closest point of $x$ onto $\gamma$ [FH10, pg.63], as

\[
\varphi(y) \approx \| x_0 - y \|_2 + \left( \frac{x_0 - y}{\| x_0 - y \|_2} \right)^T (x - y).
\]
As \( x_0 \) approaches \( \mathcal{Y} \) at the foot point \( \bar{y} \), the term \( \|x_0 - \bar{y}\|_2 \) vanishes and \( \phi_\mathcal{Y}(x) \) can be rewritten as

\[
\tilde{\phi}_\mathcal{Y}(x) \approx n^T (x - \bar{y}),
\]

where \( n \) is the normal of the surface \( \mathcal{Y} \) at \( \bar{y} \). Consequently, we can linearize the metric in Equation 8 as \( \phi(x, \bar{y}) =\|x - \bar{y}\|_2 \approx \phi(\tilde{\phi}_\mathcal{Y}(x)) \) and rewrite the optimization as

\[
\arg\min_{R, t} \sum_{i=1}^n |\tilde{\phi}_\mathcal{Y}(Rx_i + t)|^p + I_{SO(3)}(R). \tag{19}
\]

This problem can be optimized using the same techniques introduced in Section 5, resulting in the ADMM steps:

**Step 2.1:**
\[
\arg\min_Z \sum_i |z_i|^p + \frac{\mu}{2} (c_i - h_i)^2
\]

**Step 2.2:**
\[
\arg\min_{R, t} \sum_i (\delta_i - c_i)^2 + I_{SO(3)}(R)
\]

**Step 2.3:**
\[
\lambda_i = \lambda_i + \mu \delta_i
\]

where \( \delta_i = n^T (Rx_i + t - \bar{y}_i) \), \( h_i = \delta_i + \lambda_i / \mu \) and \( c_i = \bar{z}_i - \lambda_i / \mu \).

By comparing the optimization step in Equation 20 with the one of Equation 13, it becomes clear that the sparsity formulation of point-to-plane ICP involves a simple scalar problem. Furthermore, Equation 21 is nothing but a traditional least-squares point-to-plane optimization that we can solve by an elementary linearization of rotation matrices [Rus13].

\section{Evaluation}

In this section we empirically evaluate the characteristics of our method in comparison to some variations of the traditional ICP approach. In Figures 1, 2, and 4, we compare against the recent \( \ell_1 \)-ICP technique from [FH10]. Throughout the paper, unless otherwise specified, we employ the linearized point-to-plane distance metric; for Figure 8 and Figure 9, where surface normals are not well defined, we employ the point-to-point metric. We will focus our comparisons on partial overlap and outliers in the data, which are the most challenging problems for the ICP algorithms.

**Rejecting vs. penalizing outliers.** Figure 1 shows how least-squares ICP registration \((p = 2)\) performs poorly unless appropriate heuristics for outlier management are adopted. Typical heuristics include rejecting pairs on boundaries, having distance above a given threshold or with dissimilar normals; see [RL01]. Outlier rejection approaches can be difficult to tune and even be harmful as they drastically increase the number of local minima. This is especially visible when the source is far from the target at the initialization. In this case, selecting a distance threshold too small will lead to the rejection of most of the correspondences, which in turn will increase the probability of converging to a bad local minimum. Selecting a distance threshold too big will not lead to a good final alignment as not enough outliers will be rejected. Conversely, our approach weakly penalizes outliers leading to a more stable approach. In Figure 4, we compare the alignment performance of our approach with outlier rejection techniques using a distance threshold. To evaluate the quality, we simulate the scan of a digital model, registr-
ter it, and evaluate the RMSE of the registered point locations w.r.t. the ground truth. As demonstrated by the values of RMSE in Figure 4 and in the supplemental video, our technique outperforms classical ICP with outliers rejection. Another approach for robust registration selects the best $k$-percent of the set of correspondences at each step of the registration [CSK05]. As illustrated in Figure 12, contrary to this approach, where the percentage of inliers $k$ needs to be known beforehand, our method automatically selects the set of inliers for the registration.

Iterative reweighting. Discarding unreliable correspondences is undoubtedly the simplest and most common way of dealing with outliers. Another common approach is to reweight correspondences using “robust functions” [MB93] resembling the weight function presented in Equation 11; see Figure 7. An important observation is that only the weight function of $p$-norms, amongst the ones shown, tend to infinity as we approach zero. Clearly this has immediate implications in terms of sparsity, as only $p$-norms will greatly reward values approaching zero. Unfortunately, as $p$-norm weights are defined in the $[0,\infty]$ range, both stability and convergence speed of ICP algorithms employing reweighting are affected due to the ill-conditioning of the associated linear system, as the weights vary in the range $[0,\infty]$.

The slow convergence of iterative reweighting in this case is due to the ill-conditioning of the associated linear system, as the weights vary in the range $[0,\infty]$. However, when using point-to-plane distances ADMM outperforms the iterative reweighting approach. The slow convergence of iterative reweighting in this case is due to the ill-conditioning of the associated linear system, as the weights vary in the range $[0,\infty]$.

Figure 10: Convergence of the reweighting approach compared to the ADMM method. When using point-to-point distances (top), iterative reweighting converges faster that ADMM. However, when using point-to-plane distances ADMM outperforms the iterative reweighting approach. The slow convergence of iterative reweighting in this case is due to the ill-conditioning of the associated linear system, as the weights vary in the range $[0,\infty]$.

Convergence speed. Figure 11-a illustrates the a trade-off between reducing $p$ to achieve robust registration and convergence speed. As $p$ is reduced, our registration becomes more resilient to outliers; however, the convergence speed decreases. This is mainly due to the fact that when $p$ decreases, the contribution of pairs of correspondences with large distances also decreases forcing the optimization to take smaller steps. Figure 11-b reports a comparison between our sparse implementation of point-to-point and point-to-plane for $p = 0.4$. Similarly to what was previously discovered for standard ICP [RL01,PHYH06], our optimization also greatly benefits from the extra degree of freedom (i.e. tangential motion) of the point-to-plane metric approximation.

Selecting “$p$”. Our experiments show how the performance of the algorithm varies with $p$, progressively trading-off between performance and robustness. Throughout our experiments, unless otherwise stated, we selected $p = 0.4$ as it seemed to offer a good trade-off.

Limitations. In our approach we essentially focused our efforts to robustify the second of the two ICP steps, i.e. Equa-
Figure 12: Contrary to TriCP [CSK05], where the percentage of inliers $k$ is a parameter of the algorithm, our approach automatically selects the set of inliers. As shown in this example, the set of inliers used at the last iteration of the registration (shown in black) is stable and accurate in our approach.

8. Conclusions

We presented an extension of the classical ICP algorithm that systematically addresses the problem of outliers commonly observed in acquired 3D data. We express the ICP registration problem as a sparse $\ell_p$ optimization, obtaining an heuristic-free, robust rigid registration algorithm having only one free parameter. We believe that our method can replace or extend most existing ICP implementations that often are a crucial component in numerous geometry processing applications.

Acknowledgements

The authors thank Bailin Deng, Rick Chartrand, Dror Aiger and the reviewers for their valuable comments. We would also like to thank Niloy J. Mitra, Mario Deuss, Yuliy Schwartzburg, Boris Neubert, Cheryl Lau for proof-reading the paper and providing valuable comments. The model in (Figure 2) was scanned by Thibaut Weise. The models of Figures 4, 9, 11, 12 can be downloaded at http://lgg.epfl.ch/statues. This work has been supported by Swiss National Science Foundation (SNSF) grant #20PA21L_129607.

References


Appendix A: Augmented Lagrangian Method (ALM)

We briefly discuss constrained optimization methods to provide a suitable background for the optimization approach taken in this paper. Consider the equality-constrained optimization problem having $x \in \mathbb{R}^n$:

$$\min_{x} f(x) \quad \text{subject to } c_i(x) = 0 \quad i = 1 \ldots n. \quad (23)$$

Given a vector of Lagrange multipliers $\lambda \in \mathbb{R}^n$ and $\mu \in \mathbb{R}^+$, we can transform the problem into an unconstrained optimization expressed by the augmented Lagrangian function

\[ L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{n} \lambda_i c_i(x) + \mu \sum_{i=1}^{n} \frac{1}{2} c_i(x)^2 \]
\[ \mathcal{L}_A(x, \lambda, \mu) = f(x) + \sum_{i=1}^{n} \lambda_i c_i(x) + \frac{\mu}{2} \sum_{i=1}^{n} c_i^2(x). \]  

(24)

A close inspection of this expression reveals that it is a combination of a linear penalty typical of dual ascent methods [PB13, Ch. 2] and a quadratic one from penalty methods [NW06, Ch. 17]

\[ \mathcal{L}(x, \lambda) = f(x) + \sum_{i=1}^{n} \lambda_i c_i(x) \quad Q(x, \mu) = f(x) + \frac{\mu}{2} \sum_{i=1}^{n} c_i^2(x). \]

Intuitively, dual ascent methods exploit the fact that the Lagrange dual function \( g(\lambda) = \min_x \mathcal{L}(x, \lambda) \) is a lower bound on the primal problem [BV04, pp. 216]. A gradient ascent w.r.t. \( \lambda \) of appropriate size gradually closes the duality gap, effectively optimizing Equation 23 as shown in Equation 26. Conversely, penalty methods gradually increase the penalty parameter \( \mu \), resulting in progressively increased constraint satisfaction. Unfortunately, both approaches suffer severe limitations; dual ascent methods place strong assumptions on the problem, like strict convexity and boundedness of \( g(\cdot) \); penalty methods offer limited control on constraint satisfaction, running into numerical issues as \( \mu \) is increased to very large values. These issues are addressed by the augmented Lagrangian approach; the optimization involves an algorithm strongly resembling dual ascent:

Step 1: \( x^{t+1} := \min_x \mathcal{L}(x, \lambda^t, \mu) \) \hspace{1cm} (25)

Step 2: \( \lambda_i^{t+1} := \lambda_i + \mu c_i(x^{t+1}) \quad i = 1 \ldots n \) \hspace{1cm} (26)

providing an asymptotic approximate constraint satisfaction in the form [NW06, Thm 17.2]

\[ c_i(x) \approx (\lambda_i^* - \lambda_i)/\mu. \]  

(27)

Consequently, constraints can be satisfied by either increasing \( \mu \), or alternatively by providing multipliers \( \lambda \) having values close to the optimal \( \lambda^* \); note that this is achieved by the Lagrange multiplier ascent step inherited from dual ascent; see Equation 26.

**Appendix B:** Alternating Direction Method of Multipliers

The ADMM method is an extension of the augmented Lagrangian method to optimize a compound problem \( f(x) = f(x) + f(\hat{x}) \) by decoupling it into simpler subproblems [BPC*11]. Under appropriate conditions [PB13, Ch. 3], as the optimizer is separated in \( x = [\hat{x}, \tilde{x}] \), the problem can be approached by iteratively solving the following three steps:

Step 1: \( \hat{x}^{t+1} := \min_{\hat{x}} \mathcal{L}([\hat{x}, \tilde{x}], \lambda^t, \mu) \) \hspace{1cm} (28)

Step 2: \( \tilde{x}^{t+1} := \min_{\tilde{x}} \mathcal{L}([\hat{x}^{t+1}, \tilde{x}], \lambda^t, \mu) \) \hspace{1cm} (29)

Step 3: \( \lambda_i^{t+1} := \lambda_i^t + \mu c_i(x^{t+1}) \quad i = 1 \ldots n \) \hspace{1cm} (30)

Note that alternately looping over Steps 1/2 until convergence would simply correspond to a block coordinate descent decomposition of the joint minimization in Equation 25. Consequently, ADMM can be interpreted as applying a single step of block coordinate descent optimization applied to the augmented Lagrangian problem.

**Appendix C:** Shrink operator for \( \mathcal{f}(z) = \|x\|_2^p + \frac{\mu}{2} \|z - h\|_2^2 \)

The minimization of \( \mathcal{f}(z) \) can be simplified into a scalar problem by noticing that \( z \) can be expressed as \( z = \alpha h \) (see Appendix D) leading to

\[ \min_{\alpha} \|\alpha h\|_2^p + \frac{\mu}{2} \|\alpha h - h\|_2^2 = \min_{\alpha} \|h\|_2^{p-2} |\alpha|^p + \frac{\mu}{2} |\alpha - 1|^2. \]  

(31)

As proven in [MS12], the optimal \( \alpha^* \) is given by:

\[ \alpha^* = \begin{cases} 0 & \text{if } |\hat{h}| < \hat{h} \\ \beta & \text{if } |\hat{h}| > \hat{h} \end{cases} \]  

(32)

where the threshold \( \hat{h} \) is computed as

\[ \hat{h} = \alpha_u + \frac{\mu}{p} \alpha_u^{p-1}, \quad \alpha_u = \left( \frac{\mu}{p} (1 - p) \right)^{\frac{1}{p-1}}. \]  

(33)

\( \beta \) is found by using the following update scheme

\[ \beta_{t+1} = 1 - \frac{\mu}{p} \|h\|_2^{p-2} \beta_t^{p-1} \]  

(34)

by initializing \( \beta_0 \in [\alpha_u |h|_2^{-1}, 1] \). Similarly to [MS12], we noticed that this scheme converges in two or three iterations. The optimal \( \hat{z} \) is then given by \( \hat{z}^* = \alpha^* h \).

**Appendix D:** Scalar version of \( \mathcal{f}(z) = \|x\|_2^p + \frac{\mu}{2} \|z - h\|_2^2 \)

The minimization of \( \mathcal{f}(z) \) can be re-interpreted into a simpler scalar problem where the optimal solution can be expressed as \( \mathcal{z}^* = \alpha^* h, \alpha \in \mathbb{R} \). We verify this by expressing \( z \) as a linear combination of a component of the space defined by \( h \) and \( \kappa \) orthogonal to it, that is, \( \mathcal{z} = \mathcal{h} = 0 \). All that is necessary is to prove that \( \forall \kappa \in \mathbb{R}^3, \mathcal{f}(\alpha \mathcal{h} + \kappa) \geq \mathcal{f}(\alpha \mathcal{h}) \). This can be verified by checking that both of the two following inequalities \( \|\alpha \mathcal{h} + \kappa\|_2^p \geq |\alpha | \|\mathcal{h}\|_2^p \) and \( \|\alpha \mathcal{h} + \kappa - h\|_2^p \geq |\alpha | \|h\|_2^p \) hold. We can easily convert the first of these into one involving quadratic exponents only. We first raise both sides to the power \( 2/p \) and then expand the norm as an inner product obtaining \( \|\alpha \mathcal{h} + \kappa\|_2^p \geq |\alpha | \|\mathcal{h}\|_2^p \). As \( \mathcal{h} \) and \( \kappa \) are orthogonal we get \( \|\alpha \mathcal{h}\|_2^p \geq |\alpha | \|\mathcal{h}\|_2^p \); this is always verified as \( \|\mathcal{h}\|_2^p \geq 0 \). To verify the second expression, we again expand the norm as an inner product, then, after removing \( \|\cdot\|_2^p \) from both sides, we obtain \( \|\alpha \mathcal{h} + \kappa\|_2^p \geq \|\alpha \mathcal{h}\|_2^p \). Similarly to what we did before, we expand the norm and exploit orthogonality, simplifying the expression to \( \|\mathcal{h}\|_2^p \geq 0, \) that, again, is always verified.

© 2013 The Author(s)
© 2013 The Eurographics Association and Blackwell Publishing Ltd.