Linear Object Registration of Interventional Tools

Matthew S. Holden and Gabor Fichtinger

Laboratory for Percutaneous Surgery, School of Computing, Queen's University, Kingston, ON, Canada
{mholden8,gabor}@cs.queensu.ca

Abstract. PURPOSE: Point-set registration for interventional tools requires well-defined points to be present on these tools. In this work, an algorithm is proposed which uses planes, lines, and points for registration when point-set registration is not feasible. METHODS: The proposed algorithm matches points, lines, and planes in each coordinate system, uses invariant features for initial registration, and optimizes the registration iteratively. For validation, simulated data with known ground-truth and real surgical tool registration data using point-set registration as ground-truth were created to evaluate the algorithm's accuracy. RESULTS: The proposed algorithm is equally as accurate as point-set registration, and the difference between the registrations is less than the noise in the tracking system. CONCLUSION: The proposed algorithm is a viable alternative when point-set registration cannot be performed.

Keywords: Registration, Surgical navigation, Coordinate transformations.

1 Introduction and Background

Introduction. Registration is an integral part of all surgical navigation systems, and is used to display interventional tools (including needles, ultrasound probes, phantoms, etc.) and images from multiple modalities in a common navigation space. For example, in surgical simulators with augmented reality, instruments and phantoms must be registered to a common coordinate frame to view them in the augmented reality display [14]. Alternatively, in surgical planning, the plan (and possibly preoperative planning image) must be registered into the same coordinate system as the interventional tools or robots for the procedure [8].

Typical computer-assisted interventions use point-set registration using landmark points identified in both coordinate frames with known correspondence. In most cases, however, landmark points do not naturally occur on physical objects such as interventional tools, robotic devices, or calibration fixtures, and estimating landmarks point positions is insufficiently accurate. Due to engineering constraints, it is more common that an object will have a set of well-defined lines or planes, which could be conveniently and accurately collected and used for registration instead (see, for example, [12] and [15] for instances of such interventional tools).

The objective of this work is to develop and validate a registration algorithm that uses points, lines, and planes (collectively referred to as "linear objects") for registration problems such as those encountered in surgical simulation [14] and planning [8]. The algorithm is designed to provide a convenient alternative when

C.A. Linte et al. (Eds.): AE-CAI 2014, LNCS 8678, pp. 118–127, 2014.

point-set registration is infeasible. Such an algorithm should work for any set of linear objects that uniquely defines the registration and should be guaranteed to converge an approximately optimal solution in polynomial time. Additionally, the algorithm should work even when one set of linear objects is a permuted subset of the other.

Although the proposed algorithm is validated by registering man-made objects with linear features that can be localized with a pointing device, the algorithm has further applications in image registration, which could be performed using linear features extracted from images of interventional tools (which has previously been shown to be a fruitful endeavour [10]).

Background. Many point-set registration methods have been proposed for points with unknown correspondence based on Besl and McKay's [1] iterative closest point algorithm. Several methods proposed to improve convergence to the global optimum include: using symmetry to achieve a better initial guess [6], "Lipschitzizing" the error function [9], or using a Levenberg-Marquardt method [4]. Regardless, a global optimum can only be guaranteed with great computational expense.

Alternative approaches use invariant features in both coordinate frames to find the correspondences and then apply a closed-form solution. Thirion [13] used the extreme points of the dataset to determine correspondences. Xiao *et al.* [16] matched surface properties of local point clouds to determine correspondences. Similar feature extraction techniques are also used in many image registration algorithms. Unfortunately, most of these methods rely on having a complete set of points collected in both coordinate frames, which is unavailable in many situations.

Several works provide algorithms which satisfy some criteria for solving the problem outlined above. Jain *et al.* [7] use points, lines, and ellipses for registration of C-arm images. Their work, however, uses specific features of C-arm imaging and is not guaranteed to converge. Lee *et al.* [8] also perform image registration using fiducial lines and their cross-sectional images. Their work does not consider the case of fiducial planes or points, and thus, does not apply in all scenarios examined here.

The work of Meyer *et al.* [10] used a combination of points, lines, and planes (i.e. linear objects) with known correspondence for image registration. For each linear object, they calculate the projection of the centroid onto it and the direction vector from the projection to the centroid. The coordinate frames are registered using these points with known correspondence. Our proposed algorithm improves upon their algorithm by automatically determining linear object correspondence, not requiring points to define the centroid, and offering an iterative method for convergence.

Olsson *et al.* [11] provide a branch and bound method for registering a set of points to their corresponding points, lines, and planes. They provide an algorithm which is guaranteed to converge to the optimal solution for any configuration. Though our proposed algorithm only guarantees convergence to a near globally optimal solution, it has the advantage of being polynomial time, rather than exponential time.

Overall, the proposed algorithm extends previous work [10] by providing automatic linear object matching, solutions to a wider range of configurations, and offering an iterative convergence method. In contrast to [11], the proposed algorithm runs in polynomial time. Furthermore, an open-source implementation of the proposed algorithm is provided through the SlicerIGT (www.slicerigt.org) extension for 3D slicer (www.slicer.org), thus providing an accessible medical registration tool.

2 Methods

2.1 Linear Object Registration Algorithm (LORA)

Point-to-Point, Line-to-Line, Plane-to-Plane Registration. This represents the most important component of the proposed algorithm: a method for simultaneously registering points to points, lines to lines, and planes to planes. The solution is guaranteed to be approximately optimal and the algorithm runs in polynomial time.

To this end, the centroid of a set of linear objects $\overline{X}$ is defined as the point for which the sum of squared distances to all linear objects X is minimized.

$$\overline{X} = \underset{\bar{x} \in R^3}{\arg\min} \left[\sum_{x \in X} D^2(\bar{x}, x) \right] \tag{1}$$

In practice, the centroid can be calculated by finding the least-squares solution $\overline{X}$ to the below equation, where each linear object has some point $\vec{b}$ that lies on it and some orthonormal basis $\vec{n}$ to its orthogonal subspace.

$$\begin{bmatrix} \vec{n}_1{}^T \\ \dots \\ \vec{n}_m{}^T \end{bmatrix} \overline{X} = \begin{bmatrix} \vec{n}_1{}^T \vec{b}_1 \\ \dots \\ \vec{n}_m{}^T \vec{b}_m \end{bmatrix} \tag{2}$$

The method for simultaneously registering points to points, lines to lines, and planes to planes follows as below, assuming registration of linear objects in coordinate frame A to linear objects in coordinate frame B.

1. Calculate the linear object centroid for linear objects in frame A and the linear object centroid for linear objects in frame B.
2. In both coordinate frames, translate the linear objects such that the centroid lies at the origin of the coordinate frame.
3. For each coordinate frame, respectively, denote $\vec{y}_A$ and $\vec{y}_B$ as the set of closest points on each linear object to the origin.
4. For each line and plane, add the direction or normal vector to the set $\vec{y}_A$ or $\vec{y}_B$ as applicable.
5. Calculate the point-set registration between the sets $\vec{y}_A$ and $\vec{y}_B$.

Linear Object Matching. Automatically determining correspondence is an important component of the proposed algorithm in terms of usability. Ideally, intrinsic features that are invariant between the two sets of linear objects could be used. By considering the counterexample of registering three equidistant points (for which the registration is unique given correspondence), however, it can readily be seen that any matching will produce a solution with the same fiducial registration error. Thus, some external feature must be used to determine linear object correspondence.

The chosen external feature is a set of fiducials which have invariance between the two coordinate frames, which shall be called “references”. These are not used for registration directly, but are used to determine the correspondence between linear objects. These references need not be collected with high accuracy, however, since they are not used directly for registration. A linear object’s signature is defined as the vector of distances to the set of references $R = \{r_1,...,r_K\}$.

$$signature(L) = [D(L, r_1),..., D(L, r_K)] \tag{3}$$

Linear objects in the two coordinate frames can be matched by comparing their signatures. Linear objects whose signatures cannot be matched (to within some threshold) are discarded. The matching threshold is calculated as the product of the noise associated with collecting the linear objects and the number of references.

Reconstructing Linear Objects from Point Sets. In practice, linear objects may be collected discretely. Thus, rather than points, lines, and planes, one set of linear objects may look like a set of points clustered about points, lying on lines, and/or lying on planes (see Fig. 1). It is assumed that point-sets are delineated such that each subset describes precisely one linear object. Then, the principal component analysis for each subset may be calculated, where the eigenvectors associated with non-zero eigenvalues represent direction vectors on the linear object. These derived linear objects may be subsequently used for linear object registration.

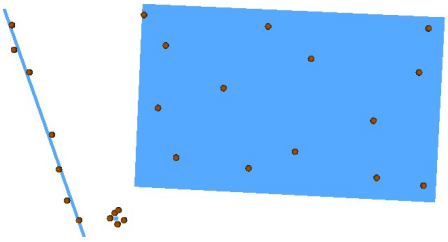

Fig. 1. Illustration of a point, line, and plane (blue) reconstructed from a set of points (red)

Moreover, these initial points used to extract the linear objects may be used to improve the registration result. Given an initial registration and known correspondence, the registration between a set of linear objects in coordinate frame A, and a set of points $b = \{b_1,...,b_n\}$ in frame B can be calculated as follows.

1. Transform each point b_i in coordinate frame B by the current transform T.
2. For each transformed point Tb_i, find the closest point a_i on its corresponding linear object in coordinate frame A.
3. Shift the sets a_i and b_i such that their centroids $\bar{a}$ and $\bar{b}$ lie at the origin.
4. Calculate the pure rotational registration R between a_i and b_i.
5. Recalculate the current transform. Its rotation is R and its translation is $\bar{a} - R\bar{b}$.
6. Iterate until some convergence criteria is met (for example, the change in fiducial registration error is below some threshold).

Linear Object Registration Algorithm. In summary, LORA proceeds as follows.

1. Reconstruct linear objects from collected points, as applicable.
2. Compute linear object correspondences using references.
3. Perform point-to-point, line-to-line, and plane-to-plane registration.
4. Use the result from step 3 as an initial transform in an iterative point-set to linear object registration.

Step 3, being the key step in the algorithm, offers a closed-form solution to the point-to-point, line-to-line, and plane-to-plane registration. It produces an exact solution to a finite version of the linear object registration problem which is globally optimal for the inifinite version in the case of no noise. This means that in practical cases where there is noise, step 3 produces an approximate solution. Step 4 is guaranteed to converge to a local minimum (by extension of the proof of convergence from [1]), so the entire solution is guaranteed to be approximate.

2.2 Algorithm Validation

Simulated Data. As an initial form of validation, LORA's feasibility was tested on simulated data. The objective of the simulation was to generate random linear objects and to create random points on these linear objects, transformed by a known transformation matrix. Since the ground-truth transformation is known, the algorithm's accuracy can be readily evaluated.

A random transformation matrix can be generated by generating a random rotation and a random translation separately and combining the results. To generate a random rotation R, any matrix M with randomly generated elements and its singular value decomposition may be used.

$$M = UDV^T \tag{4}$$

$$R = UV^T \tag{5}$$

The translation $\vec{d}$ is generated by picking each component randomly.

1. Generate a random number of each type of linear object (with random position and orientation) in coordinate frame A. Generate four references with random position in coordinate frame A.
2. For each linear object L in coordinate frame A, generate a set of random points $a = \{a_1, ..., a_n\}$ lying on L.
3. Generate a random transformation matrix T (by above described method).
4. Apply the random transformation matrix to each random point Ta_i, and add Gaussian noise in each dimension.

This algorithm generates random linear objects in coordinate frame A and simulates point collection in frame B on the linear objects. Thus, this simulated data can be used to test the feasibilty of LORA by comparing its result to the ground-truth.

Real Data. As interventional tools in the general sense, three previously developed surgical navigation phantoms were used for validation of LORA: an fCal ultrasound calibration phantom (Fig. 2a) [3], a lumbar spine phantom (Fig. 2b) [14], and a LEGO® ultrasound calibration phantom (Fig. 2c) [15]. Though these are not traditional interventional tools, the validation results apply equally to registration of tracked tools such as ultrasound probes or scalpels. The objective was to find the transformation between the phantoms' sensor's coordinate frame and the navigation system coordinate frame. This form of registration is required in surgical simulators to display all the objects in a common navigation space [14].

For each phantom, the set of linear objects in the phantom coordinate frame was defined. Both ultrasound calibration phantoms had a box-like exterior. Each face of this box was defined as a linear object. The lumbar spine phantom sits on a rectangular prism base. Each face of the base was defined as a linear object. Also, the vertebrae are mounted on a block that is attached to the base. The lines where this block attaches to the base were defined as linear objects.

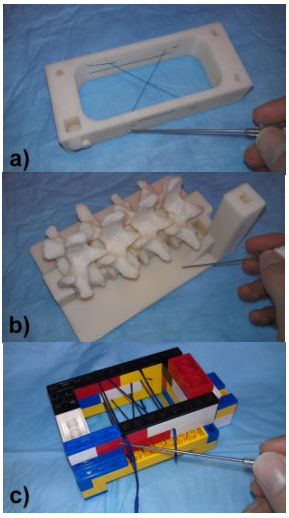

Fig. 2. Photographs of a user collecting points on a) fCal [3], b) lumbar spine [14], and c) LEGO® [15] phantoms

The fCal and lumbar spine phantoms already had points machined on them for point-set registration purposes. Using these points, point-set registration for each phantom was performed and used as ground-truth against which LORA was compared. The LEGO® phantom did not have points machined on it, so LORA was validated by comparing target registration errors and point reconstruction accuracies [2] from point-set registration (using approximate fiducial positions) and LORA.

For validation, all linear objects defined on each phantom were used, and two reference points were chosen from the set of points used for point-set registration. A 0.9mm stylus, tracked using the Ascension TrakStar electromagnetic tracking system (www.ascension-tech.com), was used to collect points on each linear object. The stylus was placed with its tip at each point, the stylus tip was traced back and forth along each line, and the stylus tip was slid over the entire extent of each plane. All data was collected, annotated, and saved using 3D Slicer (www.slicer.org).

3 Results

To compare two transformations, the translational error metric was calculated as the norm of the difference between the two translations:

$$E_{TRANSLATION} = \left|\vec{d}_1 - \vec{d}_2\right| \tag{6}$$

The rotational error metric was calculated as the angle (from the axis-angle representation) of rotation of the quotient of the two rotation matrices:

$$E_{ROTATION} = \arccos\left(\left(tr\left(R_1^{-1}R_2\right)-1\right)/2\right) \quad (7)$$

Simulated Data. To demonstrate the robustness of LORA in simulation, the average translational and rotational error over fifty trials associated with the registration (compared to the ground-truth) is plotted for varying levels of Gaussian noise (Fig. 3). Most importantly, the error increases linearly with the noise, demonstrating the robustness of LORA.

The Ascension TrakStar electromagnetic tracking system (www.ascension-tech.com) reports 1.4mm root-mean-square tracking accuracy. Using this level of simulated noise, the algorithm exhibits 0.085° rotational error and 0.21mm translational error.

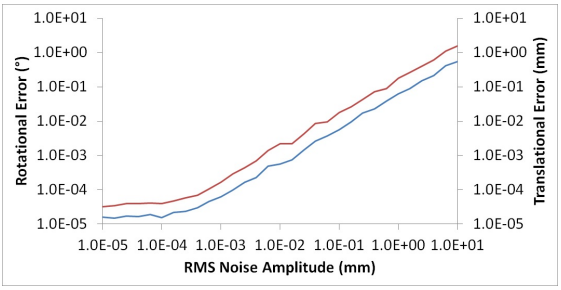

Fig. 3. Plot of translational error (red) and rotational error (blue) in the calculated transformation as a function of root-mean-square noise in the simulated data

To demonstrate the required number of references, simulated data was generated using varying number of references. The linear object matching was successful over 90% of the time when there was one reference. For two or more references, matching was successful in every simulation trial (up to 10.0mm of root-mean-square noise), indicating that two references are sufficient for practical linear object registration.

Real Data. For all three phantoms, the linear object registration had smaller average root-mean-square error than the traditional point-set registration (1.22mm vs. 2.13mm for the fCal ultrasound calibration phantom, 1.14mm vs. 1.33mm for the lumbar spine phantom, and 0.45mm vs. 0.53mm for the LEGO® phantom).

The error between the ground-truth (calculated as the mean transform from the point-set registration) and the transformation calculated using LORA is displayed in Table 1. The variability in the results produced by each algorithm is shown in Table 2. This provides a measure of each algorithm's precision.

Translational variability for the lumbar spine phantom was the only significantly different reported variability (two-tailed t-test, $p = 0.003$) between the two algorithms.

Table 1. Error metrics for linear object registration using the fCal and lumbar spine phantoms. Error is calculated as the mean difference between the linear object registrations and the mean point-set registration. The rotational and translational errors are averaged over all registrations.

Metric	fCal Phantom	Lumbar Spine Phantom
Rotational Error (°)	1.49	0.76
Translational Error (mm)	0.74	1.15

The target registration errors and point reconstruction accuracies for the LEGO® phantom are shown for each algorithm in Table 3. The point reconstruction accuracies

are so large due to noise in the electromagnetic tracking system; however, LORA still produces significantly better accuracies than point-set registration. Both metrics are significantly smaller for LORA (by two-tailed *t*-test), with medium-large effect size using Cohen's d statistic ($p = 0.005$, effect size 0.57 for target registration error; $p < 0.001$, effect size 0.68 for point reconstruction accuracy).

Table 2. Rotational and translational precisions for each registration algorithm for the fCal and lumbar spine phantoms. The precision is calculated as the mean difference between each registration and the mean registration.

fCal Phantom		
Metric	Point-Set Registration	Linear Object Registration
Rotational Precision (°)	0.46	0.43
Translational Precision (mm)	0.45	0.37
Lumbar Spine Phantom		
Metric	Point-Set Registration	Linear Object Registration
Rotational Precision (°)	0.29	0.42
Translational Precision (mm)	0.35	0.76

Table 3. Target registration error and point reconstruction accuracy for point-set registation and LORA on the LEGO® ultrasound calibration phantom

Target Registration Error		
Metric	Point-Set Registration	Linear Object Registration
Mean (mm)	1.34	1.18
Standard Deviation (mm)	0.31	0.22
Point Reconstruction Accuracy		
Metric	Point-Set Registration	Linear Object Registration
Mean (mm)	3.98	3.46
Standard Deviation (mm)	0.64	0.80

For our current Matlab implementation, LORA took on average 34s with 9,816 collected points for the fCal phantom, 71s with 11,969 collected points for the lumbar spine phantom, and 57s with 10,380 collected points for the LEGO® phantom. For surgical simulation and planning, registrations are typically performed offline; thus, this temporal performance is adequate for practical use. In all instances, point-set registration took less than 1s to complete, but uses fewer than ten points. These performance results contrast with results from Olsson *et al.* [11], which took up to 30s with fewer than twenty collected points. With our dataset, which contains up to 10,000 collected points per registration, their algorithm could take impractically long.

4 Discussion and Conclusion

Fitzpatrick *et al.* [5] prove that target registration error decreases with distance to the centroid and with larger fiducial configurations. This implies that linear objects should encompass relevant structures, and points should be collected on the entirety

of linear objects. For example, for both ultrasound calibration phantoms, the linear objects should encompass the calibration wires, and for the lumbar spine phantom, the linear objects should encompass the intervertebral spaces. Linear objects are usually completely defined by the surgical tool itself, however, and should not need to be modified for registration. The results from the simulated data using random geometry suggest that LORA is robust to the arrangement of linear objects.

Additionally, Fitzpatrick *et al.* [5], show that target registration error also decreases as more points are collected for registration. Thus, the number and range of collected points on each linear object should be maximized to improve the accuracy of LORA.

Some sets of linear objects are insufficient to specify the transformation between the tool sensor coordinate frame and the surgical tool coordinate frame completely. Importantly, LORA has the property that it will calculate the transformation between the two coordinate frames for any sufficient set of linear objects.

One less robust aspect of LORA is matching. Although simulated results show that two references is sufficient, it can be inconvenient to collect many references. One possibility is to enforce that the user manually match the linear objects. Alternatively, geometrical constraints on the defined references and linear objects could be enforced to ensure unambiguous matching.

The results reported here are strictly from surgical tool registration applications, however, the algorithm extends to related applications, for which limited further validation is required. Other applications for LORA include image registration. LORA offers a general method to register imaged points, lines, or planes with interventional tools including robots, or Z-frame and N-wire phantoms, without requiring development of a new registration algorithm. Validation of the proposed algorithm for surgical planing and image registration is a clinical setting is required.

Of course, not all interventional tools consist of well-defined linear objects. LORA could be extended to registration of any parametrically defined surfaces or curves. This, however, may be problematic since parametrically defined objects do not necessarily have the properties of linear objects used by LORA.

The proposed algorithm has been made available through the open-source SlicerIGT (www.slicerigt.org) extension for 3D Slicer (www.slicer.org). This provides a convenient interface for users to collect linear objects and use LORA. Further usability and temporal performance studies for the module are planned.

In conclusion, an algorithm (LORA) for surgical tool registration using linear objects has been developed, implemented, and validated. The algorithm does not impose the constraint of well-defined points that point-set registration algorithms impose. The algorithm works on a principle of extracting corresponding points from the linear objects in the two coordinate frames using geometric invariants and performing point-set registration on these, following by an iterative algorithm to converge to an optimum, which is guaranteed to be close to the global optimum. This algorithm was validated on simulated data by showing that the error in the registration increases linearly with noise in the data. The algorithm was validated using three different surgical navigation objects. Results showed that LORA performs practically identically to ground-truth point-set registration and it demonstrates more favorable registration error metrics while it is significantly more convenient to apply in practice.

References

1. Besl, P.J., McKay, N.D.: A Method for Registration of 3D Shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 14, 239–256 (1992)
2. Carbajal, G., et al.: Improving N-wire phantom-based freehand ultrasound calibration. International Journal of Computer Assisted Radiology and Surgery 8, 1063–1072 (2013)
3. Chen, T.K., et al.: A Real-Time Freehand Ultrasound Calibration System with Automatic Accuracy Feedback and Control. Ultrasound in Medicine & Biology 35(1), 79–93 (2009)
4. Fitzgibbon, A.W.: Robust Registration of 2D and 3D Point Sets. Image and Vision Computing 21, 1145–1153 (2003)
5. Fitzpatrick, J.M., West, J.B., Maurer, C.R.: Predicting Error in Rigid-Body Registration. IEEE Transactions on Medical Imaging 17, 694–702 (1998)
6. Foroughi, P., Taylor, R., Fichtinger, G.: Revisited Initialization for 3D Bone Registration. In: Proceedings of SPIE (2008)
7. Jain, A.K., et al.: FTRAC - A robust fluoroscopic tracking fiducial. Medical Physics 32(10), 3185–3198 (2005)
8. Lee, S., Fichtinger, G., Chirikjian, G.S.: Numerical algorithms for spatial registration of line fiducial from cross-sectional images. Medical Physics 29(8), 1881–1891 (2002)
9. Li, H., Hartley, R.: The 3D-3D Registration Problem Revisited. In: IEEE International Conference on Computer Vision (2007)
10. Meyer, C.R., et al.: Simultaneous Usage of Homologous Points, Lines, and Planes for Optimal 3D, Linear Registration of Multimodality Imaging Data. IEEE Transactions on Medical Imaging 14, 1–11 (1995)
11. Olsson, C., Kahl, F., Oskarsson, M.: Branch-and-Bound Methods for Euclidean Registration Problems. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(5), 783–794 (2009)
12. Shamir, R., Freiman, M., Joskowicz, L., Shoham, M., Zehavi, E., Shoshan, Y.: Robot-Assisted Image-Guided Targeting for Minimally Invasive Neurosurgery: Planning, Registration, and In-vitro Experiment. In: Duncan, J.S., Gerig, G., et al. (eds.) MICCAI 2005. LNCS, vol. 3750, pp. 131–138. Springer, Heidelberg (2005)
13. Thirion, J.P.: Extremal Points: Definition and Application to 3D Image Registration. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1994)
14. Ungi, T., et al.: Perk Tutor: An open-source training platform for ultrasound-guided needle insertions. IEEE Transactions on Biomedical Engineering 59(12), 3475–3481 (2012)
15. Walsh, R., et al.: Design of a tracked ultrasound calibration made of LEGO bricks. In: SPIE Medical Imaging (2014)
16. Xiao, G., Ong, S.H., Foong, K.W.C.: Efficient Partial-Surface Registration for 3D Objects. Computer Vision and Image Understanding 98, 271–293 (2005)