Range Image Registration via Probability Field

Haitao Zhang* Olaf Hall-Holt† Arie Kaufman*
Center for Visual Computing (CVC)
*Computer Science Department
†Department of Applied Mathematics and Statistics
Stony Brook University
Stony Brook, NY 11794-4400, USA

Abstract

This paper presents a powerful variant of the ICP (Iterative Closest Point) algorithm for registering range images using a probability field. The probability field (p-field) represents the probability distribution of the surface position. By capitalizing on the properties of the range image, fast construction, compact representation, and efficient query of the p-field can be achieved. Different sensor models are supported by the p-field according to the properties of the range image. Range images can be precisely aligned by maximizing the probability of overlapping surfaces via the p-field.

Keywords: Probability Field, Range Image, ICP, Image Alignment, Registration.

1 Introduction

With the advent of 3D scanning technologies, building 3D geometry models from real objects has become an important data acquisition method. After acquiring a set of range images from different viewpoints and/or view directions, finding accurate spatial relationships among them is a crucial step before reconstructing the model. The ICP (Iterative Closest Point) algorithm has been used for range image registration, which has many variants [1, 2, 3, 5, 6, 7, 10, 12, 13], differing in point pair generation, error metrics, and minimization methods. Given an initial transformation between two range images, ICP tries to improve it through iteration. For a set of sample points in one range image, a set of nearby corresponding points are found on the mesh of the other range image along certain directions. An error metric is defined on these point pairs, usually representing the total weighted distances between the pairs. A rigid transformation is refined through minimizing the metric, either by closed form or iterative solution. This sequence of selecting point pairs and minimizing an error metric is repeated until some termination criterion is reached.

Chen and Medioni [3] created a set of corresponding points by finding the intersections of the surface with the normal lines and an iterative approach was used to minimize the error metric. Besl and Mckay [1] used a quaternion-based closed form solution for minimization. Rutishauser et al. [10] and Turk and Levoy [12] generated pairs by finding the closest points on the interpolated surface. Blais and Levine [2] used reverse calibration to find the corresponding points. The error metric for these ICP variants is the sum of the weighted Euclidean distances of the point pairs. Johnson and Kang [5] defined their error metric on both geometric and color information. They also used an occupancy grid for reconstruction. Rademacher [7] utilized the consistency of empty space in range images for ICP alignment. Rusinkiewicz et al. [8] used a real-time ICP variant by assuming that the relative motion between two consecutive range images in the acquisition is small. Eggert et al. [4] and Rusinkiewicz and Levoy [9] reviewed and compared existing ICP variants.

In this paper, we present a new ICP variant, which registers range images using a probability field. A probability field (p-field) is a 3D data structure similar to a distance field. Unlike the distance field that represents the distance to the surface, the value in the p-field represents the probability that the surface goes through that position. Another similar concept is an occupancy grid [11]. For an occupancy grid, the space is represented by a regular grid and each cell has a certainty value for the likelihood of its occupancy. Occupancy grids are often used directly in robotic planning and navigation to determine if two scenes are equivalent in terms of shape. Instead of representing the likelihood of 3D spatial occupancy, the p-field represents the probability of surface spatial position.

In our registration method (Section 2), a p-field is con-

*Email: {haitao|ari}@cs.sunysb.edu
†Email: olaf@ams.sunysb.edu

structed from one range image, an error metric is defined by the sample points of the other range image to be aligned, and the corresponding probabilities queried from the p-field. Unlike most ICP variants, which generate point pairs in every iteration and the error metric is defined by these newly generated pairs, our p-field method does not have explicit point pairs. The point pair generation is implicitly incorporated into the error metric by querying the p-field at the positions of the sample points. The p-field is constructed only once and used throughout the algorithm. Our error metric is a function of the p-field from one range image, the sample points of the other range image, and the rigid transformation between them. After minimizing the error metric, we get the optimal transformation and no iteration is needed for the whole process.

Our p-field representation (Section 3) is compact without any accuracy loss and it supports efficient querying. The p-field method is flexible and efficient in dealing with different kinds of range images and different registration strategies, such as supporting multiple sensor error models (Section 4), empty space consistency (Section 5), and hierarchical alignment (Section 6). Good alignment results (Section 7) are generated through our p-field method.

2 P-field Method for Registration

A p-field is the surface location distribution in 3D space. Given a 3D point v, the corresponding value in the p-field $P(v)$ represents the likelihood that the surface would go through v. If we know exactly where the surface is, $P(v)$ equals to either 1 or 0. When the surface is represented by a mesh from a range image, there is an inevitable error in the surface position. $P(v)$ is a local maximum when v is on the mesh, and decreases to 0 when v moves away from the mesh for a certain distance.

Given two partially-overlapped range images I_1, I_2, and an initial transformation T_0 between them, we want to find an accurate transformation T via a probability field. Here is the basic algorithm: After constructing the probability field P from I_1, compute a transformation T that maximizes the formula (Equation 1) representing the fitting of the overlapping parts:

$$F(T) = \sum_{i=1}^{N} c(v_i) P(T \cdot v_i) \qquad (1)$$

v_i is a sample point in I_2, $c(v_i)$ is the confidence of v_i, and N is the number of sample points in I_2. Alternatively, we can rewrite Equation 1 as an error metric to be minimized:

$$E(T) = \sum_{i=1}^{N} c(v_i)(1 - P(T \cdot v_i)) \qquad (2)$$

Since it is hard to find a closed form solution for minimizing the error metric defined on the p-field, we use the downhill simplex method first, followed by a gradient-based minimization method, to minimize the error metric starting from T_0. The downhill simplex method is less likely to get stuck in local minima by searching near a simplex in the 6-D transformation space. The gradient-based method is used to refine the result from the simplex method, which is very likely to be near the global minimum.

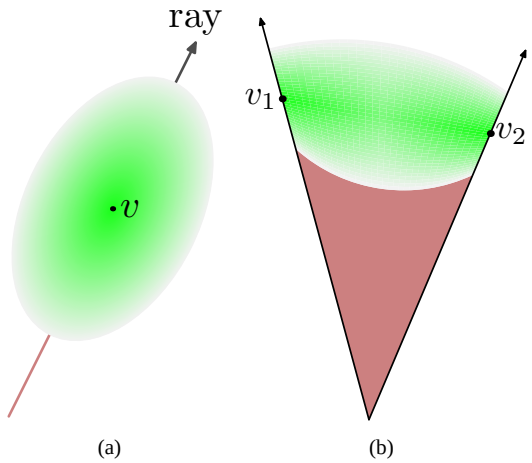

Figure 1. Sensor error model

3 P-field Construction and Representation

The p-field is a 3D data structure representing the spatial surface probability. The p-field can be generated from a range image. A range image is a set of 3D sample points, which is computed from the distance values acquired by a range scanner along the rays shooting from a 2D regular lattice. The sensor error model (Figure 1(a)) is approximated by an ellipsoidal Gaussian distribution along the ray direction centered at the sample point v. The length of the axis is determined by the confidence of the sample point. Assuming the confidence of the sample point v is proportional to the cosine value of the angle between the surface normal $\vec{n}(v)$ and the ray direction $\vec{r}(v)$, and given a sensor error range (e_{min}, e_{max}), the standard deviation σ of the Gaussian at v is defined by:

$$\sigma(v) = e_{min} + \cos(\vec{n}(v), \vec{r}(v))(e_{max} - e_{min}) \qquad (3)$$

If the distance between two neighboring sample points v_1 and v_2 is within a threshold, we assume there is a continuous surface between them. The surface probability near the sample points is greater than the probability between sample points. Assuming a 1D Gaussian distribution along any ray direction that intersects the line segment $[v_1, v_2]$ at $v(k, v_1, v_2)$, the σ of this Gaussian becomes larger when the ray is away from the sample points (Figure 1(b)). A modu-

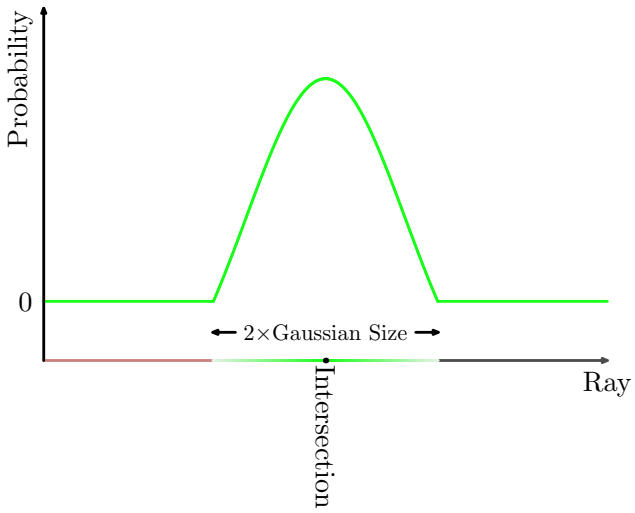

Figure 2. Surface probability along a ray.

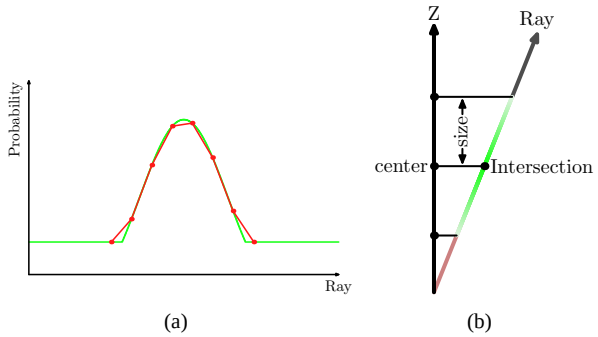

Figure 3. (a). Discrete sampled Gaussian shown in red and continuous Gaussian shown in green. The maximum position has been shifted under discrete sampling. (b). Sample ray is represented by <center,size>.

lation function $M(k, v_1, v_2)$ is used to reflect the σ change:

$$M(k, v_1, v_2) = 1 + \mathrm{C} \cdot dist(v_1, v_2)(1 - 2|k - 0.5|)^2 \quad (4)$$

where C is a coefficient and k is the parameter in the parametric representation of the line segment $[v_1, v_2]$:

$$v(k, v_1, v_2) = k\, v_1 + (1 - k)\, v_2 \quad (5)$$

Thus, the σ of the Gaussian is the interpolation of the values at v_1 and v_2 modulated by the modulation function:

$$\sigma(k, v_1, v_2) = M(k, v_1, v_2)(k\sigma(v_1) + (1 - k)\sigma(v_2)) \quad (6)$$

Creating a p-field from a range image is straightforward. First, segment the bounding box of the sample points into a 3D grid at a certain resolution. Then, each cell in the 3D grid is assigned a probability value for the likelihood that the surface goes through this cell. If the ray that goes through the center of the cell intersects with the surface defined by the range image, the intersection point and the corresponding Gaussian size along the ray can be computed. Thus, the probability of the cell can be determined (Figure 2).

This straightforward method is not efficient since it computes the probability of every cell in the 3D grid separately. Furthermore, because only a small fraction of the cells have non-zero probability (about 2% to 6% in most cases we have tested), it is possible to have a data structure supporting both compact storage and efficient query. By adopting a ray space coordinate system and capitalizing on the properties of the range image, fast p-field construction, efficient storage and query can be achieved.

All the sample points in a range image are acquired from one viewpoint (or one view direction under orthographic view). Any ray shooting from the viewpoint (or along the view direction) will intersect with the surface defined by the range image only once, if it does intersect. A grid $P(x, y, z)$ is created by partitioning the ray space. The cells with the same x and y values are on the same ray, which is denoted by $R(x, y)$. Thus, we can assign probability values ray by ray instead of cell by cell. Under the ray space coordinate system, the value of the p-field is non-linear along the z direction. Besides, the maximum position on the ray may be shifted under discrete sampling (Figure 3(a)). These problems can be avoided by using $< center, size >$ to represent the ray $R(x, y)$, where $center$ is the z value of the intersection's camera coordinate, and $size$ is the projection of the corresponding Gaussian size on the camera coordinate system z direction (Figure 3(b)). With a pre-computed lookup table of the standard Gaussian distribution $G(i)$, a query along the ray for any z value is efficient and precise:

$$P(z) = \begin{cases} 0 & d > size \\ G(d/size)/size & \text{otherwise} \end{cases} \quad (7)$$

where $d = \mathrm{dist}(z, center)$.

In this way, the p-field is represented by a set of sample rays evenly distributed in ray space, with only two float values per ray. This 3D spatial probability distribution is stored in a compact 2D data structure. The p-field resolution can be represented by the resolution of the sample rays. Given any position v, below are the steps for computing the probability $P(v)$. First, compute the x and y values of the ray space coordinate for v to find the four nearby rays; then, get the probabilities at z (the z value of the camera coordinate) of the four rays, followed by bilinear interpolation to get the result.

If v is a sample point in the range image, there is no guarantee that a sample ray will go through it under a dis-

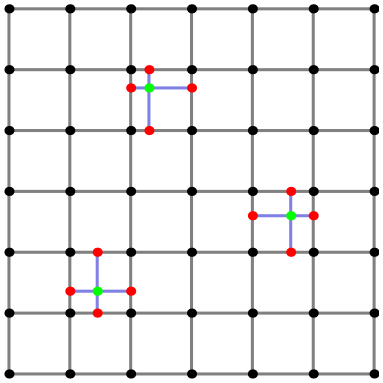

Figure 4. Five extra rays per sample point. Black dots denote evenly distributed sample rays. Green ones are sample rays through the sample points, and red ones are supporting bilinear interpolation.

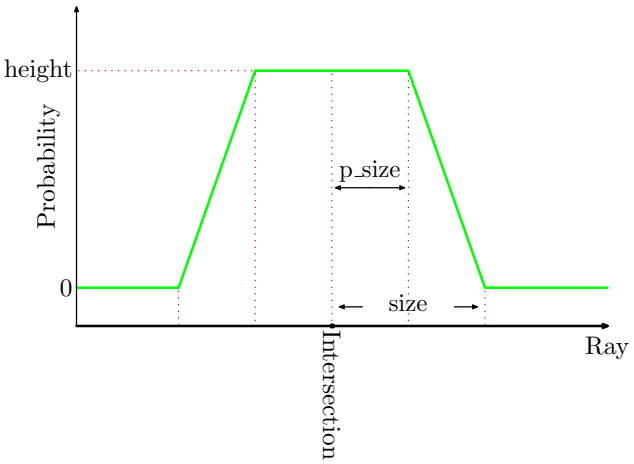

Figure 5. Platform error model.

crete sampling of ray space. The query result $P(v)$ from the interpolation of the discrete sampling may not be a local maximum of the p-field as it is supposed to be. This inaccuracy also can be eliminated. If the sample points of the range image are acquired by rays from a 2D regular lattice, we can set the resolution of the p-field as a multiple of this regular lattice so that a sample ray goes through every sample point. In the case that the range image does not have this property, we add five sample rays per sample point to rectify this deficiency (Figure 4). One goes through the sample point and the other four share the same x or y values of nearby regular rays to support bilinear interpolation. The number of these extra sample rays is determined by the number of sample points and is independent of the p-field resolution. The exact locations of the sample points are preserved with the price of a little extra storage. The p-field resolution should be finer than the range image resolution so that at most one sample point projects between every four neighboring regular sample rays.

4 Other Sensor Error Models

One advantage of our p-field method is its flexibility in supporting different sensor error models. Figure 5 is the platform error model, in which the probability is constant within a certain distance of the intersection and linearly decreases to zero afterwards. If the range image is sparsely sampled, details may be lost between the sample points. The platform error model is adopted for this kind of situation. In the p-field, $< center, size, p\_size >$ is used to represent the platform error for the corresponding ray. Since the integration along the ray equals one, the platform height $height$ satisfies $height = 1/(size + p\_size)$. The equation below is for the query in the ray:

$$P(z) = \begin{cases} 0 & d > size \\ height & d < p\_size \\ \frac{size-d}{size-p\_size} * height & \text{otherwise} \end{cases} \quad (8)$$

where $d = \text{dist}(z, center)$.

Any sensor error model can be used in our p-field method, either represented by a few parameters or by a lookup table. Each ray may have its own error model. For example, if the ray-surface intersection is near sample points, it uses the Gaussian error model, otherwise, the platform error model is used.

In case the user wants to use different sensor error models in the different stages of the alignment, updating the p-field with a different sensor error model is very efficient since we do not need to start over to find ray-surface intersections. Only a few parameters of the rays that do intersect with the surface are updated according to the new sensor error model.

5 Consistency of Empty Space

The space between the camera and the sample points of a range image is empty [7]. Consistency of the empty space can be incorporated into our p-field registration method. In our method, the space before the intersection along the ray with zero probability is empty space, such as the brown regions in Figure 1 and Figure 2. When a sample point's corresponding position in the p-field is within the empty space, instead of making zero contribution as in Equation 1, it makes negative contribution. Thus, we change Equa-

tion 1 to :

$$F(T) = \sum_{i=1}^{N} c(v_i)P(T \cdot v_i) - w \sum_{i=1}^{N} c(v_i)\delta(v_i) \quad (9)$$

$$\delta(v_i) = \begin{cases} 1 & v_i \text{ in empty space} \\ 0 & \text{otherwise} \end{cases}$$

where w is the weight for empty space. The corresponding error metric is:

$$E(T) = \sum_{i=1}^{N} c(v_i)(1 - P(T \cdot v_i)) + w \sum_{i=1}^{N} c(v_i)\delta(v_i) \quad (10)$$

6 Hierarchical Alignment

Hierarchical alignment is a strategy commonly adopted in registering range images using ICP [12, 13]. Two range images are aligned from a coarse level to the finest level by constructing range image hierarchies. In this way, not only is the computational time reduced, but also the minimization less likely to get stuck in a local minimum. For our p-field method, it is easy to use hierarchical alignment. Since the p-field is a set of sample rays, among which most are evenly-distributed, only the finest resolution p-field is needed. Querying in the next level of the p-field is done by using only one fourth of the evenly-distributed sample rays taken at every other ray in the x and y directions.

7 Results and Comparisons

We tested our p-field method with both synthesized range images and scanned ones from the real world. All tests were done on a P4 3.06GHz PC with 512M RAM.

One test scene is the Puget Sound Terrain (Fig. 6), represented by a 2045×2045 height map. Two range images are generated from different viewpoints, with most parts overlapping. The resolution of the range images is 570×570, with $265,386$ (yellow in Fig. 6) and $264,056$ (blue in Fig. 6) sample points, respectively. The range images encompass approximately one quarter of the terrain model, which is the roughest part of the terrain with high frequency features. Using the distance between two neighboring pixels in the height map as a unit length, the average distance between neighboring sample points in the range images is 2.64. Table 1 shows the p-field construction times for the range image with $265,386$ sample points at different resolutions. Except for the two lowest resolutions, the construction times also include about 3.5 seconds for the sample rays through the $265,386$ sample points. For the two lowest resolutions, the sampling rates of the sample rays are below the range image sampling rates, so that more than one extra sample ray will appear between four neighboring regular sample rays, which is not allowed in our p-field model. The memory cost of the p-field is linear with the number of sample rays, with 16 to 24 bytes per sample ray depending on the type of sensor error model.

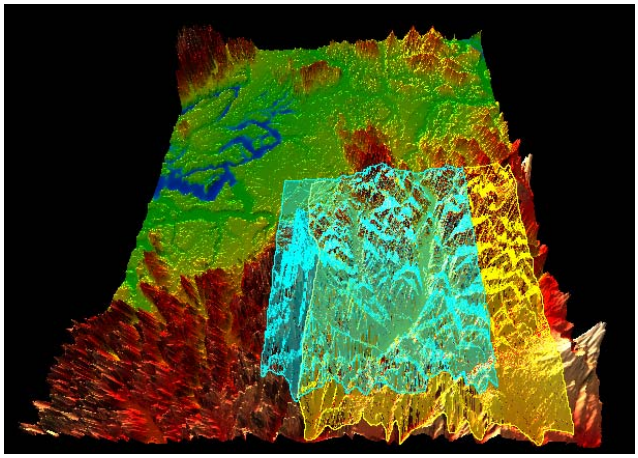

Figure 6. Synthesized test scene: two 570x570 range images from Puget Sound Terrain model, with 265,386 (yellow) and 264,056 (blue) sample points, respectively.

Table 1. P-field construction time for Puget Sound Terrain range image. The resolution is represented by the size of the evenly-distributed sample rays and the maximum distance between nearby sample rays on the far plane (shown in parenthesis).

P-field Resolution	Number of Rays	Construction Time (second)
$134 \times 134(4)$	17,956	2.0
$263 \times 262(2)$	68,906	2.2
$524 \times 523(1)$	1,600,982	6.4
$1043 \times 1041(0.5)$	2,412,693	8.5

Table 2 shows the alignment results of the terrain range images using our p-field method. The number of sample points used in the registration is the N in Equation 2. The alignment time is determined by the number of the sample points used and the number of steps in the downhill simplex method for minimizing the error metric. When the resolution of the p-field is roughly equal or above the range image sampling rate, increasing the p-field resolution does not affect the alignment result noticeably. This is because our p-field representation preserves the original range image information without any accuracy loss, and does not introduce any discretization error.

Table 2. Registration results of Sound Terrain range images.

P-field Resolution	Number of Sample Points	Time (second)	RMS Error
$134 \times 134(4)$	1,032	2.4	0.76
$134 \times 134(4)$	4,145	4.1	0.28
$134 \times 134(4)$	16,580	6.4	0.67
$134 \times 134(4)$	66,342	17	0.37
$134 \times 134(4)$	265,386	21	0.38
$263 \times 262(2)$	1,032	4.6	0.38
$263 \times 262(2)$	4,145	7.3	0.30
$263 \times 262(2)$	16,580	10	0.40
$263 \times 262(2)$	66,342	21	0.34
$263 \times 262(2)$	265,386	27	0.31
$524 \times 523(1)$	1,032	9.3	0.44
$524 \times 523(1)$	4,145	11	0.25
$524 \times 523(1)$	16,580	16	0.17
$524 \times 523(1)$	66,342	48	0.20
$524 \times 523(1)$	265,386	58	0.12
$1043 \times 1041(0.5)$	1,032	7.0	1.30
$1043 \times 1041(0.5)$	4,145	16	0.30
$1043 \times 1041(0.5)$	16,580	22	0.17
$1043 \times 1041(0.5)$	66,342	47	0.25
$1043 \times 1041(0.5)$	265,386	45	0.19

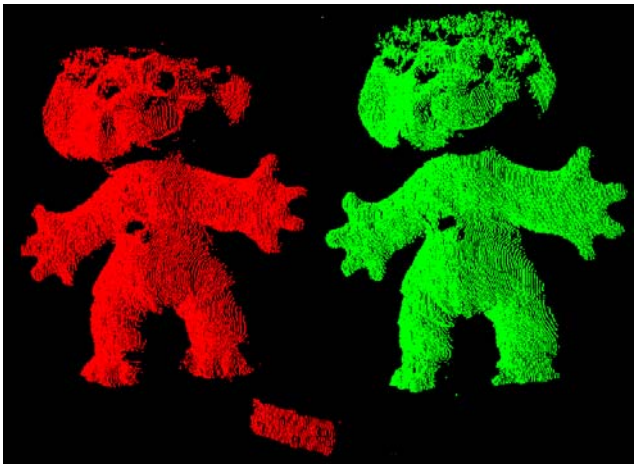

Figure 7. Two 500x500 scanned range images with 47,930 (red) and 46,768 (green) sample points, respectively.

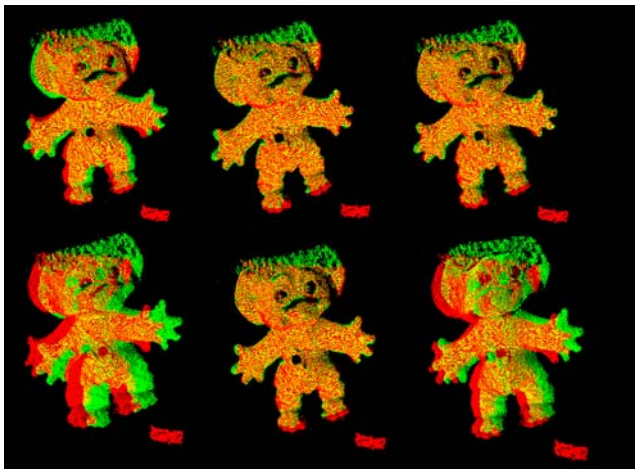

Figure 8. Left column: initial transformations. Middle column: registration results of our p-field method. Right column: registration results of the quaternion-based ICP method.

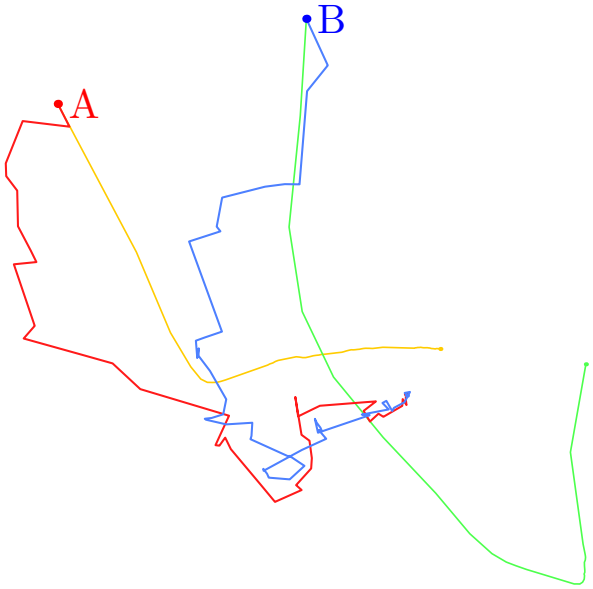

Figure 9. The transformation update routes in registration starting from two initial positions (A and B correspond to the first and second rows in Figure 8, respectively). The red and blue lines are for our p-field method, while the yellow and green lines for the quaternion-based ICP method [12].

Figure 7 displays two range images from a scanned doll. The left column of Figure 8 shows two initial transformations, and the middle column shows the registration results of our p-field method. The p-field at resolution $637 \times 482(0.3)$ is constructed in 0.46 seconds, and it takes 19 to 27 seconds to align the two range images.

We also compare our p-field method with a quaternion-based ICP method [12], which finds a set of corresponding point pairs in each iteration and the best transformation using a closed form solution. The right column of Figure 8 shows the corresponding results for this ICP method. It takes about 60 seconds for the quaternion-based ICP method to converge when all $265,386$ sample points are used. Figure 9 shows the transformation update routes in transformation space in minimizing the error metric of our p-field method, and of the routes in the iterations of the quaternion-based ICP method from the two initial positions (A for the first row of Figure 8 and B for the second row of Figure 8). Our p-field method is quite robust and much less likely to get stuck in a local minimum. Starting either from initial transformation A or B, our method can precisely register the range images by searching in many different directions in the transformation space. For the quaternion-based ICP method, the result from position A is acceptable, and the result from position B is a local minimum far away from the correct registration. This is because the closed form solution only searches in certain directions and is more likely to get stuck in a local minimum.

8 Conclusion and Future Work

This paper introduces a new ICP variant for range image registration using a p-field. We describe the p-field, the probability distribution of the surface position from range image, based on sensor error models. The p-field representation, which supports fast construction, compact storage, and efficient query, is flexible in dealing with different range images and different registration strategies. Thus, range images can be precisely registered using the p-field.

It is possible to extend our p-field registration method for multiple range image registration. Another direction for future work is to use the p-field for surface reconstruction. An error metric may be defined on the p-field for finding the surface location from multiple range images.

9 Acknowledgements

This work has been partially supported by an NSF grant CCR-0306438 and ONR grant N000140110034. The Puget Sound Terrain model is courtesy of USGS and University of Washington.

References

[1] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[2] G. Blais and M. D. Levine. Registering multiview range data to create 3D computer objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):820–824, 1995.

[3] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *IEEE International Conference on Robotics and Automation*, pages 2724–2729, 1991.

[4] D. W. Eggert, A. Lorusso, and R. B. Fisher. Estimating 3D rigid body transformations: a comparison of four major algorithms. *Machine Vision and Applications*, 9:272–290, 1997.

[5] A. Johnson and S. Kang. Registration and integration of textured 3-D data. *International Conference on Recent Advances in 3-D Digital Imaging and Modeling*, pages 234–241, 1997.

[6] P. J. Neugebauer. Geometrical cloning of 3D objects via simultaneous registration of multiple range images. *International Conference on Shape Modeling and Application*, pages 130–139, 1997.

[7] P. Rademacher. Range image registration via consistency of empty space. *http://www.cs.unc.edu/ ibr/projects/emptyspace/*, 1999.

[8] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-time 3D model acquisition. *SIGGRAPH, Computer Graphics Proceedings*, pages 438–446, 2002.

[9] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. *Proceedings of the Third Intl. Conf. on 3D Digital Imaging and Modeling*, pages 145–152, 2001.

[10] M. Rutishauser, M. Stricker, and M. Trobina. Merging range images of arbitrarily shaped objects. *CVPR*, pages 573–580, 1994.

[11] B. Schiele and J. Crowley. A comparison of position estimation techniques using occupancy grids. *IEEE International Conference on Robotics and Automation*, pages 1628– 1634, 1994.

[12] G. Turk and M. Levoy. Zippered polygon meshes from range images. *SIGGRAPH, Computer Graphics Proceedings*, pages 311–318, 1994.

[13] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.