

## Registration of Range Data Using a Hybrid Simulated Annealing and Iterative Closest Point Algorithm

Jason Luck  
Colorado School of Mines  
jluck@mines.edu

Charles Little  
Sandia National Laboratory  
cqlittl@sandia.gov

William Hoff  
Colorado School of Mines  
whoff@mines.edu

### Abstract

The need to register data is abundant in applications such as: world modeling, part inspection and manufacturing, object recognition, pose estimation, robotic navigation, and reverse engineering. Registration occurs by aligning the regions that are common to multiple images. The largest difficulty in performing this registration is dealing with outliers and local minima while remaining efficient. A commonly used technique, iterative closest point, is efficient but is unable to deal with outliers or avoid local minima. Another commonly used optimization algorithm, simulated annealing, is effective at dealing with local minima but is very slow. Therefore, the algorithm developed in this paper is a hybrid algorithm that combines the speed of iterative closest point with the robustness of simulated annealing. Additionally, a robust error function is incorporated to deal with outliers. This algorithm is incorporated into a complete modeling system that inputs two sets of range data, registers the sets, and outputs a composite model.

## 1 Introduction

Registration is a process through which two sets of geometric data are aligned. This paper concentrates on registering two partially overlapping scans of range data, which are simply sets of three-dimensional points sampled by a sensor from the surfaces of objects, as shown in Figure 1.

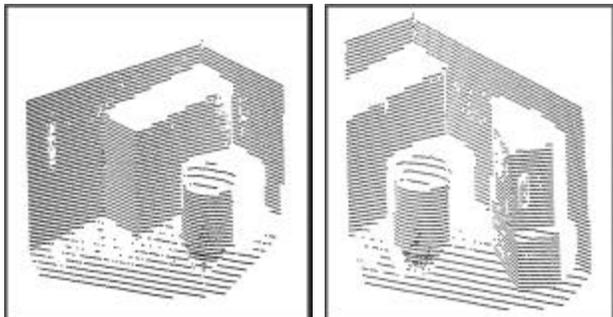


Figure 1 Two partially overlapping range data scans (Set #3) taken with a structured lighting system at Sandia National Laboratories with ~5000 points per scan of a 4x4 ft. corner.

As can be seen in the figure, range data often contains noise and points with large errors (outliers). Also, one cannot rely on a point to be sampled at the exact same position on a surface, in each scan (due to inconsistent point sampling). These characteristics make registration a very difficult problem.

The need for registration is important in applications such as world modeling, part inspection and manufacturing,

object recognition, pose estimation, robotic navigation, and reverse engineering. In this paper, we will focus on world modeling. To create a world model from the data in Figure 1, first a surface model of each scan must be made from the point data, as shown in the top images in Figure 2. Then registration must be performed, meaning that the relative translation and rotation between the two models is determined, to bring them into alignment. Finally, the scans must be stitched together to form a final model, as shown in the bottom image of Figure 2. Creating a surface model is easily performed and stitching the two models together is also relatively easy. Therefore, this paper will emphasize the registration process.

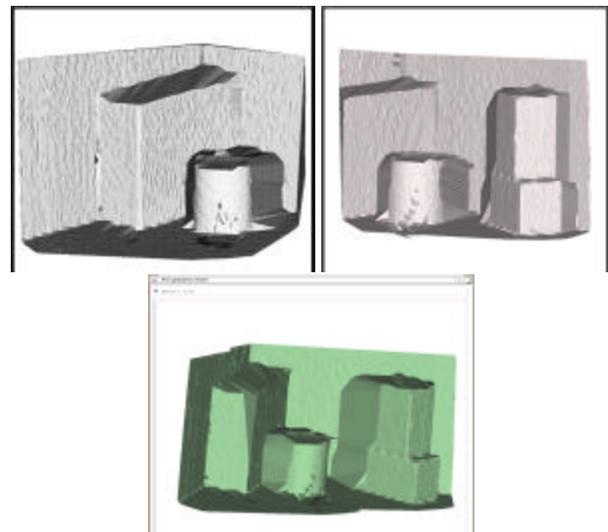


Figure 2 Top: surface models from point data in Figure 1. Bottom: final fused model.

If one has precisely calibrated the extrinsic parameters of the data acquisition devices (i.e., their position and orientation in the world), then one can readily register the scans (referred to as dead-reckoning). However, this accuracy is usually not good enough for model building applications, although it can be used to create an initial guess. Therefore, it is necessary to use information from the data itself to refine the registration parameters.

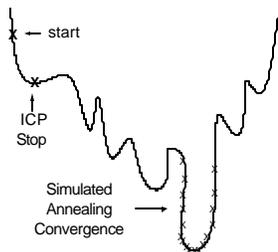
The easiest method for registering two sets of three-dimensional points is the absolute orientation algorithm [4]. However, this requires knowledge of the correspondences between the points, which is extremely difficult to find. An algorithm which deals with this correspondence problem but does not require any pre-processing or feature extraction, is a technique called iterative closest point.

The iterative closest point algorithm [1], ICP, deals with the correspondence problem while efficiently registering two range data sets. The process is designed to register a

data point set to a model point set. The algorithm uses a closest point estimation to compute correspondences. That is, for every data point, the model point that is geometrically closest to the data point is taken as the corresponding point. The process then repetitively uses the absolute orientation algorithm [4] to register the point sets. The steps of the algorithm are as follows:

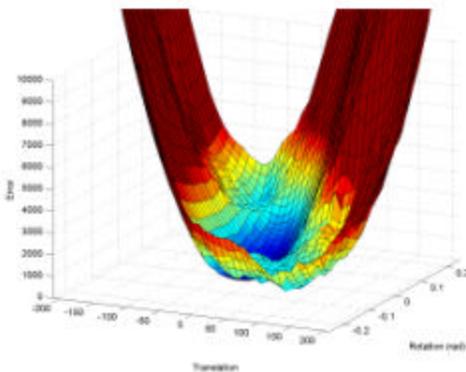
1. Compute the set of closest model points.
2. Compute the registration between the point sets.
3. Apply the transformation to register the sets.
4. Repeat steps 1–3 until  $Error < tol$ .

This process slowly pulls the data points in to the model points. However, ICP will only converge to the closest local minimum of the error function surface, as depicted in Figure 3.



**Figure 3 Conceptual diagram of the convergence of ICP and SA on a one dimensional error surface**

Our application has many local minima, as shown in Figure 4. Using the real data shown in Figure 1 the error function was plotted over 2 dimensions (1 translation and 1 rotation). The result clearly shows a number of local minima contained in a larger steadily decreasing pit.



**Figure 4 Error function vs. translation and rotation**

Accordingly, our algorithm must be able to deal with local minima. The authors of ICP suggest a simple strategy, in which the algorithm is started from several random locations and the best solution is used. However if there are a very large number of local minima, the chance of starting the algorithm close enough to the global minimum is extremely small.

An alternative is to use a stochastic optimization algorithm, such as simulated annealing (SA). Blais and Levine [2] used a modified algorithm called very fast simulated reannealing to perform registration. The technique is able to traverse local minima, but is extremely slow in converging to the global minimum, as depicted in Figure 3.

Accordingly when dealing with an error surface such as in Figure 4 and Figure 3, ICP is efficient but will get stuck in local minima and SA is robust but inefficient. Therefore, a hybrid technique is developed so that local minima can be crossed and the global minimum is found as efficiently as possible. The two methods are combined so that SA works in a similar manner to the suggested random restarts for ICP. However, instead of purely random restarts, SA should provide restarts closer to the global minimum. In this way, SA provides guided restarts that get ICP out of local minima and closer to the actual solution. The final system produces a high level of accuracy and remains as efficient as possible.

The system presented in this paper is a complete registration and modeling system. There are several novel portions of this project.

1. First, a hybrid registration algorithm is developed that employs both an efficient local minimizer and a robust global search algorithm. The resulting algorithm handles local minima while remaining efficient.
2. Secondly, a robust weighting function is developed that can deal with up to 50% outliers.
3. Lastly although segmentation is not the focus of this paper, a segmentation algorithm is developed that accurately segments the overlap data using only a position estimate of the scanners and the data itself.

The entire process for our system is as follows. First data acquisition is performed to acquire both the model and data sets, and a surface model of each scan is created. For this project a Delaunay triangulation was used to create the model: its implementation is straightforward and accordingly skipped but is covered in [7]. The description in this text starts with the segmentation of the overlapping region, which is accomplished using a novel frustum segmentation technique. Next, registration is performed on the segmented regions using a robust error function and incorporating it within the hybrid algorithm. Finally, the two scans are stitched together to yield the final model.

## 2 Detailed Description

### 2.1 Finding the Overlapping Region

The segmentation process assumes that an initial guess for the registration has already been employed, such as through dead reckoning. Since the scans are already roughly aligned, the majority of points in the overlapping region of the data scan lie within the boundary of the model scan. Therefore, one should be able to use the view of the model scanner to pick out the points in the data set that are in the region occupied by the model.

To model the view from the model scanner a frustum is created using the position of the scanner and the outline of its point set. Creating a frustum is accomplished using the surface model and the scanner position. The frustum is simply a set of triangles extending from the scanner

position around the outline of the point set. Since the Delaunay triangulation creates a convex hull around the point set, the frustum is also convex. Therefore, any point within the frustum will be on the inside of each frustum plane.

The result of this segmentation process is a data set that contains only points that could be sampled from the position estimate of the model scanner. After segmentation the overlapping data region should contain less than 50% disparity with the model, which allows for accurate registration through the use of the robust error function.

## 2.2 Robust Error Function

The goal of the error function is to provide an error score that is minimized when the best rotation parameters and translation vector,  $R$  and  $\vec{T}$ , are entered. The error is based upon the squared distances from the data points  $\vec{D}_i$  to model points  $\vec{M}_i$ :

$$d_i^2 = [\vec{M}_i - (R\vec{D}_i + \vec{T})]^2. \quad (1)$$

However range data contains both outliers and inconsistent point sampling, and the error function must be able to deal with these problems.

Chen and Medioni [3] use a surface model to minimize the errors produced by inconsistent point sampling. A surface model allows points to be matched up anywhere on the surface model, and therefore it is no longer necessary for each data point to be matched with a model point sampled from its exact location. This adaptation profoundly increases the final accuracy achieved, and is incorporated into our error function.

Building invariance to outliers inevitably means employing weights. When the scans are aligned outlier points will lie farther from the model surface than good data points. The difficulty lies in assessing what distance is close and what is far. The problem is compounded since our method iteratively pulls in the data points. With this in mind points are dynamically weighted so that the error is calculated according to:

$$error = \left(\frac{1}{n}\right) \sum_{i=1}^n w_i * d_i^2. \quad (2)$$

Where  $d_i^2$  is the squared distance from a data point to the model surface and  $w_i$  is the weight for point  $i$ .

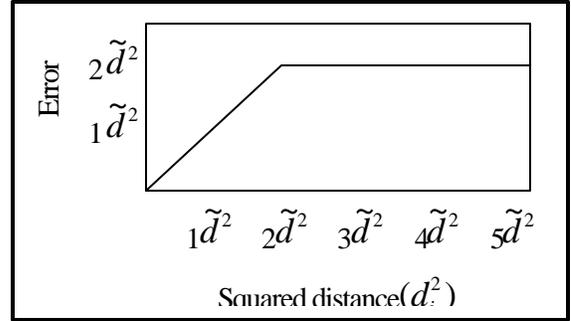
Many techniques simply use a threshold weighting scheme, which eliminates points with errors above the threshold. However, it is difficult to set the threshold when the registration is iteratively improved. Zhang [11] uses a dynamically changing threshold to overcome this problem, but his technique is not perfect. A more straightforward approach is employed by Masuda and Yokoya [8]. Their algorithm avoids the effects of outliers by using the median of the distance between point pairs.

In this way up to 50% of the points can be outliers without affecting the threshold. This is a form of robust estimation [5], which has many useful variations.

Accordingly our weighting is based around the median squared distance from each data point to the surface model,  $\tilde{d}^2 = median(d_i^2)$ . Weights are assigned as:

$$w_i = \begin{cases} 1 & \text{if } d_i^2 < 2\tilde{d}^2 \\ 2\tilde{d}^2 / d_i^2 & \text{if } d_i^2 \geq 2\tilde{d}^2 \end{cases}. \quad (3)$$

Accordingly good points are not scaled, but outliers are scaled so that the individual error contributed is equal to 2 times the median squared distance, as shown in Figure 5.



**Figure 5 Contribution of a single point ( $w_i d_i^2$ ) to the error**

The error function assures that errors accumulated by outliers are minimized but still have some effect (slightly more than a good point which contains the normal error of the sampling device). This small amount of error is incorporated because when the error is allowed to go to zero entire surfaces may be tagged as outliers. For instance if the error is allowed to go to zero, the lowest error score for the corner of a room is achieved when two planes are aligned and the third is hovering above the threshold. This occurs because in this position the hovering points will have an error of zero, but if aligned correctly the points would contribute a small amount of error due to noise in the data. The resulting error function can achieve excellent registration even in the presence of outliers of up to 50% of the data. To aid in registration this error function must be embedded into SA and ICP so that they operate on the same error surface. Once on the same error surface a hybrid algorithm is created to use the two techniques in conjunction.

## 2.3 Simulated Annealing

Simulated annealing is a capable of crossing local minima and locating the global minimum [6]. The form of SA used is a variation of the Nelder-Mead downhill simplex method, which incorporates a random variable to overcome local minima [9]. A simplex is simply a set of  $N+1$  guesses, or vertices, of the  $N$ dimensional state-vector sought and the error associated with each guess. For example we are solving for six registration parameters (translation and rotation); therefore the simplex has 7 vertices and the error associated with each of the vertices. The simplex attempts to walk downhill by replacing the

vertex associated with the highest error by a better point. Through repetitively replacing bad vertices with better points the simplex moves downhill.

SA introduces a random element, based upon a "temperature" variable, to the downhill simplex method to allow the algorithm to escape local minima. A random amount, scaled by the temperature, is subtracted from each tested replacement point. Therefore while any move that is a true downhill step will be accepted, some additional uphill steps will also be accepted. At high temperatures most moves are accepted and the simplex roams freely over the search space. At lower temperature only smaller uphill steps can be accepted. As the temperature is slowly lowered the simplex crawls out of local minima and converges upon the global minimum.

SA is set up to perform registration by incorporating our error function, and employing a temperature schedule based upon experience with the algorithm. The set-up procedure is covered in detail in [7].

## 2.4 Adapting Iterative Closest Point

In order to use both methods we had to adjust the ICP algorithm so that it operates on the same error surface as SA. To do this the algorithm must use the same point correspondences and the same error function.

Our method for computing point correspondences uses the distance from each data point to the surface model. However, the original ICP algorithm works by matching data points with the closest model point. ICP can be altered to use our correspondences by substituting the closest point on the model surface for the closest model point.

The robust error function incorporated into SA weights each distance so that outlier points have a lesser effect. ICP repetitively calls the absolute orientation algorithm to calculate the transformation parameters. Accordingly the absolute orientation algorithm had to be modified to include weighting of the point pairs. The weights are determined by the robust error function, and incorporated into the absolute orientation algorithm as described by Horn [4]. The process now operates on the same error surface as SA, and a hybrid algorithm can be developed.

## 2.5 Hybrid Algorithm

The idea behind the hybrid algorithm is to use ICP to speed up SA. The process for combining the two is relatively simple; the basic idea is for SA to choose good starting points for ICP. In other words ICP does the majority of the work, but when trapped in a local minima SA will attempt to traverse the minima and choose a new starting point for ICP. A flow chart for the process is shown in Figure 6.

The process starts by running ICP from the original position, which will converge to the nearest local minima. If dead reckoning has aligned the scans close enough to

the global minimum, ICP will converge upon it. The error should be below the error threshold and the algorithm will stop without ever using SA. If not ICP will stop above the threshold, and SA is then employed to search about the error surface for a point with a lower error score. To ensure SA is searching in the right area, one of the simplex vertices is moved to the point at which ICP stopped. SA will continue until a better point is found where it relinquishes control to ICP, or until the temperature falls below the minimum. The entire process repeats until the error is driven below a set threshold [4].

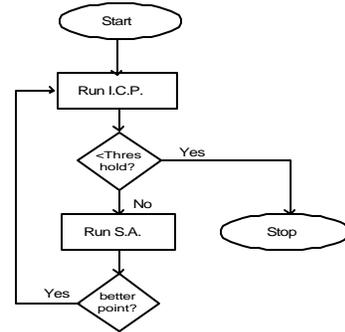


Figure 6 Flow chart of hybrid algorithm.

Figure 7 depicts how this process moves across the error surface. The error surface was created by rotating and translating the real world data shown in Figure 1 across the x-axis. The figure reproduces the route of the hybrid algorithm across the error surface and to the global minimum.

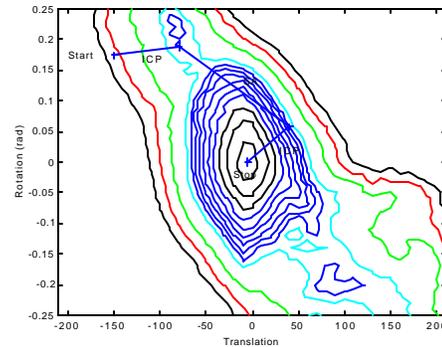


Figure 7 Hybrid algorithm on a 2-D error surface.

The figure shows the effectiveness of the hybrid algorithm. As can be seen ICP initially converges on a local minimum. Then SA moves to a better point, and the algorithm is able to swiftly converge to the global minimum. Figure 8 shows how the error score changes for the run in Figure 7.

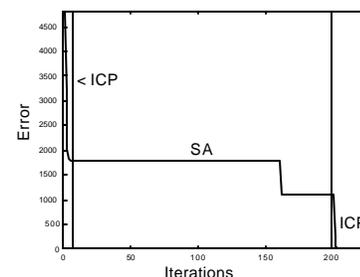


Figure 8 Error score for the hybrid algorithm.

## 2.6 Model Creation

Once the scans have been registered, the final model is created by stitching the two surface models together. The first step is to remove portions of the data that overlap with the existing model, by repeating the frustum technique. The remaining data is then stitched into the existing model at the intersections. This process is covered in detail in [7].

## 3 Results

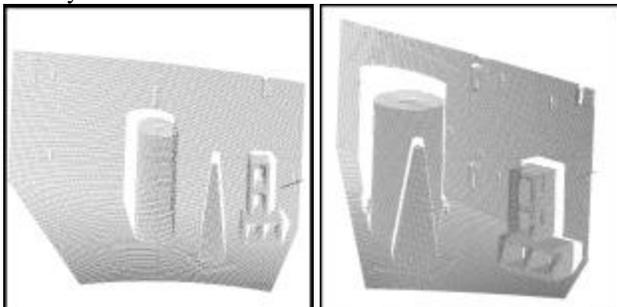
First, the hybrid algorithm was compared to ICP by itself and SA by itself. Each method was employed ten separate times from three initial positions using data set #3 and 4. The results are shown in Table 1. One iteration for data set 3 (registering ~250 data points ~5000 model points) required ~1 second, and one iteration for data set 4 (registering ~600 data points to ~10000 model points) required ~6 seconds on an SGI R4400 Indigo 2.

**Table 1 Comparison of hybrid, ICP, and SA algorithms averaged from 3 starting poses spread across 5 degrees of rotation and 250 mm of translation.**

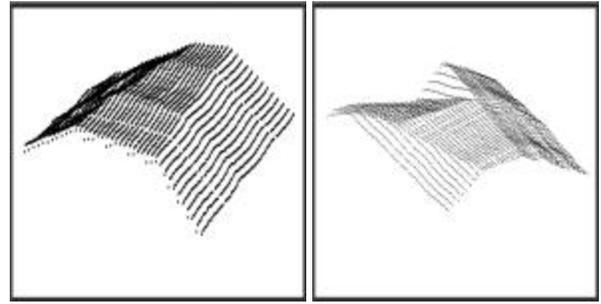
	Average Error (mm) <sup>2</sup>	Average # Iterations
Data Set # 3		
Hybrid	1.31	1479
S.A.	1.35	6508
I.C.P.	2.21	31
Data Set # 4		
Hybrid	6.61	2301
S.A.	8.34	6419
I.C.P.	14.8	37

The results indicate that the hybrid algorithm achieves the same level of accuracy as SA, while both algorithms far exceed the level achieved by iterative closest point. However, the results also show that the hybrid algorithm takes on the average only 23% of the iterations consumed by SA. Accordingly the hybrid algorithm is able to achieve the same level of accuracy as SA, and does so in about 1/4<sup>th</sup> the time.

In addition the hybrid algorithm was run on 3 real data sets: two structured lighting sets (Figure 1 & Figure 10) taken at Sandia National Laboratories and one set taken with a Coleman laser range detector (Figure 9), and also on 2 synthetic data sets.

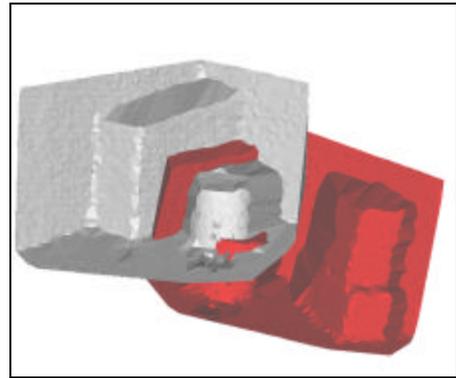


**Figure 9 Data set #4 of a 6x4 ft scene with ~60,000 points.**



**Figure 10 Data set #5 of a 3x3 ft portion of a plane wing with ~3000 points per scan.**

For each data set the algorithm was run from five separate initial positions spread across 5 degrees of rotation and 50 mm of translation, one of which can be seen in Figure 11.



**Figure 11 View of initial pose for Data Set #3.**

The results are summarized here but can be viewed in more detail in [7]. When run on the synthetic data set where perfect registration was possible the hybrid algorithm always found the exact registration. It was also noticed that the algorithm fell into an average of 2 local minima before the global minimum was found.

The results for data set 3 (Figure 1) and data set 4 (Figure 10) are shown below in Table 2 & Table 3.

**Table 2 Results for data set #3. The ground truth error is 1.20mm<sup>2</sup> for this data set.**

Pos.	Final Error (mm) <sup>2</sup>	Local Minima	Iterations ICP + SA
1.	1.38	5	28 + 1816
2.	1.28	5	40 + 1714
3.	1.29	8	28 + 1614
4.	1.37	4	27 + 1417
5.	1.21	5	35 + 1210

**Table 3 Results for data set #4. The ground truth error is 6.02mm<sup>2</sup> for this data set.**

Pos.	Final Error (mm) <sup>2</sup>	Local Minima	Iterations ICP + SA
1.	6.25	3	28 + 1816
2.	6.03	2	40 + 1714
3.	6.46	4	28 + 1614
4.	16.07	6	27 + 1417
5.	6.08	6	35 + 1210

These results show that although the final poses are extremely close, the algorithm failed to converge to the same exact location when run from different initial positions. This reveals that our error surface has many shallow local minima surrounding the global minimum. This is discussed in more detail in the conclusions.

When the algorithm was run on data set #5 (Figure 10), poor results were achieved. Since the overlap is ambiguous in at least one direction and many local minima occur along this direction, the algorithm had trouble travelling along this direction to find the global minimum. Accordingly the algorithm can be expected to perform poorly when an ambiguous overlap is used.

## 4 Discussion and Conclusions

The system presented in this paper is a complete surface registration and modeling process. It inputs two overlapping range data sets, and outputs a registered and fused surface model comprised of both scans. The largest contribution of the method is the registration process. Incorporating a robust error function into ICP and SA has been done before. However by incorporating the same robust error function into both routines the system not only gained the robustness of the error function, but it was also able to use both minimization methods in unison. By linking the methods to cooperate with one another, the system now had the robustness to deal with local minima while retaining efficiency. This novel approach proves to be much more efficient than a global technique by itself, and much more accurate than a local technique. In addition, a secondary contribution comes from the effectiveness of the novel segmentation approach.

The method does have limitations. First, the estimate of the range sensor position must be accurate enough to ensure that there is less than 50% disparity in the segmented overlap. However, it was logically shown in subsection 2.1 that even with highly inaccurate dead reckoning system the frustum technique performs well. Additionally the algorithm will have problems if the error surface is relatively flat with many local minima. This type of error surface is extremely difficult to solve, and there is no guarantee that our method will do so.

There is also an existing algorithm that would improve our process. The current method for finding the closest point on the model surface for each data point is very inefficient. In Simon's paper [10] speed improvements down to approximately 7% of the original search time are achieved through the use of k-d trees and closest point caching. The introduction of these improvements would greatly enhance the efficiency of the registration phase.

Lastly, there is an anomaly in the results that warrants further discussion. First the algorithm did not consistently converge to the same minimum. Accordingly, small local minimum pits must surround the global minimum. An acceptable solution may be to employ ICP with zero weighting from the convergence point of original hybrid

algorithm. If the shape of the error function itself is causing these shallow local minima, employing zero weighting may remove them. In addition since the scans are already very close to the proper alignment outliers will be easy to detect and zero weighting should improve the final accuracy of the method.

In conclusion, this method proves to be extremely robust while remaining efficient. The frustum segmentation technique is accurate even when the estimate of the scanner position is not precise. By incorporating a robust error function based around the median, registration is accurate even when presented with up to 50% outliers. Lastly the combination of SA and ICP proved to be very effective at negotiating the type of error surface encountered in the registration of range images.

## 5 Bibliography

- [1] Besl, P.J. and N.D. McKay, *A Method for the Registration of 3-D Shapes*. IEEE Transactions on PAMI, 1992. **14**(2): p. 239-56.
- [2] Blais, G. and M.D. Levine, *Registering Multiview Range Data to Create 3D Computer Objects*. IEEE Transactions on PAMI, 1995. **17**(8): p. 820-4.
- [3] Chen, Y. and G. Medioni, *Object Modeling by Registration of Multiple Range Images*. Image and Vision Computing, 1991. **10**(3): p. 145-55.
- [4] Horn, B.K.P., *Closed-Form Solution of Absolute Orientation Using Unit Quaternions*. Journal of the Optical Society America, 1987. **4**(4): p. 629-42.
- [5] Huber, P. J. (1981). *Robust Statistics*. New York, John Wiley & Sons.
- [6] Kirkpatrick, S., J.C.D. Gelatt, and M.P. Vecchi, *Optimization by Simulated Annealing*. Science, 1983. **220**(4598): p. 671-680.
- [7] Luck, J., *Registration of Range Images Through the Use of a Hybrid Simulated Annealing and Iterative Closest Point Algorithm*, Masters Thesis in Engineering, 1999, Colorado School of Mines: Golden, CO.
- [8] Masuda, T. and N. Yokoya, *A Robust Method for Registration and Segmentation of Multiple Range Images*. Computer Vision and Image Understanding, 1995. **61**(3): p. 295-307.
- [9] Press, W.H., *et al.*, *Numerical Recipes in C*. 1994, New York: Cambridge University Press.
- [10] Simon, D.A., M. Hebert, and T. Kanade. *Real-time 3-D Pose Estimation Using a High-Speed Range Sensor*. IEEE Conference on Robotics and Automation. 1994.
- [11] Zhang, Z., *Iterative Point Matching for Registration of Free-Form Curves and Surfaces*. International Journal of Computer Vision, 1994. **13**(2): p. 119-52.