Registering Two Overlapping Range Images

Roth, Gerhard

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de Technologie de l'information

NRC-CNRC

*Registering Two Overlapping Range Images**

G. Roth
October 1999

***published in** Proceedings of the Second International Conference on Recent Advances in 3-D Digital Imaging and Modeling (3DIM'99),Ottawa, On. October 4-8, 1999. pp. 191-200. NRC 43637.

Registering two overlapping range images

Gerhard Roth
Visual Information Technology Group
National Research Council of Canada
Ottawa, Canada K1A OR6
Gerhard.Roth@iit.nrc.ca

Abstract

This paper describes a method of automatically performing the registration of two range images that have significant overlap. We first find points of interest in the intensity data that comes with each range image. Then we perform a triangulation of the 3D range points associated with these 2D interest points. All possible pairs of triangles between the two 3D triangulations are then matched. The fact that we have 3D data available makes it possible to efficiently prune matches. We do this pruning by using a simple and effective set of compatibility tests between potentially matching triangles and vertices. The best match is the one that aligns the largest number of interest points between the two range images. The algorithms are demonstrated experimentally on a number of different range image pairs.

1 Introduction

Registration is the process of aligning images so that they are in a common co-ordinate frame. When building geometric models from dense 3D range images [1] it is necessary to acquire images from different viewpoints. The first step in the model building process is the registration of these range images. This registration step currently relies on accurate mechanical positioning devices, or on manual processing. The idea of using the 3D data itself to perform the registration automatically is attractive. The assumption is that individual range images will have a significant overlap. When the registration between two images is correct these overlapping portions of both images should blend together with little error, since they represent the same surface region.

In practice, this type of data driven registration can be further divided into two subclasses: constrained and unconstrained. In the constrained case the assumption is that the transformation between the two range images is already approximately known. This initial estimate of the transformation is then refined with an iterative closest point (ICP) algorithm [2]. When there are no prior constraints on the transformation the standard ICP approach will often fail, since it requires an approximately correct initial transformation to converge.

The problem of automating the registration of two overlapping range images when there are no prior constraints on the transformation between them is far from being solved. Some methods first triangulate each range image, and then attempt to match the triangulations. The difficulty is that the resolution of the triangulations is selected heuristically and the matching process is rather complex [3, 4]. Attempts at making the ICP process itself more robust have had some success [5]. However, they do not address the basic limitation of the ICP algorithm, that is the necessity of a-priori having an approximate estimate of the transformation. Without this estimate the ICP often finds a local minimum instead of the global minimum which represents the best transformation. This is not surprising since the ICP is searching a non-convex, multi-dimensional space using a gradient descent algorithm. It is possible to search this multi-dimensional space using some kind of randomized search [6]. Such a search process is very costly computationally, and has not been shown to be practical for the problem of unconstrained range image registration.

Other work on registering 3D points sets has been done in the context of medical imagery [7], but this does not seem to be applicable to range image registration because of the differing nature of the data. The most successful methods so far in attacking the problem of unconstrained range image registration use the rigidity constraint. This constraint says that when undergoing a rigid transformation the distances between 3D points is preserved. In [8], the rigidity constraint is used in conjunction with a random sampling algorithm to find the registration between two range images. This algorithm takes as input all the points in both range images, and has a running time proportional to the total number of 3D points. Since a single range image may have up to 100K of 3D data points the initial range images are sub-sampled to be about 5k points. However, this means that the resulting transformation that registers the two range images is not as accurate as it could be. It is also not clear how much the range images must be sub-sampled for the algorithm to be practical. The rigidity constraint is also used in [9]. Here, 3D point triangles are matched using principal curvatures and the Darboux frame. This method is applied to a number of range images, but again since the running time is proportional to the total number of 3D points the images are sub-sampled to be around 4K data points. A similar curvature based approach for registration of rigid bodies has also been proposed [10]. The problem is that the computation of curvatures is known to be very noise sensitive.

We propose a new way to automatically register two range images, without any prior knowledge of the transformation between them. It is based on the fact that for every range image, there is also a corresponding intensity image which is in one to one registration with the range image. This intensity image represents the amount of light returned by the reflected laser beam. Since this intensity image is only obtained at the frequency of the laser, it is not the same as a standard broad spectrum intensity image. However, the texture of the objects scanned by the rangefinder is still clearly visible. Such an intensity image has previously been used to select matching features between two range images [11]. In the past this feature matching process was done manually. Our idea is to automate this feature matching process to make it possible to register two range images without requiring any user interaction. The basis of our approach is the rigidity constraint, which is also the basis of other registration methods [8, 9]. Our method differs because we find feature points in the intensity image, and then use the associated 3D values of these feature points in the range image to compute the registration. Since there are usually no more than 500 feature points our approach is very efficient. It does however, require texture on the objects being scanned, which is not a requirement of some other methods [8, 9]. However, we are able to handle geometrically symmetric objects, which can not be said of methods that use only the 3D data and ignore the corresponding intensity images [8, 9]. Our method has been tested on full sized 3D range images, which contain from 60K to 100K data points, as opposed to sub-sampled range images. We therefore obtain more accurate registration results, and still do so in reasonable time. To our knowledge, our approach is the only automated registration method which does not require that the range images be significantly sub-sampled.

Instead of using the intensity data of the laser rangefinder to find features it is possible to use the intensity data of a separate colour digital camera. Such a camera can be co-mounted with the laser rangefinder [12, 13]. Then a calibration process can be used to find the associated range point for each pixel of the color camera. This type of color image differs from the intensity image that comes from the reflected laser light of the rangefinder since it is a broad spectrum image. It is therefore easier to find features in such a separate color image. These features can also be used in our registration algorithm, assuming the 2D and 3D sensors are properly calibrated. In this paper we use only the features that come from the intensity data associated with the laser rangefinder data. This is a harder test of our algorithm, but even in this case it is often possible to successfully perform reliable registration.

Note that for our approach be successful there must be at least 20% to 30% overlap between the two range images in order for there to be enough common feature points. This is also a requirement of other data based range image registration schemes [8, 9]. Our algorithm uses corner-like features as the basis of the matching process. Such features have been used with 2D intensity images to compute the fundamental and essential matrix [14, 15] when there is no 3D data at all. For these 2D matching algorithms to be efficient it is necessary that the viewpoints of the two intensity images be relatively close together. This is because they gain their efficiency by limiting the range of possible correspondences of a feature point. The assumption is that the matching feature point in the other image is within a limited distance of the original feature point, usually no more than one quarter of the image size [15]. If there is no limit on the range of possible correspondences for each 2D feature point then these algorithms become computationally infeasible. Such a correspondence search limit is natural if the 2D images are obtained from a video sequence, since in this case they are acquired at a rate of 30hz. By contrast, range images are normally acquired from substantially different viewpoints. Therefore, it is not reasonable to make any assumptions on the set of possible correspondences for a given feature point. In other words, we are solving the problem of unconstrained image registration, not the less difficult problem of constrained image registration. The thesis of this paper is the that when 3D data is available this more difficult problem can still be solved efficiently because the rigidity constraint makes it possible to prune the vast majority of false matches. This is not the case when using 2D data alone, since in this case the rigidity constraint does not hold.

2 Algorithm Description

The registration algorithm has as its input two overlapping range images along with their associated intensity images. The output is the relative 3D transformation between these two images. The algorithm proceeds as follows:

1. Compute interest points in the intensity data of each of the two range images.

2. Compute a 3D Delaunay tetrahedrization of these interest points using their associated 3D range points.

3. Find all the compatible triangle pairs between the two range images. The compatibility criteria consider both the triangles and the individual triangle vertices.

4. For each compatible 3D triangle pair, compute a transformation between the range images.

5. Return the transformation which aligns the largest number of 3D interest points.

We will now describe each step in the registration algorithm in more detail.

2.1 Compute interest points

Informally, an interest point is an area of an image where there is a large change in the local intensity. The procedure to compute such points is to first find the local maxima of the intensity gradient [16]. Then the maxima that are above a certain threshold are classed as interest points. An example of a good interest point is a corner, since it is clearly a maximum of the gradient in both directions. By contrast, an ordinary edge has a significant intensity gradient only in one direction, and is therefore not a good interest point.

The number of interest points found by this process depends on the threshold value of the interest operator, and also on the texture of the object. We will show in our experiments that successful registration results can be achieved using a wide variety of interest operator thresholds. We do require that the objects have enough texture for a significant number of interest points to be found. The feature points found by this interest operator in the intensity component of the range image of a duck is shown in Figure 1.

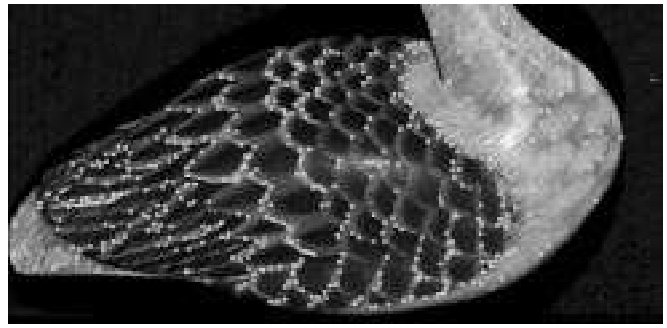


Figure 1: Feature points in the intensity component of the range image of a duck.

2.2 Compute Delaunay Tetrahedrization

Each interest point in the intensity data has an associated 3D data point in the range image. Three such interest points therefore define a 3D triangle. Matching a single pair of triangles between two range images is sufficient to compute the relative 3D transformation between them. If there are m feature points, then there are $\binom{m}{3}$ possible triangles. Therefore if one image has m feature points, and the other has n, then the number of possible matching triangles is $\binom{m}{3}\binom{n}{3}$. Considering all such possible matching pairs of triangles is computationally infeasible. Our idea is to first compute a Delaunay tetrahedrization of the 3D coordinates of the interest points in each image, and to then only use the triangle faces of the tetrahedrization as possible matches.

The Delaunay tetrahedrization of a set of 3D points is both unique and invariant to viewpoint. We compute it using an efficient algorithm which runs in $O(n \log n)$ expected time [17]. By considering only pairs of triangles in the tetrahedrization as possible corresponding triangles we greatly limit the search space. This is because these triangles are a small subset of all the possible triangles that can be created from a set of 3D interest points.

Consider the Delaunay tetrahedrization of the interest points that come from the overlapping region of the two range images. If these computed interest points are the same in each range image, then the triangles of the tetrahedrization of these 3D points in the overlapping region will also be the same. In practice, the interest points will differ somewhat between images, so the triangles will also differ. Even so, it is usually the case that a majority of the interest points computed from the overlapping regions of each range image are identical. So there is very likely to be at least one matching pair of triangles between the two

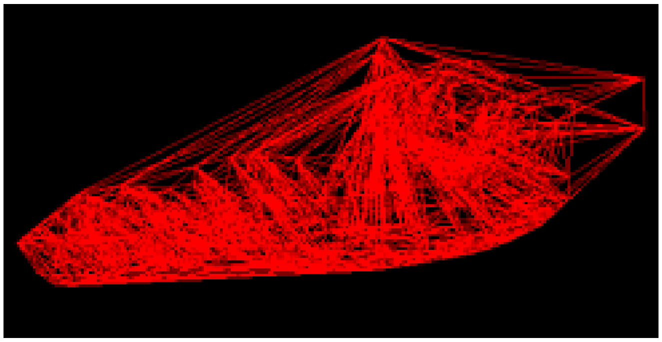


Figure 2: The triangles in the Delaunay tetrahedrization of the 3D co-ordinates of the 2D feature points.

range images in this overlapping region. This makes it feasible to consider only these triangles as potential matches. The triangles in the Delaunay tetrahedrization of the 3D feature points of the duck is shown in Figure 2. The 3D feature points are obtained from the 2D interest points that were shown in Figure 1.

2.3 Compatibility Measures between Triangles

Typically, there are in the order of 250 feature points, and around 5,000 triangle faces in the 3D Delaunay triangulation of the interest points. A brute force algorithm to match these triangles is still not practical, since there are in the order of 10^7 potential matches. To make this triangle correspondence search efficient we make use of the rigidity constraint. This constraint says that the length of each triangle edge is unchanged after applying the transformation that aligns the two range images. Therefore only triangles that have approximately the same edge lengths need be considered as potential matches. By using the appropriate data structures it is possible to efficiently find such matching triangle pairs.

To do this we sort the edge lengths of each triangle, from the largest to the smallest. Each of these lengths is then quantized into k bits so that a triangle can be represented by a $3k$ bit string. Then we sort all the bit strings for the Delaunay triangles in each image. All the triangles that have the same bit string are put together in a linked list. Using this data structure it is easy to find all the triangles with the same bit string in both images. We only consider such pairs of triangles as potential matches, all other triangle pairs can be ignored. If there are a total of t triangles, and their edge lengths are distributed uniformly, then there are on the average $t/(2^{3k})$ triangles that have the same bit string. Of course, this is the ideal case, and in practice the edge lengths are more clustered.

	Type of object			
Compatibility test	Duck	Vase	Exca	Boat
Triangle edge length	118.2	64.0	321.3	273.7
Vertex face normal	15.4	31.5	63.2	94.0
Vertex inter-feature	25.8	50.4	66.6	64.3

Table 1: Comparison of the efficiency of various compatibility tests: ratio of possible matches to accepted matches, the higher the number the more efficient the test.

The value of k in our experiments is usually 5 or 6, and is chosen so that the length represented by the smallest bit is approximately equal to the resolution of the range image.

The total number of possible triangle pairs is the product of the number of Delaunay triangles computed from each range image. The actual number of triangle pairs that have the same bit strings is considerably less. The ratio of the two numbers; the total number of possible triangle pairs over the actual number of pairs that have the same bit string is an indication of the efficiency of the compatibility test. This edge length compatibility test for pruning possible triangle matches is very efficient, as is seen in Table 1. In this table we show the efficiency of the triangle edge length test for the four range image pairs that are discussed in the experimental section.

2.3.1 Compatibility Measures between Feature Points

Each pair of triangles that have approximately the same edge length generates three potentially matching vertices. We now describe some compatibility tests to further prune these matching triangles by checking the compatibility of these matching vertices. We use two such vertex compatibility tests.

We know that each triangle vertex is also a 3D point in the range image where it has a number of local neighbours. An estimate of the surface normal for each 3D point can be computed by fitting a plane through the neighbours in a local window surrounding that point [18]. The angular difference between the local surface normals of any two triangle vertices is invariant to a rigid transformation. So for any two potentially matching triangles this angle difference for each vertex pair should be approximately the same. If any vertex pair does not pass this vertex angle compatibility test it means that the triangles are not com-

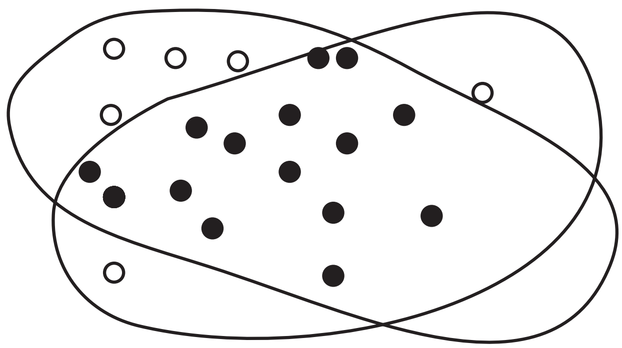


Figure 3: Feature points in the overlapping regions of two range images (the black dots) should have approximately the same inter-point distances.

patible, even if they have the same edge lengths.

The final compatibility measure used to prune triangle vertices is more involved. It is based on the observation that the distances of a feature point to all the other feature points in the overlapping portion of each range image should be approximately the same. In Figure 3 we show the feature points for two overlapping range images. The feature points that are in the overlapping region are drawn in black. Let all the 3D feature points in the first range image be $p_1, \ldots, p_m$, and in the second image be $q_1, \ldots, q_n$.

For every feature point in a single range image we compute the 3d distances of all the other feature points in that image from the given point. Then we sort these distances, from the closest to the farthest. When there are m feature points this will produce a vector $(d_1, ..., d_{m-1})$ of sorted distances. We call such a vector of sorted feature distances with respect to a given feature an inter-feature distance vector. Similarly, in the other image, when there are n feature points, there is an inter-feature distance vector $(d_1, ..., d_{n-1})$ of sorted distances for each of these feature points. The inter-feature distances should be similar for matching 3D feature points. The degree of similarity depends on the overlap of the range images, and on the position of the particular feature point in the overlapping region. Feature points that are closer to the center of the overlapping region are more compatible than others.

To compute this compatibility between any two feature points in the two range images we compare their sorted inter-feature distance vectors. If two features match they are in the an overlapping region (see Figure 3), so some subset of the distances in their inter-feature distance vectors should also be similar. Finding out how many such distances are similar can be done with a simple $O(m+n)$ merge sort of their inter-feature distance vectors. The larger the number of similar distances then the more the likelihood that these two feature points are a valid match. We require that all the vertices of a matching triangle pair have more than a certain number of similar distances in their sorted inter-feature distance vector. If any vertex pair of the matching triangles does not pass this test, then the triangle pair is rejected. Other compatibility measures have been proposed for matching points in dense range image [9, 19]. However, these measures are not suitable for a relatively sparse set of 3D feature points. Also, these approaches have a costly pre-processing phase in which it is necessary to build the data structures required for the matching process [19].

In the same way as for the triangle compatibility test, it is possible to compute the efficiency of these two vertex compatibility test. The test efficiency is the ratio of the number of all possible matching vertex pairs to the number of compatible vertex pairs. The higher this number, the more efficient the compatibility test. In Table 1 we show the efficiency of the local surface normal test and the inter-point feature compatibility test for the four range image pairs that are discussed in the experimental section. Since these tests are performed serially their effect is multiplicative, which means that the vast majority of false matches are pruned.

2.3.2 Computing Transformation

Once both the triangle and vertex assignments passes these compatibility tests the transformation that aligns the two triangles is computed. If this transformation is correct it should also align all the matching interest points in the overlapping regions of the two range images. After the transformation has been applied to align the two images the quality of the match can be determined by counting the number of interest points in the first range image that are within a small distance of an interest point in the second range image. To do this efficiently we use a voxel grid, which has an occupied voxel for every interest point in the second image. The grid resolution is simply set to the resolution of the range image. The total compatibility score for a transformation is the number of interest points that are aligned correctly in this fashion.

2.4 Algorithm Summary

The pseudo-code for the entire algorithm is as follows:

```
for each of the two range images
  -find the interest points in the intensity image;
  -obtain the 3D range value for each such interest point;
  -compute the 3D Delaunay tetrahedrization for all these 3D points;
  -compute the bit length key for each Delaunay triangle;
  -store the Delaunay triangles in a list associated with each key;
endfor

for all the triangles in one image with the same bit length key
  -find all the triangles in the second image with the same key;
  -if the surface normal differences of the triangle vertices match
   and their inter-feature distance vectors are compatible
   then
    -compute the transformation that aligns the two triangles;
    -apply this transformation to all the interest points
        in the first image;
    -count the number of interest points in the second image
        that are close to an interest point the first image;
    -save the transformation that aligns the largest
        number of interest points;
  endif
endfor
```

3 Complexity Analysis

The computation time to find the interest points is proportional to the number of 3D points in the range image. Since finding the interest points requires only simple local operations little time is taken for this step [16]. If l interest points are found in the range image then the time required to compute the Delaunay tetrahedrization is $O(l \log l)$. The computational time for both these steps is much less than the time required for the matching step. To analyze the complexity of the matching step, assume that we have m 3D Delaunay triangles in one image, and n in the other. Also, assume that the key length is k bits for each edge. Then the average number of triangles in the first image with a given key is $m/(2^{3k})$, and in the second is $n/(2^{3k})$. All the triangles with matching keys in both images are potential matches, which means that each bit length key will have on the average $mn/(2^{6k})$ potential matching triangle pairs. Each potential matching triangle pair requires $O(m)$ operations to evaluate the number of interest points aligned by the associated transformation. Since there are (2^{3k}) possible bit strings this implies that the time required on the average to perform the matching step is $m^2n/(2^{3k})$. This analysis assumes that the Delaunay triangles are equally distributed across the entire key set. It also does not take into account the vertex compatibility test, which reduces the number of matches. Nevertheless, it is fair to say that the overall running time is affected by the actual distribution of the edge lengths of the Delaunay triangles. If there are many triangles of almost equal size, then the algorithm will be slower.

There is also the time taken for the two vertex compatibility tests. To compute the local surface normal difference between the vertices is a very simple and fast operation. To compute the inter-feature distance compatibilities for a given vertex pair requires a merge algorithm, which is $O(m+n)$. Here m and n are the number of feature points in both images. The computation of the inter-feature distance compatibility measure requires that we first sort the inter-feature point distances for all feature points, which take $O(n^2 \log n)$ and $O(m^2 \log m)$ time. This sorting operation needs to be done only once before the matching process begins. Computing the compatibility checks between triangle vertices is therefore not a significant computational burden. Our conclusion is that assuming a reasonably uniform distribution of triangle edge lengths, the average running time of the entire process is a low order polynomial function of the number of interest points in both images.

4 Experimental Results

In our experiments, we register overlapping range images of a number of different objects. The rangefinder used to take these images returns both the 3D range data and three channels of registered color intensity data [20]. We convert these R,G,B color channels to hue, intensity and saturation. We then find the interest points in the intensity component and currently ignore the hue and saturation components.

The first two objects in our experiments are a duck and vase. In parts (a) and (b) of Figures 4 and 5 are the computed interest points in the intensity components of the two overlapping range images of each object. Part (c) of both figures shows two renderings of the correctly registered 3D range images. In these renderings, each range image is distinctively shaded, which demonstrates that the two views have been accurately registered. It should be noted that the initial registration obtained by our method has been further refined by a fast ICP algorithm which operates only on the interest points, not the entire image [2]. Note that the two range images of the vase could not be registered correctly using just the 3D data, since they are rotationally symmetric. However, they were successfully registered using the intensity features, which are not rotationally symmetric.

As a further experiment we took all the range images of a number of objects and by hand selected those pairs that had significant overlap and attempted to register only these pairs. We then classified the registration results as successful (very accurate registration), partially successful (accurate enough for an ICP

	Type of object		
Results	Duck	Vase	Head
Success	1	3	3
Partial success	1	1	2
Failure	1	3	2

Table 2: The results of range image pair registration for different objects.

post-processing step to converge), or failed (not close enough for an ICP to converge). Table 2 below shows the results for three objects which have different degrees of texture. It seems that success depends on both the object texture and the amount of overlap. Currently the range image acquisition is done in a fashion which produces around 10% to 20% overlap per image pair. Our registration algorithm requires at least 20% to 30% overlap for success. Given the inherent advantages in automating the registration process it is likely that the data collection process will be modified in the future so that the range images have more overlap.

If there is not sufficient texture on the object, then we can use the texture of the background to aid in the registration. To demonstrate this we printed a pattern of small randomly placed rectangles on a sheet of paper, and placed an object to be scanned on the paper. We have taken a small object, a toy excavator, and rotated it by approximately 90% degrees. Since the object is fixed to the paper, as the object rotates, so does the background pattern. In this case the rotation axis is fixed, but we do not make use of this fact in our algorithm, nor do we assume that the background texture is coplanar. We also do not have self-identifying targets in the background. Instead we use only the corners of this random box pattern as feature points. The experimental results are shown in Figure 6. In parts (a) and (b) of the Figure we see the interest points and in part (c) we see the two registered range images. In this figure there is no distinct shading of the points from each range images, instead all the 3D data points are simply drawn in their final location.

When there is not enough natural texture on the object, or on the background, our registration method will not work. In such cases one possibility is to project texture using, for example, a set of projected dot patterns. For now, we have conducted a simple experiment which shows that the basic idea of texture pattern projection will work. We have drawn by hand on a toy boat without texture some black dots using a marker, in order to simulate a set of projected dots. Our goal is simply to show that the registration process does work when using only these dot patterns on an object that has no significant texture. The results of this experiment is shown in Figure 7. In parts (a) and (b) we see the interest points obtained from the black dots. In part (c) we see the two registered range images drawn as a set of 3D points.

In terms of execution time each registration took on the order of thirty seconds to three minutes on a Silicon Graphics Indigo R4000 workstation. The required time grows with the number of interest points. When there are up to 500 interest points the execution time is still reasonable. Of course, if we add more a-priori constraints on the registration transformation, such as a limited amount of rotation, then the execution time decreases significantly. In all these experiments there was very little tuning of the thresholds of the interest operator. Examination of the feature points in Figures 4 and 5 shows that this operator does not produce identical points of interest in each range image. Since we only require a single Delaunay triangle pair to be in correct correspondence, such differences in the interest points can easily be handled by our approach. By performing an exhaustive search we are very likely to find at least one correctly matching triangle pair. For this reason our method is relatively insensitive to the threshold settings of the interest operator, and the triangle and vertex compatibility tests.

In these examples the two range images have significant rotational differences. Since the algorithm makes no assumptions about the transformation between the two images it can still find the correct transformation even under these conditions. Success does depend on there being sufficient texture and overlap in the range images. While these experiments show the method has promise, more systematic experiments are necessary to accurately quantify the performance.

5 Conclusions

This paper has presented a method of registering range images by finding interest points in the corresponding intensity data of each range image. The basic idea is to triangulate the 3D co-ordinates of these 2D interest points, and then to only match the resulting 3D triangles. Putative matches are pruned using a set of compatibility tests derived from the rigidity constraint. The best match is the one that aligns the largest number of interest points between the two views. It is found using an exhaustive search of the possible matching Delaunay triangle pairs. We believe that exhaustive search is practical when 3D data is available. The fact that a rigid transformation preserves distances (the rigidity constraint) means that simple compatibility tests can be used to prune many false matches. These compatibility tests are very efficient, as was shown in Table 1, especially since their effect is multiplicative. The fact that the rigidity constraint holds for 3D range data is one of the main advantages of processing such data. The rigidity constraint does not hold for 2D data, since the projection operator used to create 2D images does not preserve distances.

Using only 3D interest points, and furthermore using only the Delaunay triangles computed from these interest points to do the matching, greatly reduces the computational requirements of the registration algorithm. We have demonstrated that our method works on a number of experimental examples. We are able to register range image pairs without assuming any prior knowledge of the transformation between these images. To use the automatic registration algorithm there should be sufficient texture in both images, along with at least 20% to 30% overlap between them. This degree of overlap is not difficult to ensure when collecting range images. The next step will be to make the method practical for objects that have no natural texture by using projected texture to generate feature points.

References

[1] P. Besl, "Active, optical range imaging sensors," *Machine Vision and Applications*, vol. 1, no. 1, pp. 127–152, 1988.

[2] P. Besl and N. McKay, "A method for registration of 3-d shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239–256, Feb. 1992.

[3] A. Ashbrook, R. Fisher, C. Robertson, and N. Werghi, "Finding surface correspondence for object recognition using pairwise geometric histograms," in *Computer Vision-ECCV'98*, (Freiburg, Germany), pp. 674–686, June 1998.

[4] R. Bergevin, D. Laurendeau, and D. Poussart, "Registering range views of multipart objects," *Computer Vision and Image Understanding*, vol. 61, pp. 1–16, January 1995.

[5] T. Masuda and N. Yokoya, "A robust method for registration and segmentation of multiple range image," *Computer Vision and Image Understanding*, vol. 61, pp. 295–307, May 1995.

[6] D. Laurendeau, G. Roth, and L. Borgeat, "Optimisation algorithms for range image registration," in *Vision Interface 96*, (Toronto, Ontario), May 1996.

[7] J. Thirion, "New feature points based on geometric invrariants for 3d registration," *International Journal of Computer Vision*, vol. 18, no. 2, pp. 121–137, 1996.

[8] C.-S. Cheng, Y.-P. Hung, and J.-B. Chung, "A fast automatic method for registration of partially overlapping range images," in *International Conference on Computer Vision*, (Bombay, India), pp. 242–248, 1998.

[9] C. Chua and R. Jarvis, "3d free-form surface registration and object recognition," *International Journal of Computer Vision*, vol. 17, 77-99 1996.

[10] J. Feldmar and N. Ayache, "Rigid, affine and locally affine registration of free-form surfaces," *International journal of computer vision*, vol. 18, no. 2, pp. 99–119, 1996.

[11] E. Gagnon, J.-F. Rivest, M. Greenspan, and N. Burtnyk, "A computer assisted range image registration system for nuclear waste cleanup," in *IEEE Instrumentation and Measurement Conference*, (Brussels, Belgium), pp. 224–229, June 1996.

[12] P. Jasiobedski, "Fusing and guiding range measurements with colour video images," in *Proceedinsg International Conference on Recent Advances in 3-D Digital Imaging and Modelling*, (Ottawa, Ontario), pp. 339–347, IEEE Computer Society Press, 1997.

[13] K. Ng, V. Sequeira, S. Butterfield, D. Hogg, and J. Goncalves, "An integrated multi-sensory system for photo-realistic 3d scene reconstruction," in *ISPRS Commission V Symposium: Real-Time Imaging and Dynamic Analysis*, vol. XXXII, pp. 356–363, 1998.

[14] P. Torr, *Motion segmentation and outlier detection*. PhD thesis, University of Oxford, 1995.

[15] Z. Zhang, "Determining the epipolar geometry and its uncertainty: a review," *International Journal of Computer Vision*, vol. 27, March 1998.

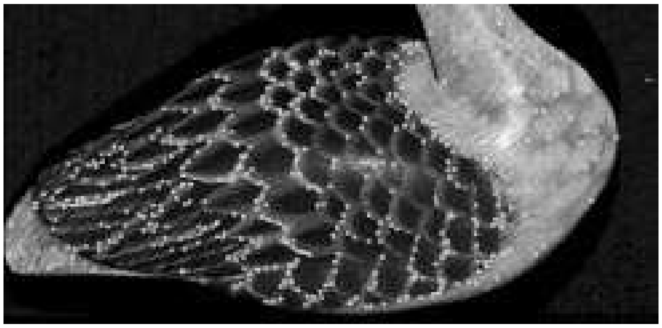
[16] S. Smith and J. Brady, "Susan - a new approach to low level image processing," *International Journal of Computer Vision*, pp. 45–78, May 1997.

[17] K. Clarkson, K. Mehlhorn, and R. Seidel, "Four results on randomized incremental constructions," in *Computational geometry, theory and applications*, pp. 85–121, 1993.

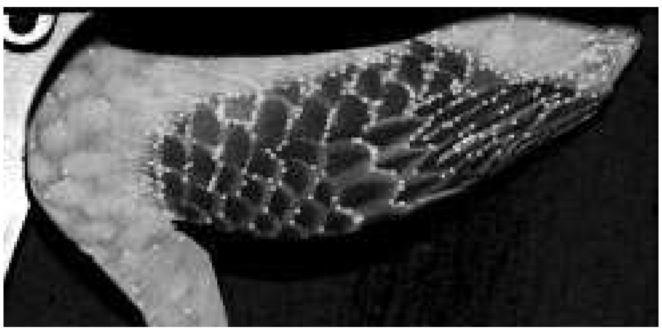
[18] P. J. Besl, J. B. Birch, and L. T. Watson, "Robust window operators," in *Second International Conference on Computer Vision*, (Tampa, Florida), IEEE Computer Society, Dec. 1988.

[19] A. Johnson and M. Hebert, "Surface matching by oriented points," in *International Conference on Recent Advances in 3-D Digital Imaging and Modelling*, (Ottawa, Canada), pp. 121–129, IEEE Press, May 1997.

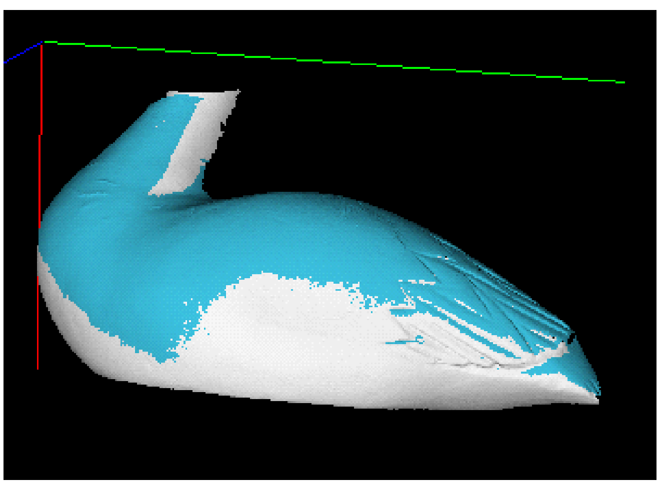
[20] M. Rioux, "Digital 3-d imaging: theory and applications," in *Videometrics III, International Symposium on Photonic and Sensors and Controls for Commercial Applications*, vol. 2350, pp. 2–15, SPIE, 1994.



(a)

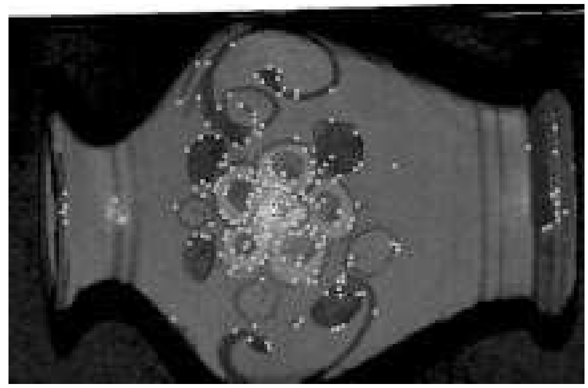


(b)

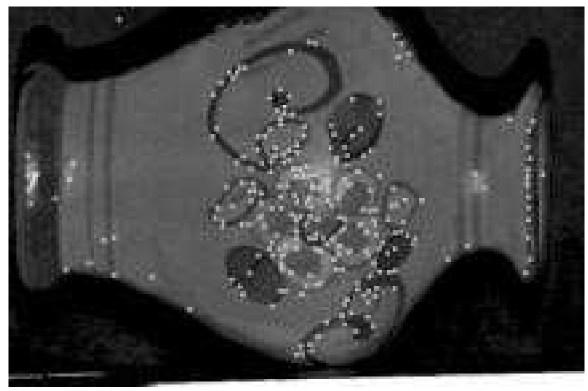


(c)

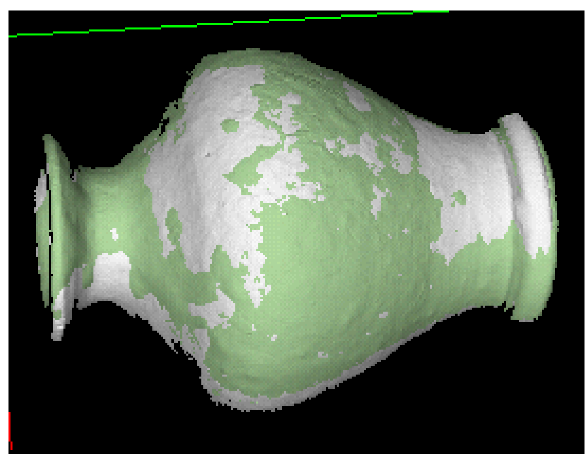
Figure 4: (a) and (b) Feature points found in the intensity component of two different range images of a duck. (c) The two duck range images registered together.



(a)

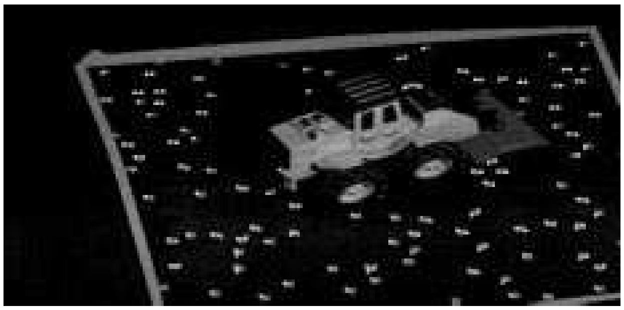


(b)

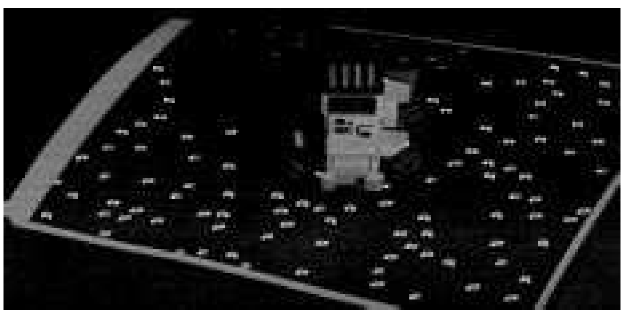


(c)

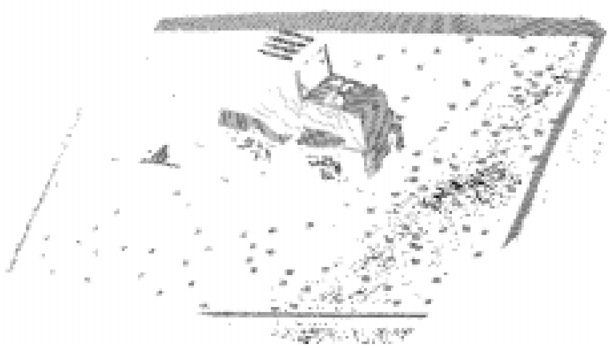
Figure 5: (a) and (b) Feature points found in the intensity component of two different range images of a vase. (c) The two vase range images registered together.



(a)

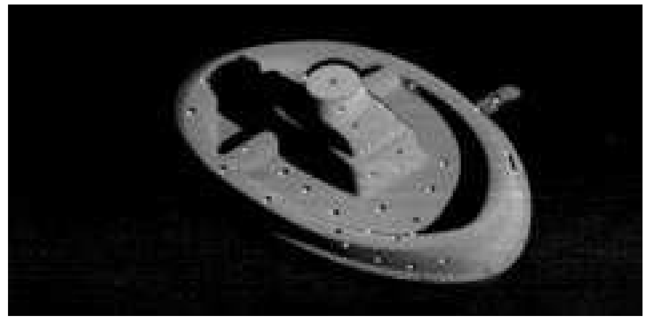


(b)



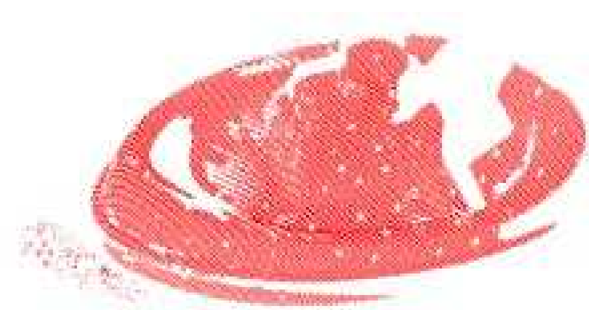
(c)

Figure 6: (a) and (b) Feature points found in the intensity component of two different range images of an excavator model. (c) The two excavator range images registered together.



(a)

(b)



(c)

Figure 7: (a) and (b) Feature points found in the intensity component of two different range images of a boat model. (c) The two boat range images registered together.