MVA2011 IAPR Conference on Machine Vision Applications, June 13-15, 2011, Nara, JAPAN

Alignment of 3D Shape Data by Hashing Sets of Feature Points

Yuka Kohno Osamu Yamaguchi Toshio Sato Bunpei Irie
Toshiba Corporation
Power and Industrial Systems R&D Center
{yuka.kohno, osamu1.yamaguchi, toshio4.sato, bunpei.irie}@toshiba.co.jp

Abstract

This paper presents a method to automatically align a pose of 3D shape data to fit another shape data taken from different viewpoints. One of the difficult issues is to handle shape data which have surface information in different sides due to the difference in viewpoints, and to deal with objects in different scale. We detect local feature points on the two shape data, make potentially corresponding pairs of three feature points, calculate transformation parameters to align the three points, and get optimal alignment parameters by the voting of parameters obtained from the pairs of three points. We used hash table to avoid combinatorial explosion in making the pairs, and used geometric invariants for its key which are calculated from the positions of the points to keep the scale invariance. The method was evaluated with some public data and a set of laser-scanned data, and proved to be effective in alignment of shape data in different angles or scales.

1. Introduction

3D shape data, such as range data and mesh data, help to achieve higher performance of automatic object detection, segmentation, and recognition compared to independent use the conventional 2D image. For the efficient use of 3D shape data, we developed a method to automatically align two shape data taken from different viewpoints. Our goal is to find transformation parameters, composed of rotation, translation, and scaling, which make the surface of an object in one data overlap with the surface of the same object in another data.

One of the difficulties in dealing with 3D shape data taken from different viewpoints is handling loss of surface information in occluded surface or the back of objects. Although the visible surface information is consistent among different range data, the occluded area differs due to different views. Hence some surface areas in one data may have no corresponding area in another data.

Wahl et al. introduced "surflet-pair-relation histogram" representing a whole shape data [1]. It is capable in classification of complete 3D shape model, but is not suitable for incomplete singe-sided data. To deal with such data, some methods have been proposed such as representing shape data by local shape features. Li et al. classified shape data by comparing local surface descriptors using pyramid matching [2][3]. Knopp et al. detected and described local patches by 3D SURF, and classified shape data by voting approach [4]. For alignments of shape data, Gelfand et al. presents an algorithm to pick small number of potentially corresponding feature points using uniqueness of descriptors [5]. Chen et al. introduced local surface descriptor for object recognition, using their correspondence to vote for candidate models [6]. Their methods use small number of potential correspondence, which requires accurate local patch descriptors. However, single-sided data with loss of shape information in different occlusion parts tends to show different behaviors in corresponding points.

Instead of using small number of feature points, Shan et al. and Drost et al. used two-point pair features to estimate object pose [7][8]. Their methods perform well in matching shape data in same size, but lose scale invariance as they use absolute distances.

We also use feature points for the alignment, but use combinations of potentially corresponding pairs of three feature points, or pairs of triplets. Our method obtains the alignment of single-sided shape data taken from different viewpoints, with loss of surface information in different parts, and including objects in different sizes. We use hash table in making the pairs to avoid combinatorial explosion of two triplets, which contributes in process time reduction. To keep scale invariance, and to be able to handle objects in different sizes, we use as the key for the hashing angles and ratios of distance which are defined by the three points, and are invariant under rotation, translation, and scaling transformation.

2. Proposed approach

In this section, we describe the overview of our method. First we get two mesh data as inputs, target data and query data. They are converted from range data captured by stereo camera, range finder, or any other device to capture depth data. To obtain the transformation parameters which align the pose of query data to overlap with target data, we go through the process shown in figure 1.

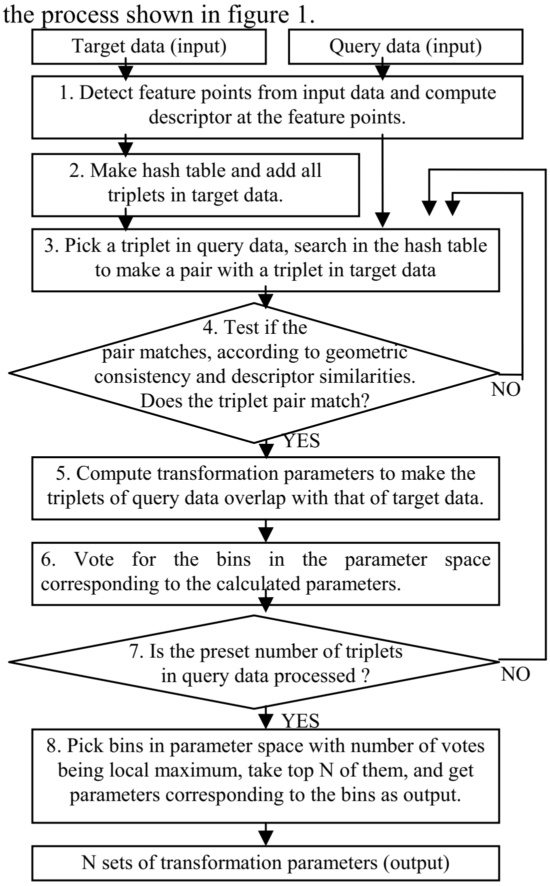


Fig.1 Flowchart of our method

2.1 Feature point detection (step 1)

For feature point detection and the computation of descriptors on mesh data, we used a method proposed by Zaharescu et al. [9]. Their method takes as an input the curvature value on every point on mesh data. The curvature value is then smoothed in various scales to make difference of Gaussian images defined on mesh data, which they call "Mesh DOG". Feature points are detected at the local maximum of Mesh DOG in both the scale space and the neighboring points on the mesh surface. Then, for each feature point, a local coordinate system is defined, and descriptor called "MeshHOG" is calculated which are histograms of curvature gradient plot on the local coordinate system. Figure 2 shows an example of curvature value and the detected feature points.

(a) (b) (c)

Fig.2 (a) Original object. (b) The curvature value on the surface with light colored area representing large value. (c) Detected feature points.

2.2 Making of hash table (step 2)

Using the position and descriptor information of feature points, we make pairs of triplets and calculate their transformation parameters. To avoid combinatorial explosion of two triplets, we use hash table introduced in 2D image matching by Yamaguchi et al. [10]. As the key, the hash table uses angles θ and ratios of distance R=a/b of two line segments, as shown in figure 3, which are invariant under the similarity transformation.

In making of hash table, all triplets in target data are added to hash table according to their key. Each entry contains information about the three points (P_i, P_j, P_k) including their position ($\mathbf{p}_i$, $\mathbf{p}_j$, $\mathbf{p}_k$), normal vectors ($\mathbf{n}_i$, $\mathbf{n}_j$, $\mathbf{n}_k$), and descriptors ($\mathbf{D}_i$, $\mathbf{D}_j$, $\mathbf{D}_k$).

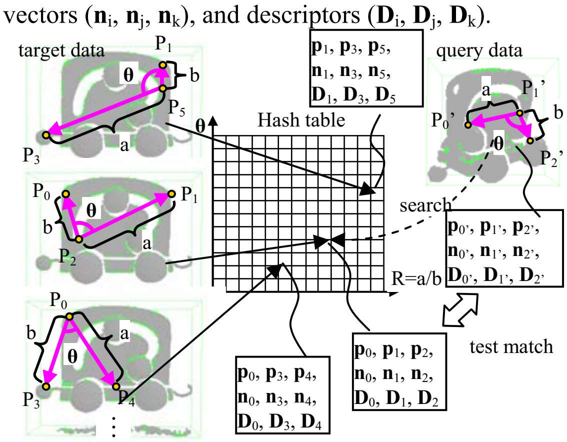


Fig.3 Matching of triplets of points using hash table

2.3 Matching of triplets using hash table (step 3-5)

A triplet in query data is then picked to search for triplets in target data at the corresponding key in hash table. When the hit occurs, the matching of the pair of triplets is tested, according to the similarity of corresponding feature point descriptors, and the similarity of corresponding inner products of two normal vectors within a triplet. If the pair of triplets matches, rotation, translation, and scale parameters are computed which make the total distance between the corresponding points smallest.

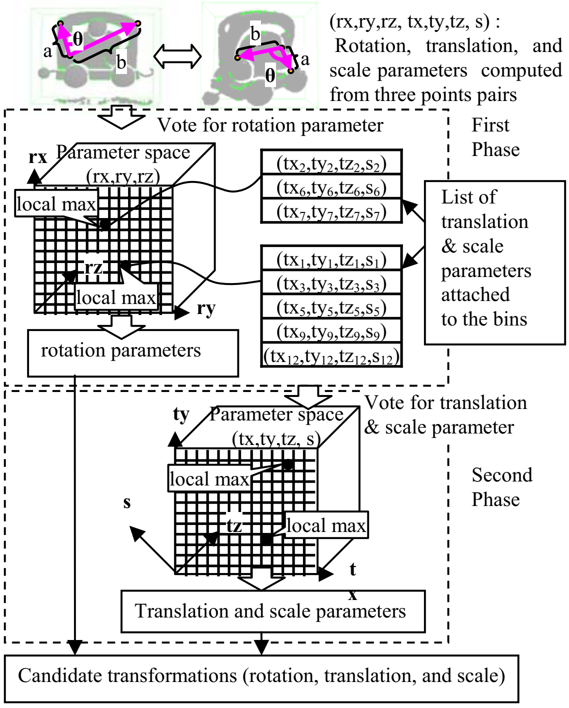


Fig.4 Estimation of pose parameters by voting

2.4 Acquisition of transformation (step 6-8)

The computed parameters are then voted for in the corresponding bins of the parameter space. These processes, including steps 3-6, are done for preset number of three points in mesh data 2. After the voting process is finished, we pick the bins with the number of votes being the local maximum and top N within them, to get the corresponding parameters.

In actual implementation, it is not realistic to allocate parameter space of 7-dimensional array, which would be more than 11TB on the computer memory with our design for range and step size. Instead, we divide the process into two phases, estimation of rotation parameters which requires 750KB, and estimation of translation and scale parameters which requires 5MB. Figure 4 shows the overall process of parameter acquisition. In the first phase, we vote for the rotation parameters and make a list of translation and scale parameters for each bin. Then for the bins in the rotation parameter space with top N local maximum votes, we get the list of translation and scale parameters and vote for them to get the parameters with the most votes. Finally we put together all the parameters to get candidate transformations.

3. Experiments

3.1 Experiment with laser-scanned data

We tested our method using shape data of five objects. The aim of the experiment is to evaluate the alignment of two shape data in different viewpoints. The data are scanned by Minolta laser rangefinder VIVID700, and consist of one of five objects in two viewpoints, as shown in figure 5. Each object is scanned 5 times with its movable parts located in different positions to test the alignment of partially matching data. We made 125 trials for the experiment, each of them having two scan data of the same object, and in different viewpoints. The data contains background noise, shape cut off at the edge of scan area, noise caused by reflection, repeated shape pattern, and are very challenging.

We test our method with various numbers of feature points, 25, 50, and 75. The results were evaluated by the cumulative success rate in top N candidates and the execution time of matching process. The criteria for the success is the error against ground truth data being below 5mm in translation and 10 degrees in rotation, expected to be a good initial pose for iterative closest point (ICP) refinement. The execution time of matching process was measured with Intel® Core™ 2 CPU (2.66GHz) with 2GB RAM and does not include the time for other processes such as feature detection. The results are shown in table 1. The use of hash table reduces the process time dramatically, though the increase rate becomes larger as the number of feature points increases. The success rate was sufficient with enough number of feature points and candidates, though there is a tradeoff between process time and the number of feature points.

For comparison, we also tested alignment by ICP using all surface points, with N initial poses. The first initial pose is the original laser-scanned data, and the rest were set randomly within -30 to 30 degrees in angle and within half the length of object size in translation. The result is shown in table 2. The success rate is roughly equal to that of our methods with 50 points, improving as the number of randomly set initial pose increases. But the process time also increases, in contrast to our method whose process time does not depend on the number of candidates. The success rate was worse for ICP with completely random initial pose, due to the property of the method which depends heavily on the initial pose.

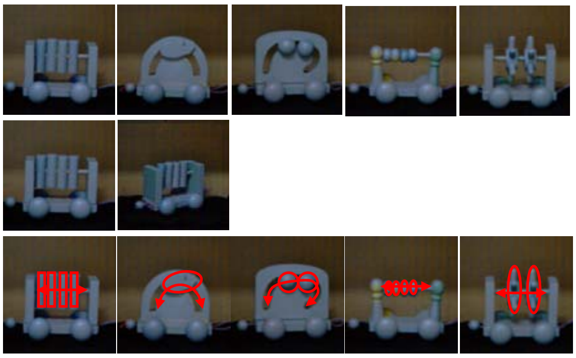


Fig.5 (Top) Five objects used for the experiment. (Middle) Two viewpoints. (Bottom) Movable parts and their range of movement.

Table 1. Cumulative success rate of top N candidates

#of FP	N=1 [%]	N=3 [%]	N=5 [%]	N=10 [%]	time w/ hashing	time w/o hashing
25	30.4	48.8	53.6	59.2	0.72s	21s
50	62.4	80.0	84.8	91.2	1.1s	23min
75	79.2	89.6	90.4	96.8	6.8s	> 4hrs

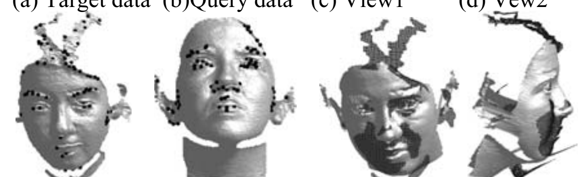
Table 2. Result of ICP method with N initial pose.

	N=1	N=3	N=5	N=10
ICP	68.8%	76.8%	80.8%	91.2%
time	2.72s	5.31s	7.96s	15.2s

3.2 Experiment with public data

We evaluated our method using public data. The results are shown in Figure 6. We picked two Bunny data with different viewpoints from Stuttgart Range Image Database [11] and re-sampled to make them in different scale. The query data is aligned successfully to fit target data in different view and scale. Another data we used is face data looking down and looking up, obtained from GabavDB face database [12]. The result

(a) Target data (b)Query data (c) View1 (d) Vew2



(e) Target data (f)Query data (g) View1 (h) View2

Fig. 6 (a),(e) Target data and (b),(f) Query data with feature points. (c),(g) The result of pose alignment, target data drawn in black mesh, and query data in gray surface. (d),(h) The result seen from another direction.

shows the method is valid for the alignment of face data in different angles.

4. Conclusion

In this paper, we have introduced a method for 3D shape data alignment. By using potentially corresponding pairs of three feature points, our method is applicable to single-sided shape data including objects of different sizes. To avoid combinatorial explosion, we used hash table for picking the pairs of triplets. By using angle and distance ratio as the key, the matching is tolerant to scale variability of objects.

The result of the experiment shows that our method is effective with enough number of feature points, and is superior to naive ICP method with initial pose set randomly. The alignment algorithm is expected to be used for object recognition, identification of shape data, understandings of object structures, or 3D modeling, and to contribute for application such as robot vision, facility inspection, and biometrics.

References

[1] E. Wahl, et. al: "A statistical 3d-shape representation for rapid classification," In 3-D Digital Imaging and Modeling, 2003.

[2] X. Li and I. Guskov: "Multi-scale Features for Approximate Alignment of Point-based Surfaces," *Eurographics Symposium on Geometry Processing*, 2005.

[3] X. Li and I. Guskov: "3D object recognition from range images using pyramid matching," *Proc.ICCV*, 2007.

[4] J. Knopp, et. al.: "Hough Transform and 3D SURF for robust three dimensional classification, *Proc. ECCV*, 2010.

[5] N. Gelfand, et. al.: "Robust global registration," *In Proceedings of the third Eurographics symposium on Geometry processing*, p. 197. 2005.

[6] H. Chen and B. Bhanu: "3D free-form object recognition in range images using local surface patches," *Pattern Recognition Letters*, vol.28, no.10, pp.1252-1262, 2007.

[7] Y. Shan, et. al.: "Linear model hashing and batch ransac for rapid and accurate object recognition." *Proc.CVPR,* 2004.

[8] B. Drost, et. al: "Match Locally: Efficient and Robust 3D Object Recognition, *Proc. CVPR*, 2010.

[9] A. Zaharescu, et. al.: "Surface Feature Detection and Description with Applications to Mesh Matching," *Proc. CVPR*, 2009.

[10] O. Yamaguchi and K. Fukui: "Pattern Hashing - Object recognition based on a distributed local appearance model," *Proc. ICIP*, vol.3, pp.329-332, 2002.

[11] G. Hetzel, et. al.: "3D Object Recognition from Range Images using Local Feature Histograms," *Proc. CVPR*, 2001.

[12] A.B. Morenom et. al.: "A 3D Face Database," *Proc. 2nd COST Workshop on Biometrics on the Internet: Fundamentals, Advances and Applications*, pp. 77-82, 2004.

Intel, and Core are trademarks of Intel Corporation.