ADAPTIVE DUAL AK-D TREE SEARCH ALGORITHM FOR ICP REGISTRATION APPLICATIONS

Jiann-Der Lee[1], Shih-Sen Hsieh[1], Chung-Hsien Huang[1], Li-Chang Liu[1], Chien-Tsai Wu[2,3], Shin-Tseng Lee[2,3], and Jyi-Feng Chen[2,3]

[1]Department of Electrical Engineering, Chang Gung University, Tao-Yuan, Taiwan
[2]Medical Augmented Reality Research Center, Chang Gung Memorial Hospital, Lin-Kou, Taiwan
[3]Department of Neurosurgery, Chang Gung Memorial Hospital, Taiwan

ABSTRACT

An algorithm for finding coupling points plays an important role in the Iterative Closest Point algorithm (ICP) which is widely used in registration applications in medical and 3-D architecture areas. In recent researches of finding coupling points, Approximate K-D tree search algorithm (AK-D tree) is an efficient nearest neighbor search algorithm with comparable results. We proposed Adaptive Dual AK-D tree search algorithm (ADAK-D tree) for searching and synthesizing coupling points as significant control points to improve the registration accuracy in ICP registration applications. ADAK-D tree utilizes AK-D tree twice in different geometrical projection orders to reserve true nearest neighbor points used in later ICP stages. An adaptive threshold in ADAK-D tree is used to reserve sufficient coupling points for a smaller alignment error. Experimental results are shown that the registration accuracy of using ADAK-D tree is improved more than the result of using AK-D tree and the computation time is acceptable.

1. INTRODUCTION

Registrations on computer medical imaging have been used to assistant medical doctors and surgeons for tracking specific locations such as tumors or developing a medical reality environment in last decades. Besl and McKay [3] proposed the Iterative Closest Point algorithm (ICP) which has become a popular registration trend [1][4][5][6][7]. The K-D search method (K-D tree) was used in [3] as the nearest neighbor search algorithm. Greenspan and Yurick [2] proposed the Approximate K-D tree search algorithm (AK-D tree) by excluding the backtracking in K-D tree and giving a more tolerable minimum distance between query and returned point sets. Greenspan and Yurick claimed that the computation time of the best performance using AK-D tree is 7.6% and 39% of the computation time using K-D tree and Elias respectively. It is indicated that AK-D tree is

suitable to outlier data. One weakness of using K-D tree and AK-D tree is that a false nearest neighbor point may be found because one projection plane of k planes is considered on partitioning a k-dimension space into two groups to build the node tree. Based on the efficiency of AK-D tree's computation, we use AK-D tree twice in two different geometrical projection orders for determining true significant coupling points which influence the registration accuracy in later ICP stages. An adaptive threshold in the proposed Adaptive Dual AK-D tree search algorithm (ADAK-D tree) is used to reserve sufficient coupling points for a valid result. Facial point data of the same object from different capture equipments are tested in the experiments. Experimental results show that the registration of using ADAK-D tree is more robust than the registration of using AK-D tree.

This paper is organized as in the following. The details of ADAK-D tree algorithm are presented in Section 2. The experiment results on facial surface data are shown in Section 3. We discuss the future research directions in Section 4.

2. ADAPTIVE DUAL AK-D TREE SEARCH ALGORITHM

In the search structure of K-D tree and AK-D tree, the node tree is generated by splitting a k-dimension space along with a fixed geometrical projecting direction order. For example, 3-D data are continuously split into two groups by projecting data into x, y, and z axes iteratively to build the node tree. Because only one projection direction or a projection plane of k-dimensions is considered to split the space each time, two truly nearest neighbor points may be located at two far sub-root nodes in a node tree. The backtracking in K-D tree reduces this error but it takes massive runtime. To a true geometrical nearest-neighbor point, the same point should be returned in any geometrical projection plane order when using AK-D tree or K-D tree. A simple example is shown in Fig. 1. Four points as P1, P2, P3 and P4 are split by the x-y axis order first in Fig. 1(a). A given query point Q1 which is really close to P2 recognizes

177

P4 as its closest point in Fig. 1(a). On the other hand, if 4 points are split by the y-x axis order, then the Q1's closest point is P2.
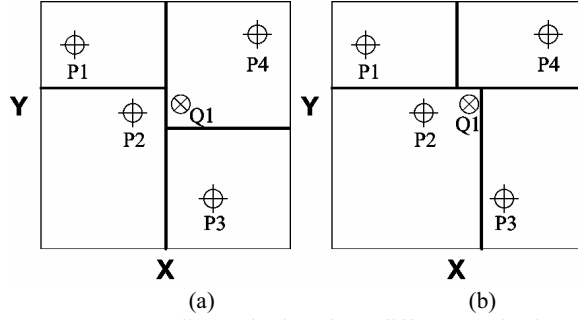

FIGURE 1. 2-D split results based on different projection axis orders. (a) the split result based on the x-y axis order, and (b) the split result based on the y-x axis order

Based on this assumption, we utilize AK-D tree twice in two projection axis orders of "x, y, z" and "z, y, x" to examine the queried results. AK-D tree is used here because of its runtime efficiency. If two returned queried results using AK-D tree in two different projection plane orders are very close, then this query point and the returned queried point are reserved as significant coupling points, otherwise this query point is rejected.

ADAK-D tree is a 5-step process to search and determine significant coupling points used as control points in ICP. In each iteration in ICP, the nearest neighbor points are found in the following 5 steps. Assume that there is a minimum 3-D rectangular box C to contain the complete floating data and M is the area of the biggest plane of C. The initial P is the amount of total floating points.

Step 1. The insert tree of reference/model data is built as Database 1 by using AK-D tree in the projection axis order of "x, y, z" iteratively.

Step 2. The insert tree of reference/model data is built as Database 2 by using AK-D tree in the projection axis order of "z, y, x" iteratively.

Step 3. The threshold T is computed by Eq. 1.

$$T = \frac{M}{P} \tag{1}$$

Step 4. To a query point from the floating data set, if the distance between two returned queried points from Database 1 and Database 2 is smaller than the distance T, this query point and the returned queried point from Database 1 are reserved as a pair of significant coupling points.

Step 5. After that all floating points are queried, P is updated by the reserved pair number.

The reserved significant points after using ADAK-D tree are coupling points [6] to compute the best translation and rotation parameters in later ICP stages. Avoiding falling into an extremely bad local minimum in ICP because of insufficient coupling points, T in Step 3 is automatically adjusted by the previous iteration information. If the situation without sufficient coupling points happens, i.e. P decreases in this iteration, then T increases, which cause the increase of P in the next iteration.

3. EXPERIMENTAL RESULTS

Facial surface data from laser scan and face-camera equipments were used to examine the performances of using AK-D tree and ADAK-D tree in the synthetic experiments. Laser-scan facial surface data were 11291 points captured by LinearLaser (LSH II 150) made by 3dfamily Technology Co.. Face-camera facial surface structured data were 4255 points captured by DigiFace Face Camera made by 3dfamily Technology Co.. In the synthetic experiments, the reference data were translated and rotated to generate variant floating data. In the cases of (a), (b) and (c) as in Table 1 and Table 2, the reference data were translated by 10 mm along x, y, and z axes respectively to generate 3 floating data sets. In the cases of (d), (e) and (f), the reference data were rotated by 5 degrees around X, Y and Z axes respectively to generate the other 3 floating data sets. To measure the performance, root-square-error (RMS) distance [8] is the measurement unit. The numerical comparisons of registration results of laser scan facial surface data and face-camera surface data are shown in Table 1 and Table 2 respectively. The visual registration results in Table 1 of using AK-D tree and ADAK-D tree are shown in Figure 2 and Figure 3 respectively.

TABLE 1. The comparison of registration results of laser scan facial surface data.

	AK-D tree method [2]		ADAK-D tree method	
	RMS (mm)	Computation time (second)	RMS (mm)	Computation time (second)
(a)	16.536	2.25	0.197	5.5
(b)	6.474	1.734	0.196	5.531
(c)	2.54	1.984	0.197	5.75
(d)	2.551	2	0.198	5.797
(e)	2.538	2	0.196	5.64
(f)	2.527	1.985	1.062	5.656

TABLE 2. The comparison of registration results of face-camera surface data.

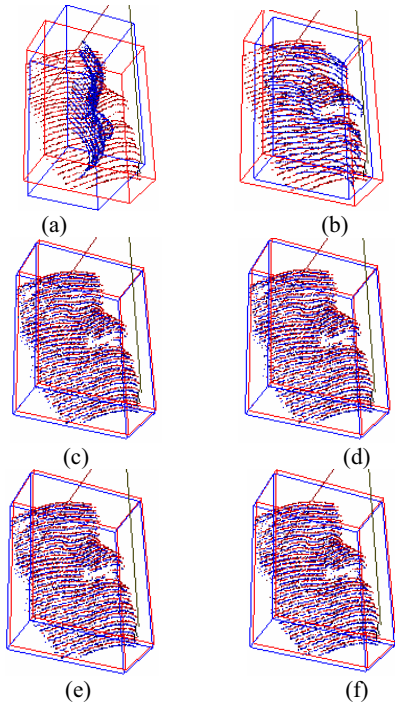|	AK-d tree method [2]		ADAK-d tree method	
	RMS (mm)	Computation time (second)	RMS (mm)	Computation time (second)
(a)	1.584	3.141	0.198	4.375
(b)	1.502	3.156	0.192	4.329
(c)	1.387	3.14	1.64	4.39
(d)	1.71	3.156	1.572	4.453
(e)	1.71	3.172	0.196	4.375
(f)	1.531	3.125	0.401	4.329

FIGURE 2. The visual registration results using AK-d tree on laser scan surface data in Table 1. (a) the visual result in Table 1(a), (b) the visual result in Table 1(b), (c) the visual result in Table 1(c), (d) the visual result in Table 1(d), (e) the visual result in Table 1(e), and (f) the visual result in Table 1(f).
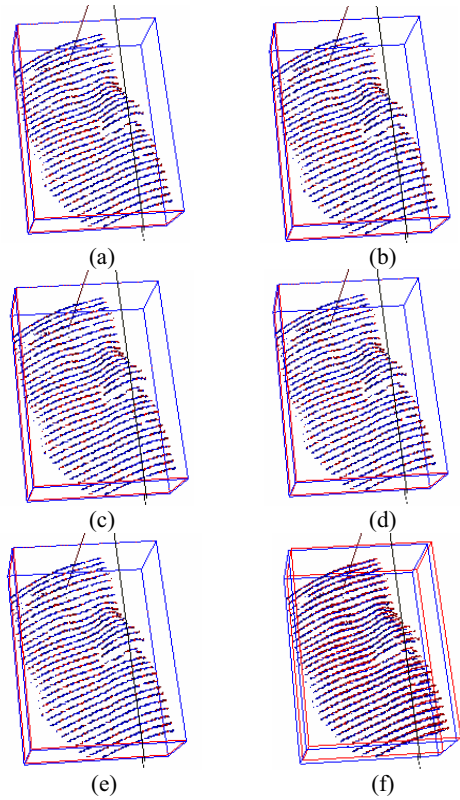

FIGURE 3. The visual registration results using ADAK-D tree on laser scan facial surface data in Table 1. (a) the visual result in Table 1(a), (b) the visual result in Table 1(b), (c) the visual result in Table 1(c), (d) the visual result in Table 1(d), (e) the visual result in Table 1(e), and (f) the visual result in Table 1(f).

About the computation time in the experiments, face-camera data use more bits on storage and computation than laser scan data, and the more bits result in consuming more runtime. The RMS of registration result of using AK-D tree in Table 1(a) is high because the solution is trapped into a local minimum. In Table 1, the computation time of using ADAK-D tree is 2 to 3 times the computation time of using AK-D tree because ADAK-D tree requires extra time on Step 3 and 4.

In the third experiment, CT facial surface data and laser scan facial surface data are registered. CT facial surface data are extracted from CT data first. The model data are CT facial surface data with 17876 points as shown in Fig. 4(a), and the floating data are laser scan facial surface data with 9889 points as shown in Fig. 4(b). The RMS of the registration result using AKD-tree as in Fig. 4(c) is 2.475 mm and the runtime is 3.219 seconds. The RMS of the registration result using ADAK-D tree as in Fig. 3(d) is 1.163 mm and the runtime is 4.438 seconds.
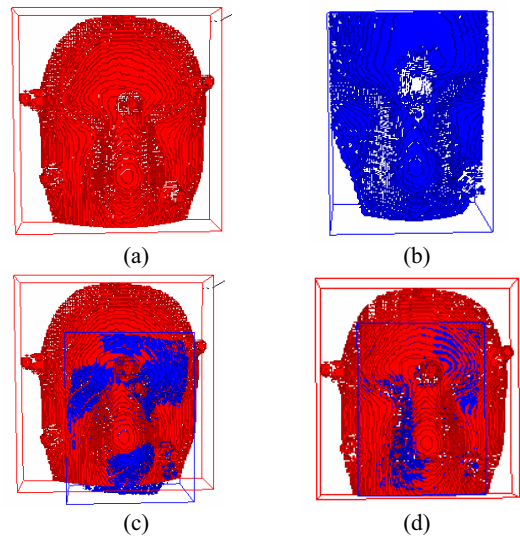

FIGURE 4. The visual registration results from laser scan facial surface data to CT facial surface data. (a) the CT facial surface data, (b) the laser scan facial surface data, (c) the registration result using AK-D tree, and (d) the registration result using ADAK-D tree.

From the experiment results, ADAK-D tree has more robust registration results for an ICP registration rather than AK-D tree in most cases.

4. CONCLUSIONS AND FUTURE DIRECTIONS

A novel nearest neighbor search algorithm for determining significant coupling points has been presented. This proposed method utilizes AK-D tree twice in different projection direction orders to determine the true nearest neighbor point. An adaptive threshold for a true nearest neighbor point matins sufficient control points and enhances the robustness. From the experimental results of facial point data, the ADAK-D tree method has more robust registration accuracies than the AK-D tree method in most cases, and the runtime of using ADAK-D tree doesn't always linearly increase because fewer significant coupling points are used in later ICP stages. In the current experiments, the operation time to capture facial surface data requires 5 to 15 minutes according to desired resolutions, and this period is not preferred in a practical situation. In the future, more surface data captured from variant equipments will be tested. For example, we like to perform the registration between pre-stored CT data and the outline data captured from a 3-D digitizer in a short period to compute the coordinate relation between coordinates of an indoor operation space and a medical virtual environment. The final purpose of this study is to assist medical doctors on searching desired information immediately within pre-stored medical imaging during an operation simulation.

ACKNOWLEDGEMENTS

REFERENCES

[1] D. Chetverikov, D. Stepanov and P. Krsek, "Robust Euclidean Alignment of 3D Point Sets: The Trimmed Iterative Closest Point Algorithm," *Image and Vision Computing*, vol. 23, No. 3, 2005, pp. 299-309.

[2] M. Greenspan and M. Yurick, "Approximate K-D Tree Search for Efficient ICP," *Proceedings of Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM)*, pp. 442-448, 2003.

[3] P.J. Besl and N.D. McKay, "A Method for Registration of 3D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14(2), pp. 239-256, 1992.

[4] S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm," *Proceedings of Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM)*, pp. 145-152, 2001.

[5] S.M. Bhandarkar, A.S. Chowdhury, Y. Tang, J. Yu and E.W. Tolllner, "Surface Matching Algorithms Computer Aided Reconstructive Plastic Surgery," *Proceedings of IEEE International Symposium on Biomedical Imaging: Macro to Nano*, vol. 1, pp. 740-743, 2004.

[6] T. Jost and H. Hugli, "A Multi-resolution ICP with Heuristic Closest Point Search for Fast and Robust 3D Registration of Range Images," *Proceedings of Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM)*, pp. 427-433, 2003.

[7] V. Patoglu and R.B. Gillespie, "A Closest Point Algorithm For Parametric Surfaces With Global Uniform Asymptotic Stability," *First Joint Eurohaptics Conference and Symposium On Haptic Interfaces For Virtual Environment And Teleoperator Systems (WHC 2005)*, pp. 348-355, 2005.

[8] V. Zagrodsky, V. Walimbe, C.R. Castro-Pareja, J.X. Qin, J.M. Song, and R. Shekar, "Registration-assisted Segmentation of Real-time 3-D Echocardiographic Data Using Deformable Models," *IEEE Trans. On Medical Imaging*, vol. 24, No. 9, Sept. 2005, pp. 1089-1099.