Morphological Iterative Closest Point Algorithm

C. A. Kapoutsis, C. P. Vavoulidis, and I. Pitas

***Abstract*—This work presents a method for the registration of three-dimensional (3-D) shapes. The method is based on the iterative closest point (ICP) algorithm and improves it through the use of a 3-D volume containing the shapes to be registered. The Voronoi diagram of the "model" shape points is first constructed in the volume. Then this is used for the calculation of the closest point operator. This way a dramatic decrease of the computational cost is achieved.**

***Index Terms*—Iterative closest point algorithm, Voronoi tesselation.**

I. Introduction

The iterative closest point (ICP) algorithm proposes a solution to a key registration problem in computer vision [1]: given a "model" three-dimensional (3-D) shape and a "data" 3-D shape, estimate the optimal translation and rotation that register the two shapes by minimizing the mean square distance between them. An important application of this algorithm is to register actual data sensed from a 3-D object with an ideal 3-D model. It is also useful in multimodal 3-D image registration in medical imaging (e.g., between NMR and CT volumes), in the shape equivalence problem as well as in estimating the motion between point sets when the correspondences are not known.

A crucial drawback of the ICP algorithm is the high computational complexity of the closest point operator. The morphological ICP algorithm solves this problem by using a 3-D array representing a volume in $\mathcal{Z}^3$ and by constructing the Voronoi diagram of the model points within this volume, by using the morphological Voronoi tesselation method. Then, the ICP algorithm is employed, with distance calculations substituted by simple array references.

A. Iterative Closest Point Algorithm

In the ICP algorithm, a data shape P is registered to be in best alignment with a model shape X. Both shapes must be decomposed first into point sets. Let N_p, N_x be the number of points in the shapes. Then P and X are, respectively, the N_p-tuple $P = (\vec{p}_1, \vec{p}_2, \cdots, \vec{p}_{N_p})$ and the N_x-tuple $X = (\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_{N_x})$. The closest point operator is denoted by C. Equation $Y = C(P, X)$ means Y is an N_p-tuple $(\vec{y}_1, \vec{y}_2, \cdots, \vec{y}_{N_p})$ of points of X, such that $\vec{y}_i$ is the point of X closest to $\vec{p}_i$.

The least squares quaternion operation is denoted by Q [1]. Given P and Y, equation $(\vec{q}, d_{ms}) = Q(P, Y)$ means $\vec{q}$ is the registration vector that best aligns P with Y. And d_{ms} is the point matching mean square error in this alignment. Notation $\vec{q}(P)$ denotes P after transformation with $\vec{q}$.

The ICP algorithm follows.

1) Shapes X, P, a tolerance $\tau > 0$ and an initial registration $\vec{q}_{\mathrm{i}}$ are given.

2) Initialize by setting $P_0 = P$, $\vec{q}_0 = \vec{q}_{\mathrm{i}}$ and $k = 0$.

3) a) Compute the closest points: $Y_k = C(P_k, X)$.
   b) Compute the registration: $(\vec{q}_k, d_k) = Q(P, Y_k)$.
   c) Apply the registration: $P_{k+1} = \vec{q}_k(P)$.
   d) If $(d_{k-1} - d_k) < \tau$, terminate. Else, increase k and go back to Step 3a.

The algorithm guarantees that a local minimum of a mean square objective function is found [1]. This minimum depends on the initial registration $\vec{q}_{\mathrm{i}}$ and may not be the global one. In the typical case, many runs of the algorithm are needed, each one with an appropriately chosen different initial registration. Finally, the registration vector that corresponds to the smallest of all local minima is selected.

B. Morphological Voronoi Tessellation Algorithm

Voronoi tessellation is a classical topic in computational geometry [13]–[15]. For our purposes, the Voronoi diagram has to be constructed in a volume in $\mathcal{Z}^3$. Thus, mathematical morphology [2]–[6] methods are used [7], [10]–[12].

The basic data structure used by the algorithm is a 3-D integer array representing the volume to be tesselated. Each item in the array corresponds to a voxel of the volume and contains the index of the Voronoi region to which the voxel has been assigned. Initially, all voxels (i.e., the respective array items) are set to zero except for those that correspond to the seeds of the tesselation, which contain the proper index. At each iteration, every region grows by one voxel toward all directions and the newly appended voxels accept the correct indices. The growth of each region is supported by a queue, much like the way a queue supports a breadth first traversal of a graph. Initially, every queue contains only the seed of the respective region.

During this growing process, collisions of neighboring regions occur at voxels that reside close to the region borders. They are resolved by explicitly calculating the distances from those voxels to the corresponding seeds.

The algorithm terminates when all queues have got empty. Then, all voxels in the volume have been assigned to a Voronoi region.

II. Morphological ICP Algorithm

The computation of the closest point operator in the ICP algorithm is performed by means of the brute force method: for each data point calculate the distances from it to all model points and select the model point that corresponds to the smallest distance. Clearly, this is an $O(N_p N_x)$ procedure, with Euclidean distance calculations as the elementary operations.

Now suppose that a portion of the $\mathcal{Z}^3$ volume that contains both the model shape and the data shape is available and that it has already been tesselated with respect to the model shape. Finding the model point closest to a given data point would be a matter of simple reference: an access to an item of the 3-D array that represents the volume. Therefore, the complexity of the closest point operator would decline to $O(N_p)$, with the elementary operations now being table look-ups, which are much faster than Euclidean distance calculations.

This is exactly the novel approach of the morphological ICP algorithm, compared to the classical ICP algorithm. Provided that the proper portion of the $\mathcal{Z}^3$ is selected, the algorithm first tesselates the volume using the morphological Voronoi tesselation method, then employs the ICP algorithm using the efficient version of the closest point operator implementation.

Manuscript received October 20, 1997; revised February 2, 1999. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dmitry B. Goldgof.

The authors are with the Department of Informatics, Aristotle University of Thessaloniki, GR-54006 Thessaloniki, Greece (e-mail: pitas@zeus.csd.auth.gr).

Publisher Item Identifier S 1057-7149(99)08763-1.

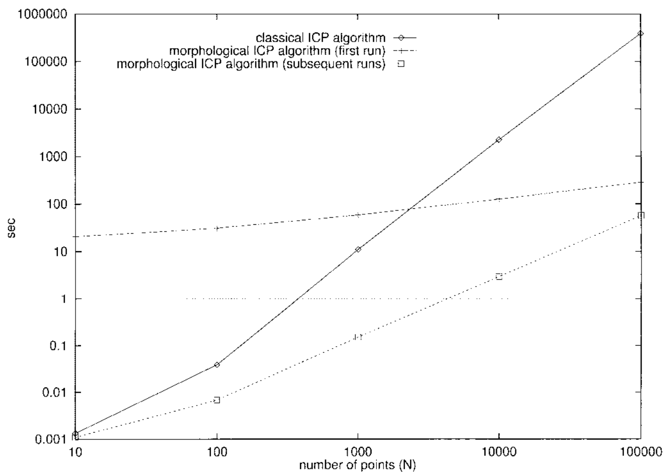

Fig. 1. Time performance of the classical and morphological ICP algorithms.

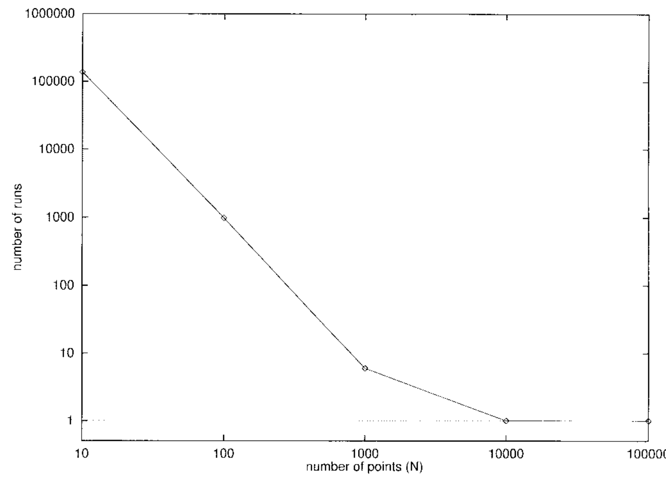

Fig. 2. Minimum number of runs for preferring the morphological ICP algorithm.

To establish the efficiency of this approach, we have to argue on the following three issues.

The first one is whether the integer arithmetic introduced by the $\mathcal{Z}^3$ volume affects the validity of the algorithm. The answer, given by the experimental results (see Section III), is negative. It is though possible that the same initial registration will lead the morphological and the classical ICP algorithms to different local minima. However, this is a rather unusual effect and, should adequately many initial registrations are tested, it is unlikely to prevent the discovery of the global minimum.

The second issue regards the memory requirements of the method. If w, h and d are the volume dimensions in $\mathcal{Z}^3$, the method needs at least $\lceil whd \log_2(N_x + 1)/8 \rceil$ bytes of RAM. For the typical case where $w = h = d = 100$ and $N_x = 1000$, this is approximately 1.2 MBytes, a significant but tolerable requirement. The additional amount of memory needed by the queues that support the growing of the regions can be kept relatively low, especially if a dynamic queue implementation is selected.

Finally, there is a question on the time needed for the volume tesselation. According to the experimental measurements, the computational complexity of this task is of the order $O(N_x^a)$, for some positive constant a which depends on the size of the volume. For example, in the case of the experiments presented in Section III, where a volume of 100 ×100 × 100 voxels has been used, $a = 0.3$. This can be easily deduced from Fig. 1, where the upper dashed curve practically shows the time spent for the tesselation. Therefore, a is equal to the slope of this curve.

In addition, Fig. 1 proves that the time consumption for the Voronoi tesselation is so large that renders the morphological ICP algorithm a poor choice for all one-run cases where $N_x < 0.0025whd$. Nevertheless, it is most likely that

- the ICP algorithm will have to run repeatedly for the same model and data shapes (with a different initial registration each time) till the global minimum is detected; and/or
- many data shapes will have to be matched with the same model shape.

In these cases, the tesselation can be performed only once and the same tesselated volume may be used by all runs of the algorithm. Taking also into consideration that typically N_x will be greater than 0.25% of the total number of volume voxels, we conclude that the morphological ICP algorithm is expected to outperform the classical one in most practical situations.

The above conditions characterize the typical case in 3-D medical imaging, where high resolution 3-D object representation is required and, at the same time, there is the need for an efficient method that matches a sensed 3-D object with a 3-D model which contains many points and is taken out of a fixed, known in advance data base. Experiments on such real-world cases have been performed. Their results are in par with those of the simulation study presented in Section III.

III. Experimental Results

In the experiments described in this section we have used a volume of 100 × 100 × 100 voxels centered at the origin. The number of data points has been equal to that of model points, namely $N_p = N_x = N$. Five cases for N have been examined: $N = 10$, 10^2, 10^3, 10^4 and 10^5. That is, 0.001%, 0.01%, 0.1%, 1%, and 10% of the number of voxels in the volume, respectively.

For each test case, N model points have been generated and were uniformly distributed within the volume. The data point set was then produced from the model point set via successive rotations of the latter around the x-axis $(111°)$, the y-axis $(-37°)$ and the z-axis $(-69°)$. No translation has been employed, because its impact to the experiment is minimal: the first iteration of the algorithm always translates the data point set so that its "center of mass" coincides with that of the model point set (this is an immediate consequence of the definition of the least squares quaternion operation Q). Care has been taken so that all points fall within the volume of size 100 × 100 × 100. The same was true for all intermediate point sets produced by the registration of the initial data point set. The classical and the morphological ICP algorithms were tested on a Silicon Graphics Indy MIPS R4400 200MHz workstation running IRIX 5.3.

Fig. 1 displays the results of the time measurements. Both axes are logarithmic. The continuous line shows the performance of the classical ICP algorithm. The dashed ones refer to the morphological ICP algorithm. The upper dashed curve shows the total time needed for the tesselation and the ICP algorithm. The lower dashed curve shows the computation time for the ICP algorithm only. Notice that tesselation time is much larger than the time needed for the ICP part.

We see that, if only one run is needed, the morphological ICP algorithm is preferable only in cases having many model/data points $(N > N^* \approx 2500)$. For $N = 10\,000$, the proposed algorithm is close

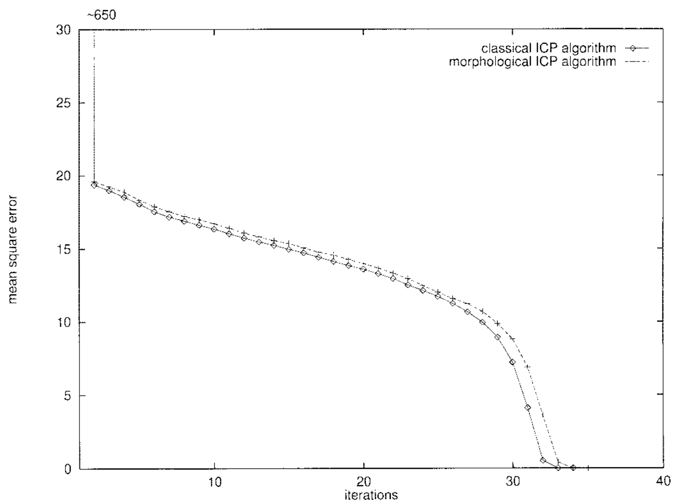

Fig. 3. Progress of the point matching mean square error for the test case $N = 1000$.

to three orders of magnitude faster than the classical one. However, if the tesselated volume is available, the morphological ICP algorithm is superior in all cases: it is one to three orders of magnitude faster than the classical algorithm.

For each test case, there is a minimum number of runs over which the morphological ICP algorithm becomes the best choice. Fig. 2 presents this number for all cases. When $N = 1000$, six runs are enough to make the morphological ICP algorithm outperform its classical counterpart.

Finally, in order to convince about the insignificant effect of the integer arithmetic on the validity of the algorithm, we display in Fig. 3 an example of the progress of the point matching mean square error during the iterations of the algorithm. The two lines almost coincide with each other. The example is taken from the $N = 1000$ test case.

IV. Conclusion

The morphological ICP algorithm is a strong and fast method for the registration of actual data sensed from a 3-D object with an ideal 3-D model. It is faster than the classical ICP algorithm, especially in cases where multiple runs of the algorithm are required and/or a lot of data/model points are involved. Since these cases are the most likely to occur, the morphological ICP algorithm is expected to be much more useful than the classical one.

References

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 239–256, 1992.

[2] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*. Boston, MA: Kluwer, 1990.

[3] I. Pitas, *Digital Image Processing Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[4] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley, 1992.

[5] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic, 1982.

[6] L. Vincent, "Graphs and mathematical morphology," *Signal Process.*, vol. 16, pp. 365–388, 1989.

[7] I. Pitas, "Performance analysis and parallel implementation of Voronoi tessellation algorithms based on mathematical morphology," in *Advances in Digital and Computational Geometry*, R. Klette, A. Rosenfeld, and F. Sloboda, Eds. Berlin, Germany: Springer-Verlag, 1998, pp. 227–254.

[8] G. Borgefors, "Distance transformations in digital images," *Comput. Vis., Graph., Image Process.*, vol. 34, pp. 344–371, 1986.

[9] J. E. Mazille, "Mathematical morphology and convolutions," *J. Microscopy*, vol. 156, pp. 3–13, 1989.

[10] C. Kotropoulos, I. Pitas, and A. Maglara, "Voronoi tessellation and Delaunay triangulation using Euclidean disk growing in $\mathcal{Z}^2$," in *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, 1993.

[11] F. Y.-C. Shin and O. R. Mitchell, "A mathematical morphology approach to Euclidean distance transformation," *IEEE Trans. Image Processing*, vol. 1, pp. 197–204, Apr. 1992.

[12] Q. Z. Ye, "The signed Euclidean distance transform and its applications," in *Proc. 9th Int. Conf. Pattern Recognition*, 1988, pp. 495–499.

[13] F. P. Preparata and M. I. Shamos, *Computational Geometry*. Berlin, Germany: Springer-Verlag, 1985.

[14] G. T. Toussaint, "Pattern recognition and geometrical complexity," in *Proc. 5th Int. Conf. Pattern Recognition*, 1980, pp. 1324–1347.

[15] D. T. Lee, "Two dimensional voronoi diagram in the L_p-metric," *J. ACM*, vol. 27, pp. 604–618, 1980.

[16] B. J. H. Verwer, P. W. Verbeek, and S. T. Dekker, "An efficient uniform cost algorithm applied to distance tranforms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 425–429, Apr. 1989.