Robustness to initial conditions in registration of range images using the ICP algorithm

Andres Restrepo Specht *   Michel Devy *   Angel Sappa †
∗ LAAS-CNRS
7 avenue du Colonel Roche, 31077 Toulouse Cedex 04
† RTS Advanced Robotics Ltd.
Gilchrist Rd, Northbank Industrial Park, Irlam, Manchester, M44 5AY, UK
E-mails : michel@laas.fr, arestrep@laas.fr, sappa@ieee.org

***Abstract*—This paper presents a study about the robustness of the classical Iterative Closest Point (ICP) algorithm in the initial condition selection using two different point extraction methods from the point clouds. The ICP method allows to register two point clouds, here extracted from two range images acquired from different view points on a same scene; registering two images consists in estimating the 3D transform T (rotation and translation) between the two sensor positions. This algorithm requires (1) the selection of some discriminant points, at first extracted from every range images and then matched by the ICP method; (2) an initial estimate of the transform T. It is well known that this method is very sensitive to the selection of this initial estimate. This paper presents an analysis about the ICP robustness with respect to this selection: on several images, all possible initial solutions have been applied in order to measure the convergence bassin around the true solution.**

I. Introduction

The application of 3D modelling are very numerous: medicine -body modelling-, architecture -building inspection-, industry -reverse engineering, inspection of metal parts after a blending process-, art -sculpture modelling for Internet display-,... It exists now very accurate sensors [6], [12] dedicated to the acquisition of range images on environments or on objects. The construction of a 3D model requires several steps: data acquisition, data registration, generation of a dedicated representation depending on the application, and if needed, perception planning to determine when the built model is complete and how to control the sensor (Best next view point, best sensor configuration, ...) in order to acquire more images.

In this paper only the registration function is considered. The registration could be made on different representations extracted from the range images to be registered. The more classical method proposed initially in [5], is the *Iterative Closest Point* (ICP) method, which takes as inputs, two clouds of points expressed in two different reference frames, and provides as result, the 3D transform **T** between these reference frames. Many other registration methods have been proposed, working on more compact representations than a cloud of points [7] or selecting discriminant control points that are matched at first in order to have a good initial estimate of the **T** transform [3]. We have proposed in [10] an edge based representation that could be extracted from the original range images in order to limit the size of the cloud of points given as inputs to the ICP function. In [9], we have compared this edge based representation with the classical triangular mesh representation that is used generally to deal with the registration step; ICP looks for matchings between the edge points in the former, between mesh vertices in the latter. This paper is focused on a specific problem: using either the mesh registration or the edge registration, how to select the initial estimate of the transform **T**? The ICP method requires this estimate in order to find the first matchings, and then to refine this **T** estimate along an iterative process. It is well known that the ICP convergence depends mightily on this first estimate. In the section II, the pre-processing methods (edge point extraction or adaptive mesh generation) are summarized and the main characteristics of our ICP implementation are described. Then in the section III, an exhaustive analysis of all possible initial solutions is presented: the registration methods can be characterized by the proximity of the convergence bassin to the exact solution for the **T** transform, or by the number of initial solutions that lead to ICP convergence.

This work has been carried out as part of the CAMERA project (CAd Modelling of Built Environments from Range Analysis). CAMERA is an EC funded TMR network (ERB FMRX-CT97-0127)

II. The registration algorithm

This section describes the techniques used to extract a compact set of points from every range images to be registered. Then the ICP algorithm is summarized. Examples are presented using range images obtained either from the Stanford 3D scanning Repository [11], or from the OSU (MSU/WSU) data base [8].

Fig. 1. A single view of a sculptured object represented by triangular meshes (left) and by an edge based representation (right).

A. Mesh registration

A public surface refinement algorithm has been used (GTS Open Source [1]). This fine-to-coarse algorithm is applied over a dense triangular mesh obtained from all the points contained in the given range image. The initial triangulation is trivially computed by joining all range image points horizontally and vertically and then dividing each quadrilateral cells into two triangles. The figure 1(left) shows an example of a sculptured object represented by triangular meshes at different resolutions.

B. Edge registration

Edge based representation is a compact way to describe the geometry of the objects present in a given range image. These representations are easy to obtain and they consist in extracting characteristic points where appear either a depth discontinuity (jump edge) or a surface orientation discontinuity (crease edge). Our edge extraction technique (see [10] for a complete description) [4] implements a scan line method: it consists in computing a binary edge map R, represented as a two dimensional array, where each element R(r,c) is a binary value indicating whether that point is an edge point or not.

Every row and column (hereinafter called scan lines) is approximated by a set of oriented quadratic functions. The algorithm requires two steps. First, the jump edge points are detected using a threshold adapted according to the local image resolution; these points are used to cut the original scan line into a set of sections (set of consecutive points) and to define the starting and ending points of each one of them. Second, a classical recursive splitting algorithm [2] approximates each section separately, by quadratic functions oriented along edge y, represented by the following equation: $y = ax^2 + bx + c \quad (1)$. The result of this recursive algorithm is a set of quadratic curves approximating the considered scan line's section. Once this section is approximated, the recursive algorithm is carried out over the next section of the given scan line. From each quadratic curve, the positions of the first and last points used to compute the parameters of function (1), are marked in the binary map R. Once a given scan line has been approximated, the algorithm starts again over the next new scan line, thus all the scan lines -rows and columns- are processed. Fig. 1(right) shows the edge representations for the *Dragon* object; the two representations presented on figure 1 are defined by a similar amount of points.

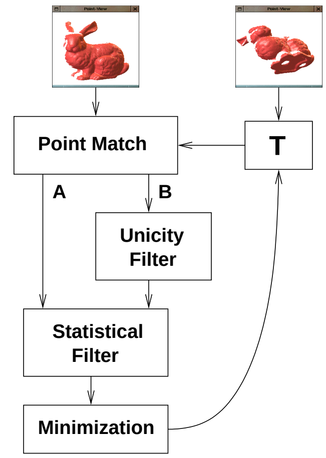

Fig. 2. ICP algorithm structure.

C. Our ICP implementation

In this section, the well known Iterative Closest Point, ICP, algorithm is summarized. Let us note C_i and C_j, two compact clouds of points, expressed in two different reference frames. The registration objective is to obtain the parameters of a matrix $\mathbf{T}$ which allows to express the points contained into C_j in the reference frame of C_i. ICP is an iterative process applied while the registration error is higher than a given threshold max_ε. As seen in figure 2, at iteration t, let us note $\mathbf{T}_{t-1}$ the current estimate of the $\mathbf{T}$ transform. With $\mathbf{T}_{t-1}$, the points from viewpoint j are transformed and each point $\vec{P}_j$ is matched with the closest point $\vec{P}_i$ from viewpoint i. Depending of the ICP error level, the matched points are filtered by using an unicity criterion (two points $\vec{P}_j$ cannot be matched with the same $\vec{P}_i$) (path B) or not (path A). Then, a statistical filtering technique [13] is used to discriminate the pairs by a limit distance. Afterwards, the remaining couples of matched points ($\vec{P}_i$, $\vec{P}_j$) are used to compute the registration error: $\varepsilon_t = \frac{1}{n} \cdot \sum_n ||\vec{P}_i . \vec{P}_j||$, where n is the number of matched points. If ε_t is below or equal to max_ε, the ICP process has converged: the matrix $\mathbf{T}_{t-1}$ is the final solution. On the contrary, if ε_t is higher than max_ε, couples ($\vec{P}_i$, $\vec{P}_j$) are used to compute a new set of parameters by minimizing:

$$\sum_n ||\vec{P}_i - (\Theta \vec{P}_j + \Gamma)||$$

With the parameters (Θ,Γ), a new transformation matrix $\mathbf{T}_t$ is computed, and the process starts again. This iterative

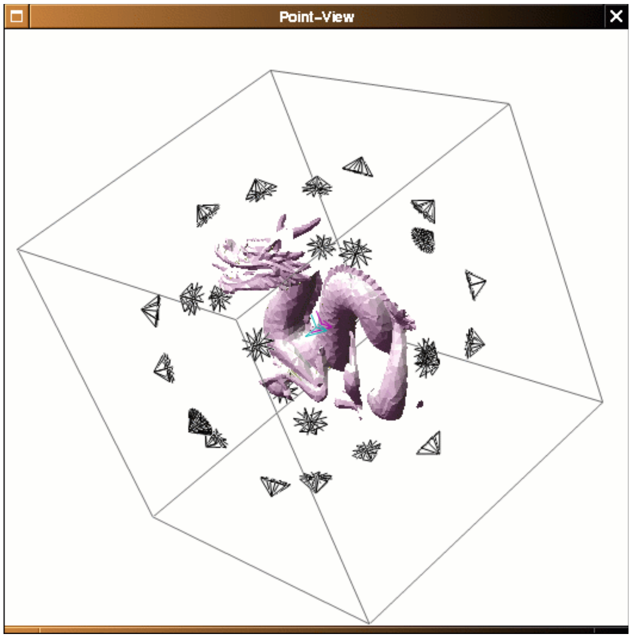

Fig. 3. Example 1: All initial positions (represented as a crosses) that have been tested.

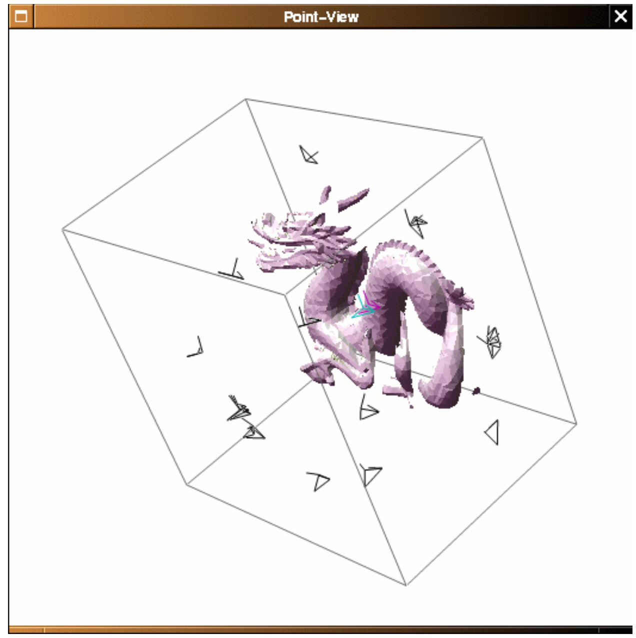

Fig. 4. Example 1: the positions that lead to a good convergence using edge-based images

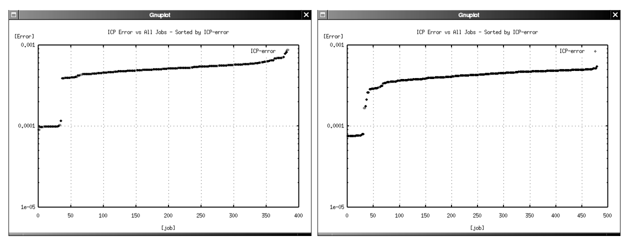

Fig. 5. Example 1: All ICP errors for edge-based (left) and mesh-based (right) tests. The flat line on the left of the graphics represent correct convergence in the corresponding tests.

process is executed until the convergence or until a maximum number of iterations max_t is reached.

Our implementation of the ICP algorithm does not use an explicit threshold max_ε, but a test on the gradient of the error; the convergence is reached at the iteration t if $\frac{\delta\varepsilon_t}{\varepsilon_t} = \frac{\varepsilon_t - \varepsilon_{t-1}}{\varepsilon_t} < 0.05$.

III. Robustness to the initial conditions

The ICP algorithm is based on the assumption that an estimated transformation $\mathbf{T}_0$ between two views is known beforehand. The question is how precise has to be that initial approximate transform in order that the method converges. In most of the cases, there is no information about that value and the solution becomes experimental.

Our objective is to evaluate the stability area for the $\mathbf{T}_0$ transform, it means, to estimate the maximal allowed difference between the initial and exact transforms $\mathbf{T}$ denoting a rotation Θ, and a translation Γ. From the description of the ICP method, the error on the initial translation Γ_0 has no effect on the convergence time; this error will be converted in an offset for every distance between the matched points, and consequently, on the ε_0 error. This translation error cannot cause by itself matching errors, and will be reduced during the first iterations. On the contrary, an important error on the initial rotation Θ_0 can allow a lot of wrong matchings, and finally, produce no convergence or a wrong solution for the ICP method.

Then, our objective has been at first to evaluate the convergence based only in the initial rotation parameters. A two stage algorithm, has been implemented. A list of N values (uniformly sampled around the 360 degree range) for every Euler angle is computed. Then, in a second stage, all the possible combinations from the previous values (Euler angles) are obtained, combined with a $(0,0,0)$ translation and used as starting point for our ICP methods (in total, N^3 initial transforms: typically, N equal to 10) edge-based registration and adaptive triangular mesh registration. All the initial positions are presented on a sphere around the *Dragon* object on figure 3.

After testing all the possible combinations the edge-based registration technique has succeeded in 15% of the cases while the adaptive triangular mesh registration has succeeded in 14% of the cases when the sculptured object was considered; on figures 4 and 8, "good" initial positions are presented. On the contrary, when a polygonal object was considered, the success in the edge based registration technique rise up to 20% and in the adaptive triangular mesh it was 19% (see the "good" initial positions on figure 6. Those cases in which neither the edge based

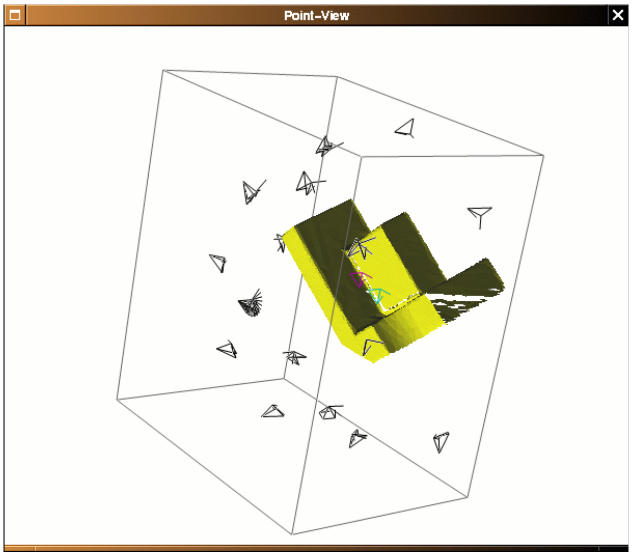

Fig. 6. Example 2: the positions that lead to a good convergence using mesh-based images

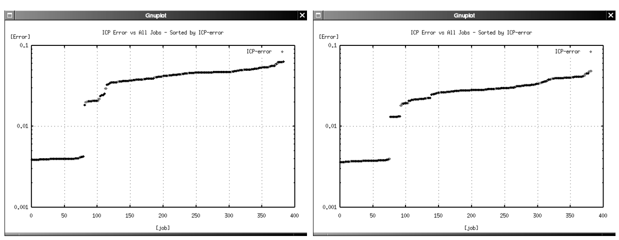

Fig. 7. Example 2: All ICP errors for edge-based (left) and mesh-based (right) tests. The flat line on the left of the graphics represent correct convergence in the corresponding tests.

representation nor the adaptive triangular mesh did not succeed in finding the correct registration parameter were considered as non-valid. An interesting property appears on the error diagrams on figures 5 and 7: The initial positions are sorted by increasing ICP errors; the gap between the "good" initial positions and the "bad" ones is always very clear ... for these objects, ambiguities are not possible. But, as it appears on the figures 4, 8 and 6, the "good" initial solutions are not connected as a bassin around the exact transform between the two images.

Considering the computation time required for this evaluation, one ICP run requires typically some seconds, depending on the number of points on the image to be registered. To reduce processing time for these evaluations (almost 1000 ICP runs by tested objects), the tests had been executed in 2, 4 and 8 CPUs in parallel.

IV. Conclusion

This paper presents another approach using the ICP algorithm in the sense of the search of the best initial position, and compares two types of image representation in the

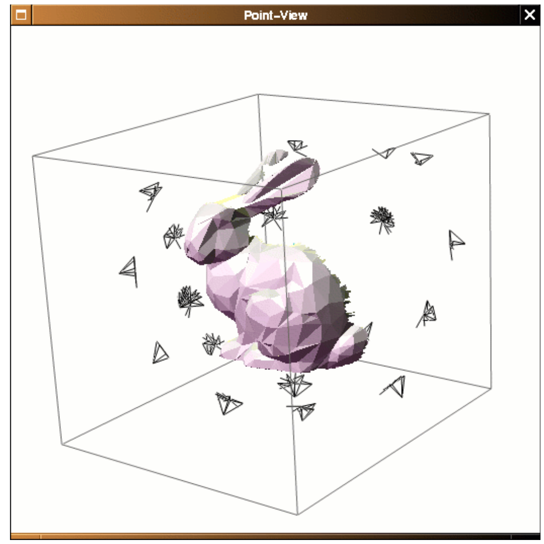

Fig. 8. Example 3: the positions that lead to a good convergence using mesh-based images

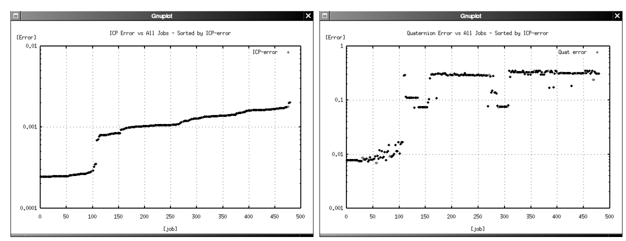

Fig. 9. Example 3: All ICP errors (left) and error-to-known-final-position (right) for mesh-based tests. The flat line on the left of the graphics represent correct convergence in the corresponding tests.

study; the edge-based and the mesh-based representations. As expected after [9], the results of this comparison gives very similar values for the two representations. Therefore the conclusions for this part are the same as [9]. Regarding the initial position search, in that an exhaustive scan of the Euler angles is made, the results gave aproximately a rate of 16% of success. To achieve more efficiency, this method can be modified to detect convergence in the first iterations, or in other words convergence tendence detection.

References

[1] http://sourceforge.net/projects/gts/. Web site.

[2] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley & sons, 1973.

[3] Martial Hebert and Owen Carmichael. Unconstrained registration of large 3d points sets for complex model building. In *IEEE International Conference on Intelligent Robots and Systems (IROS'98)*, pages p 360–367, October 1998.

[4] X. Jiang and H. Bunke. Edge detection in range images based on scan line approximation. *Computer Vision and Image Understanding*, 73(2):183–199,, February 1999.

[5] G. Medioni and Y. Chen. Object modeling by registration of multiple range images. In *IEEE Conference on Robotics and Automation, Sacremento, (USA)*, April 1991.

[6] MENSI. http://www.mensi.com/. Site web de la compagnie française mensi.

[7] F. Nashashibi and M. Devy. 3d incremental modeling and robot localization in a structured environment using a laser range finder. In *Proc. IEEE International Conference on Robotics and Automation, Atlanta (USA)*, May 1993.

[8] Ohio State University (MSU/WSU) Range Image Database. http://sampl.eng.ohio-state.edu/ sampl/data/3ddb/rid/index.htm. Web site.

[9] A. Restrepo Specht, A. Sappa, and M. Devy. Edge registration versus triangular mesh registration, a comparative study. In *submitted to 7th European Conference on Computer Vision, Copenhagen (Denmark)*, May 2002.

[10] A. Sappa and M. Devy. Fast range image segmentation by an edge detection strategy. In *Proc. Third International Conference on 3D Digital Imaging and Modeling, Quebec (Canada)*, May 2001.

[11] Stanford 3D scanning repository. http://graphics.stanford.edu/data/3dscanrep. Web site.

[12] ZF. http://www.zofre.de. Site web de la compagnie allemande zoeller&frohlich.

[13] Zhengyou Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):p 119–149, February 1994.