Available online at www.sciencedirect.com
SCIENCE DIRECT®
Signal Processing: *Image Communication* 20 (2005) 853–868
SIGNAL PROCESSING: IMAGE COMMUNICATION
www.elsevier.com/locate/image

Edge registration versus triangular mesh registration, a comparative study ☆

Andres Restrepo Specht[a], Angel D. Sappa[b,*], Michel Devy[a]

[a] *LAAS-CNRS, 7 avenue du Colonel Roche, 31077 Toulouse Cedex 04, France*
[b] *Computer Vision Center, Edifici O Campus UAB, 08193 Bellaterra, Barcelona, Spain*

Received 30 September 2004; accepted 6 May 2005

Abstract

This paper presents a comparative study of two techniques for registering range images. The first one consists in registering range images represented by means of compact adaptive triangular meshes. The second approach registers the same range images but represented by means of their edges. In both approaches the ICP algorithm is used to compute the registration parameters (rotation and translation). The main objective in both approaches is to register a compact representation instead of all original data points. The proposed comparative study is performed in an experimental way by using a set of real range images considering both structured and sculptured objects. Four different criteria are taken into account to perform the comparison: (1) robustness with respect to initial conditions (estimated relative positions between the range images to be registered), (2) robustness with respect to resolutions of set of points on which the ICP method is executed, (3) robustness with respect to overlapping between view fields, and (4) number of iterations versus registration error. By employing these criteria, triangular mesh based and edge based registrations are tested on several range images and results are compared with respect to the ground truth. Conclusions from these experimental results are presented.

Keywords: Registration; Edge extraction; 3D modeling

1. Introduction

Registering range images consists in finding the transformation matrix (defined by rotation and translation parameters) which will be used to transform one of these range images from its associated reference frame, to the reference frame associated with the other image. The registration can be understood as a step of an incremental

☆This work has been carried out as part of the CAMERA project (CAd Modeling of Built Environments from Range Analysis). CAMERA was an EC funded TMR network (ERB FMRX-CT97-0127). The second author was supported by The Ramón y Cajal Program.
*Corresponding author.
E-mail addresses: arestrep@laas.fr (A.R. Specht), angel.sappa@cvc.uab.es (A.D. Sappa), michel@laas.fr (M. Devy).

0923-5965/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2005.05.010

modeling process: the first image is a local image of a scene, expressed in the current sensor reference frame; the other one is a partial model which is being built, expressed in a global reference frame.

Generally, the parameters of the transformation matrix are found by using some minimization technique over a set of matched data points. Nowadays range sensors allow to obtain very large range images—millions of data points—in a few seconds [7]. But, as bigger is the range image, as much CPU time will be necessary to process it. The latter, converts the registration task in the bottleneck of any 3D modeling system. Bearing in mind this problem, an interesting point which should be considered is that not all points contained in a range image will provide the same useful information. Therefore, a preprocessing stage extracting interest points (e.g., points placed in high curvature regions, points defining edges) seems to be a good attempt to tackle the registration problem. Hence, further processing will be applied only over that reduced set of points.

Thus, the problem now is not only focused on the registration algorithm itself but on the strategy that allows to represent and handle range images in a compact way. Several researchers dealt with the same issue; their works have been done or are still in progress, in order to obtain compact representations of range images to speed up further processing (e.g. segmentation, recognition, object modeling) or in order to save space in storing or speed up transmission (e.g. compression of VRML files [24]).

Traditionally, adaptive triangular meshes are the "de facto" standard representation to build compact models from range images. Thus, several works have been proposed to register compact triangular meshes (e.g., [2,13,21]). Recently other works have been done to register range images represented by means of a set of features (e.g., [3,14,20]). In [14], a skeleton representation has been used to obtain a pre-alignment, but only scenes containing pipelines or cylindrical objects are considered. On the contrary, a more general approach has been proposed in [3]; an interest point is defined with a vector describing the local shape around it; only discriminant points are extracted from each range image and then, matched using a correlation measurement. In [20], we propose another method; it consists first, in extracting edge points from the original range images and then in registering these sets of points.

Although it is well known that the registration process can be applied over compact representations, improving considerably CPU time, to our knowledge there is not any work comparing the different registration strategies (see a first qualitative comparison on the WEB site of our project [17]). A comparison among the different techniques will help to choose the best one according to several criteria. In the current work only triangular mesh and edge based representations are considered. In both of them the same number of points is used to describe range images to be registered. The classical iterative closest point algorithm (ICP) is used in both approaches to compute the registration parameters.

Section 2 introduces the triangular mesh and edge based representations generated for the registration. Section 3 gives a short description of the registration method and especially of our implementation of the ICP algorithm. Section 4 presents the results of comparisons between registrations of both representations according to four criteria: (1) Robustness to initial conditions; (2) Robustness to different resolutions of the range images to be registered; (3) Robustness to register range images acquired with a different overlapping between the sensor's field of view and (4) CPU time versus registration error. Finally, conclusions and further works are given in Section 5.

2. Triangular mesh and edge based representations

This section describes the techniques used to compute adaptive triangular meshes and edge based representations. They will be used through this work to deal with the registration of range images.

2.1. Triangular mesh representation

During the last decade several algorithms have been proposed in order to generate adaptive

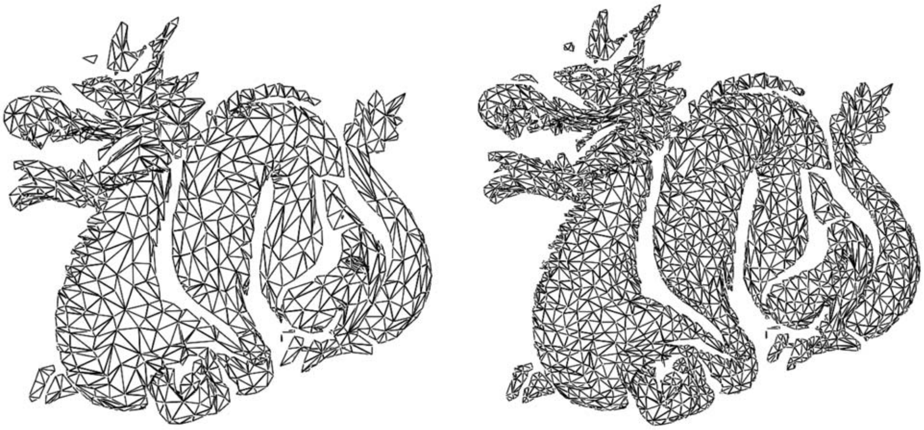

Fig. 1. A single view of a sculptured object represented by triangular meshes at different resolutions.

triangular meshes from range images. These approaches can be classified in two different groups: *fine-to-coarse* and *coarse-to-fine*. Fine-to-coarse algorithms (e.g., [6,12,22] to mention a few) start with a dense triangulation of all points of the given range image. Then, an iterative algorithm proceeds by either joining adjacent triangles or removing successive vertices; the decimation goes on either while the maximum approximation error between the current triangular mesh and the original range image is within a user defined tolerance, or while the number of preserved triangles or vertices, is higher than a user defined number. The proposed methods differ on the heuristics applied at each iteration to decide which triangles are joined or which points are removed, and on how the mesh is re-triangulated.

On the contrary, coarse-to-fine algorithms (pioneering work in [5], other examples in [9,19]) start with a coarse triangulation of a reduced set of points chosen from all points of the given range image. Then, an iterative algorithm proceeds by adding more points and updating the triangulation until the maximum approximation error between the current triangular mesh and the original range image is below or equal to a required tolerance.

In the current work a public surface refinement algorithm has been used (GTS open source [11]). This fine-to-coarse algorithm is applied over a dense triangular mesh built from 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. For every range image a set of triangular meshes at different resolutions is generated. The resolution of these triangular meshes is set by their corresponding edge based representations (Section 2.2). Hence, an initial dense triangular mesh is decimated until its number of vertices is the same than the corresponding edge based representation. Figs. 1 and 2 show examples of two objects represented by triangular meshes at different resolutions.

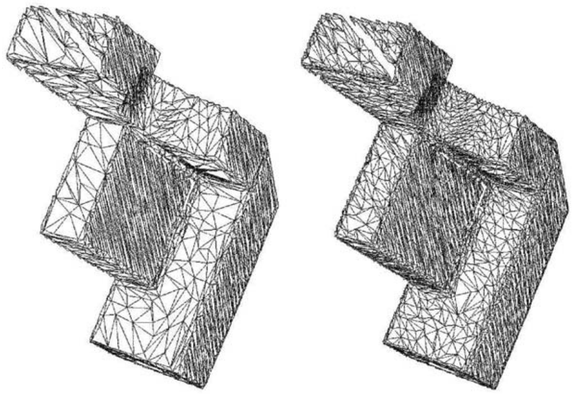

Fig. 2. A single view of a structured object represented by triangular meshes at different resolutions.

2.2. Edge based representation

An edge based representation is a compact way to describe the geometry of objects present in a given range image. These representations are easy to obtain and they consist in extracting characteristic points where either a depth discontinuity (jump edge) or a surface orientation discontinuity (crease edge) appear. Several approaches have been proposed in the literature in order to extract edge points from range images (e.g., [8,10,16]). Through this section a brief summary of the edge extraction technique is given (see [18] for a complete description). This technique works with range images provided by 3D sensors like stereovision or laser range finder—where range data are structured in an array. The algorithm consists in computing a binary edge map R, represented as a two dimensional array, where every element $R(r, c)$ is a binary value indicating whether that point is an edge point or not. In the current work a scan line processing algorithm has been used (only rows and columns were considered as scan lines).

Every row and column (hereinafter called scan lines) is approximated by a set of oriented quadratic functions. Quadratic functions have been selected due to the fact that they allow to generate a more generic edge based representation than if only straight lines were considered; moreover, quadratic approximations of edges allow to reduce the number of points to be considered during the registration stage. The algorithm consists of two steps. First, 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 [4] approximates each section separately, by means of oriented quadratic functions represented by the following equation:

$$y = ax^2 + bx + c. \tag{1}$$

Such a function is approximated by using only the first, middle and last points of the considered scan line's section. Let us note $(P_{f(x,y)}, P_{m(x,y)}, P_{l(x,y)})$, the first, middle and last points, respectively. Then, before obtaining the parameters of function (1), the set of points contained into the section, are rotated around the first point until the following configuration is reached: $P_{f(y)} = P_{l(y)} > P_{m(y)}$ (see illustration of Fig. 3(*middle*)).

The parameters of function (1) are then obtained analytically by using those three points. The approximation error between the obtained quadratic function and every rotated point is computed. If this error is greater than a given threshold $\max_{\text{error}}$, the set of points is split into two sections at that position where the biggest error appears. Then, the splitting algorithm is applied recursively to these two sections until all approximation errors are lower than $\max_{\text{error}}$.

The result of this recursive algorithm is a set of quadratic curves approximating the considered

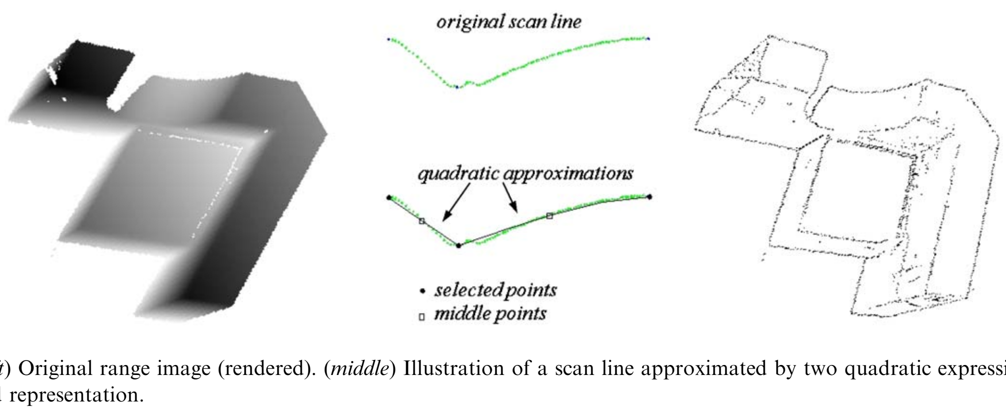

Fig. 3. (*left*) Original range image (rendered). (*middle*) Illustration of a scan line approximated by two quadratic expressions. (*right*) Edge based representation.

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 scan lines—rows and columns—are processed, the obtained binary edge map is the final result and the next stage is applied. Fig. 3(*right*) shows the binary edge map (edge based representation) obtained after processing the range image shown in Fig. 3(*left*). This image has been acquired on a polyhedral object; points close to object edges have been marked, but due to the sensor noise, some other points on faces have been detected also as discontinuity points. As the same as in the triangular mesh approach, several edge based representations with different resolutions, can be computed by setting different approximation errors $\max_{\text{error}}$. A high approximation error gives a representation with few points and on the contrary, a low approximation error gives a representation with many points. Fig. 4 shows the edge representations corresponding to the triangular meshes presented in Figs. 1 and 2; each representation is defined by a similar amount of points as the corresponding triangular mesh (i.e. Fig. 1(*left*), Fig. 2(*left*)).

3. Registration technique: the ICP algorithm

Range images represented as triangular meshes or by means of their edges are registered by using the well known ICP algorithm. Let us note C_i and C_j, two compact clouds of points, expressed in two different reference frames. These clouds of points could be either the vertices defining two triangular meshes or the points from two edge based representations. Even if it is possible to use attributes associated with these vertices or with these edge points, in both cases only the point positions are considered by our registration algorithm. Experimental results [3] have shown that ICP considering additional information to characterize the points converges in fewer iterations than ICP considering only the point positions. However, there is an extra overhead involved in working in a higher dimensional space ($3+n$ space, n being the dimension of the attribute space) thus on an average, the CPU time is equal or worse than ICP working only in a 3D space.

The registration objective is to obtain the parameters of a matrix T—denoting a rotation Θ, and a translation Γ—which allows us to express the points contained in C_j in the reference frame of C_i.

The parameters of the matrix T are computed by means of the following iterative process. This process is applied while the registration error is

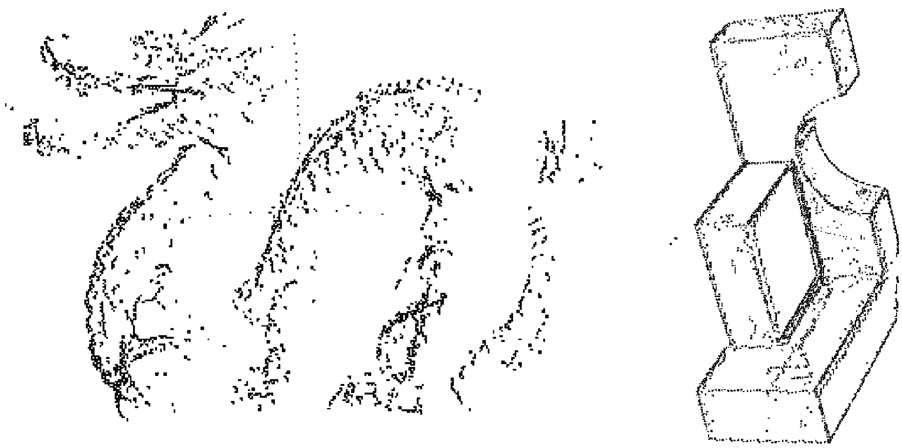

Fig. 4. Edge points corresponding to the images presented in Fig. 1(*left*) and 2(*right*).

higher than a given threshold $\max_\varepsilon$. First, an initial transformation matrix T_0—with parameters Θ_0 and Γ_0—is estimated. Each point P_j from C_j is transformed with this matrix, and then, matched with the closest point P_i from C_i, using a 3D structure in buckets in order to limit the complexity. The list of matched points are filtered by using two methods: a consistency test based on a unicity criterion (two points from C_j cannot be matched with the same P_i from C_i) and a statistical filtering technique [26]. This filtering method tries to obtain a consistent set of point matches, i.e. to keep only the paired points between the clouds that have consistent euclidean point-to-point distances. Consistent distances are defined as follows. At first, a statistical analysis of the point distances gives the mean distance and the standard deviation σ; then, matches with a distance higher than a given threshold (typically 3σ, then $2\sigma\ldots$) are removed, and the method is iterated until all matches are consistent.

Finally, the preserved couples of matched points (P_i, P_j) are used to compute the registration error:

$$\varepsilon_0 = \frac{1}{n}\cdot\sum_n \|\vec{P}_i - \vec{P}_j\|, \tag{2}$$

where n is the number of matched points. If ε_0 is below or equal to $\max_\varepsilon$, matrix T_0 is the final solution and parameters (Θ_0,Γ_0) are used to express the points P_j in the reference frame of the image C_i. On the contrary, if ε_0 is higher than $\max_\varepsilon$, couples (P_i, P_j) are used to compute a new set of parameters by minimizing the next expression [1]: $f = \sum_n (P_i - (\Theta_1 P_j + \Gamma_1))^2$.

With this new set of parameters (Θ_1,Γ_1), a new transformation matrix T_1 is computed, and the process starts again from the beginning (now considering the obtained matrix T_1). This iterative process is executed until the convergence—the error ε_t is lower than $\max_\varepsilon$—or the nonconvergence—a maximum number of iterations $\max_t$—is reached.

Our implementation of the ICP algorithm has the following characteristics:

- we do not use an explicit threshold $\max_\varepsilon$, but a test on the gradient of the error. The convergence is reached at the iteration t if $\delta\varepsilon_t/\varepsilon_t = (\varepsilon_t - \varepsilon_{t-1})/\varepsilon_t < 0.05$.
- during the first iterations only the statistical filter is applied assuming that the convergence is faster as the number of matched points increase.
- then, when a convergence is reached, the unicity criterion is verified in order to remove more outliers before applying the statistical filter. The unicity criterion is applied once $\delta\varepsilon_t/\varepsilon_t < 0.1$.

4. Comparison of the mesh and edge registrations

The registration process depends on several factors. Therefore, in order to carry out a comparison between mesh and edge based registrations we propose a study based on factors such as: type of surfaces perceived in the considered range images (e.g. sculptured surfaces, polygonal surfaces), initial conditions, resolution of the clouds of points to be registered, overlapping between views, to mention a few. They will be presented through this section.

The contribution of this paper is a systematic evaluation using some well known sets of range images containing polyhedral and sculptured surfaces. Sets of range images were obtained either from the Stanford 3D scanning repository [23], or from the OSU (MSU/WSU) data base [15]. Each range image contains a single object scanned from different view points. For some sequences, the sensor or the object displacements are known with a good accuracy, so that we can compare the ICP results with that exact positions, considered as the ground truth.

The results presented along this section are obtained by using image sequences acquired on the following objects:

- nonstructured objects: *Bunny* (12 images), *Dragon* (6 images, Fig. 1), *HappyBuddha* (6 images) and *Dough* (8 images).
- structured objects: *Block2* (5 images)—only polyhedral faces, *Curvblock* (5 images, Fig. 2)—a polyhedra with a cylindrical face,

Wye (2 images)—only cylindrical faces and *Valve* (8 images)—a complex object.

In this work we are not interested in an incremental fusion, where the registration is done between an image and the current object model provided by the fusion of the previous analyzed images: see [20] where such a fusion is illustrated using the *Bunny* sequence. In this paper, only image-to-image registrations are done to compare the ICP algorithm using either mesh vertices or edge points.

Below, the methodology proposed to do the comparison is introduced. Next, the evaluation of the proposed criteria is presented by using some range images to illustrate them. Finally, these criteria are evaluated over more than one hundred registration tests, considering eight different sets of range images; statistical results are presented at the end of this section.

4.1. Comparison methodology

As aforementioned, the comparison of the two approaches has been carried out taking into account four criteria. The first three robustness criteria are studied separately. The last criterion (CPU time versus registration error) consists in studying the registration robustness when two of the previous criteria are considered together (i.e., range images at different resolutions with different initial positions).

Hereinafter every time that the paper mentions a registration test, it refers to the two registration approaches (by using triangular meshes and by using edge based representations) both of them carried out over the same conditions—resolution and relative position between the representations to be registered. A data base defined by a set of couples (two triangular meshes at different resolutions; and two edge based representations at the same resolution than the corresponding triangular mesh) was generated to evaluate the comparison criteria.

The evaluation is based on several measurements: (1) the convergence, and if ICP converges, the number of iterations and the convergence regularity (how monotonic is the decrease of the ICP error), (2) the final ICP error (mean distances between the matched points), (3) the final 3D error with respect to the ground truth when this measurement is available, and finally, (4) the CPU time for the global procedure, from the control point extraction to the final results.

4.2. Evaluation of the criteria

4.2.1. Robustness to initial conditions (rotation and translation)

Range image registration can be studied as an optimization problem. In other words, based on the assumption that an estimated transformation between two views is known beforehand, a cost function could be defined. This cost function will measure the quality of the alignment between the partially overlapped surfaces contained into each view. Hence, by minimizing this cost function the range images will be registered. However, as in every optimization problem, the success in reaching the minimum value is highly dependent on the initial conditions (i.e., how precise is the initial transformation matrix T_0, Section 3). In most of the cases there is no kind of information about that value and the solution becomes experimental.

After testing all examples with a user selected initial condition, the edge based registration technique has succeeded in 76% of the cases while the adaptive triangular mesh registration has succeeded in 92% of the cases when sculptured objects were considered. On the contrary, when structured objects were considered, the success in the edge based registration technique rises up to 95% and in the adaptive triangular mesh falls up to 85%. Those cases in which neither the edge based representation nor the adaptive triangular mesh did not succeed in finding the correct registration parameter, were considered as non-valid and they were not considered in the statistics. As it was expected, for this comparison criterion the edge based registration gives better results for the structured or polyhedral objects.

4.2.2. Robustness to register range images represented at different resolutions

This comparison has been performed by registering triangular meshes which were decimated at different values [25] and edge based representations obtained by considering different approximation errors. The initial conditions in both cases were the same and they were selected from the previous comparison (initial conditions in which both techniques converge). As the same as in the previous comparison, when at least one of the used techniques succeeds in finding the registration parameters, that representation's resolution is considered as a valid one and it is taken into account for the statistical result.

The representations to be registered are obtained in the following way. First edge based representations at different resolutions (approximation error: see Section 2.2) were computed. Then the corresponding triangular meshes are obtained by decimating the original dense range image according to the amount of points contained into the edge based representations.

At first, the ICP method using either the mesh vertices or the edge points, has been applied to representations generated on the *Dragon* object (see in Fig. 5 the higher and the lower resolutions for the mesh representation). When the resolutions of the two images to be registered are almost the same, all results are summarized in Table 1. For the different resolutions on the *Dragon* object, the ICP method using the mesh vertices always converges faster, and succeeds in matching more points. It is the reason why the ICP error is lower than for the ICP method using the edge points. Even at convergence, the ICP error for the edge registration, is generally higher.

This result will be confirmed by our final statistical analysis: the mesh representation is better for the nonstructured objects. Nevertheless, a good point for the two methods, is that ICP can succeed in registering clouds of points with low resolution (about 350 points in our test), and if needed by the application, such a coarse registration could provide a good initial condition for a fine registration using a higher resolution. Following this strategy, we have evaluated several range images and noticed that the CPU time needed to register two range images using only the higher resolution, is approx. 500% larger (!!!) than the CPU time of a two-steps procedure: at first, a coarse registration using a lower resolution, and then a refinement using the higher one.

Different from Table 1, where range images of similar resolution are registered, the registration of range images represented at different resolutions is compared with registration of the same resolutions. Fig 6(top) presents an illustration when range images at different resolutions were considered. In this case another sculptured object has been evaluated (*HappyBuddha*). Fig. 6(*bottom*) presents the evolution of ICP's error along the iterations, when the ICP is applied over both edge and mesh based representations. In both cases high resolutions (H-H)—almost 5000 points, low resolutions (L-L)—almost 2000 points, and

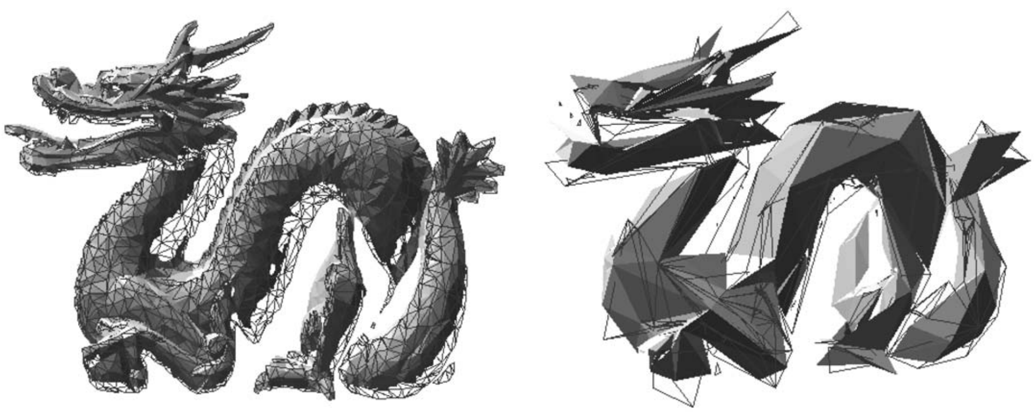

Fig. 5. Evaluation of the ICP methods according to the image resolution: (*left*) registration using the mesh vertices with a high resolution; (*right*) a very low resolution. The same evaluation, using the edge points, with the same number of points, has been performed.

Table 1
Comparison of the two methods when range images of similar resolution are registered

Image1 resol.	3365	3265	1835	1774	720	695	509	331
Image2 resol.	3851	3723	1903	1836	894	857	668	427
Edge registration								
Nb. iter. at conv.	50	60	47	56	66	58	100	39
% matched pts	37%	41%	38%	27%	39%	38%	22%	30%
ICP err. at mesh convergence	0.00010	0.00010	0.00036	0.00025	0.00012	0.00031	0.00091	0.00077
ICP err. at conv.	0.000046	0.00010	0.00033	0.000044	0.00010	0.00031	0.00047	0.00078
Mesh registration								
Nb. iter. at conv.	31	53	27	19	55	49	35	35
% matched pts	36%	60%	56%	32%	60%	56%	41%	37%
ICP err. at conv.	0.00017	0.000073	0.00019	0.00019	0.000070	0.00018	0.00055	0.00071
Comparison								
Nb. iterations	mesh	mesh	mesh	mesh	mesh	mesh	mesh	mesh
Accuracy	edge	mesh	mesh	mesh	mesh	mesh	edge	mesh

ICP error at mesh convergence shows the ICP error of the edge registration at the final iteration of the corresponding mesh registration, which has ended before in all cases.

different resolutions (H-L) were used. Both methods showed a good performance under this evaluation. In principle, they would have to converge faster when one of the representations to be registered has less points; in fact, for the mesh registration, the three curves are very close and the iteration numbers are equal; for the edge registration, the convergence is faster with high resolution, because more matchings are found. At the convergence, the ICP's errors are almost equivalent, even if the edge registration is slightly better for this object with a complex relief.

4.2.3. Robustness with respect to the overlapping between the views

The evaluation of the robustness with respect to the overlapping rate between the two views to be registered has been done using two objects: the *Dough* object and the *Valve* object. In Fig. 7, the image superpositions are shown, using a textured representation for one view, and a mesh for the other. In both examples, a large number of range images, from different points of views, were considered.

For the two objects, we have tried to register a reference view acquired for an angle of 0°, with other views acquired at every 10° from that reference position (10°, 20°, 30°,... until the nonconvergence is reached due to the lack of overlap—this event occurs for the two representations at: 80° for the *Valve* sequence and 100° for the *Dough* sequence). Table 2 summarizes all results for the *Valve* example, in this case as a summary we can say that the edge representation is better taken into account the accuracy point of view, even if the mesh registration converges faster and finds more matchings.

4.2.4. Number of iterations versus registration error

In this section, two of the previous criteria were considered together, representations at different resolutions with different initial conditions. Then, by using the values computed in the previous sections, an algorithm selects in a random way a combination of initial conditions and resolution for the representations which will be registered (these values are considered twice, first for the triangular mesh registration and second for the edge based registration).

The graphics shown in Fig. 8, represent the evolution of the 3D error along the ICP iterations: (*top*) registration of two *Dragon* images with two resolutions using either edge points or mesh vertices, (*bottom*) the same for two *Curvblock* images. The representations to be registered contain the same amount of points and their

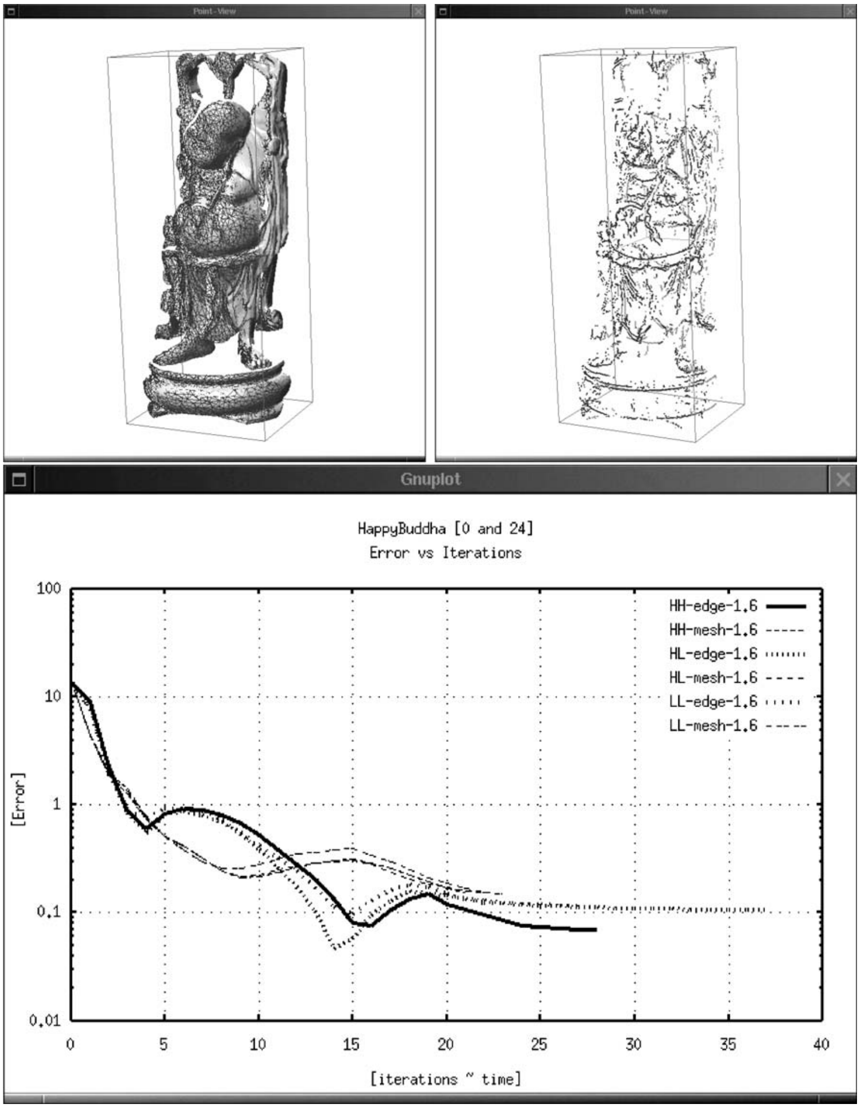

Fig. 6. Evaluation of the ICP methods with different resolutions: (*top left*) mesh registrations, (*top right*) edge registrations, (*bottom*) ICP errors decreases.

original positions have been the same in all examples. The error evolution is more regular for the *Curvblock* images; some local minimums appear (more often for the *Curvblock* images), when the unicity criterion is tested in order to discard more outliers.

In Fig. 9(*bottom*), the same graphic is shown for the registrations of one front view of the *Bunny* object with one top view and with a back view. In these examples, the edge based registration technique is better due to the fact that the points over the edges represent better the shape of the surfaces contained in the given range images. For the registrations of the front and back views, the error evolution, for both methods, is slower due to the weaker overlapping rate between the views. Notice as the final 3D error for the mesh registration of the front and top views is very high.

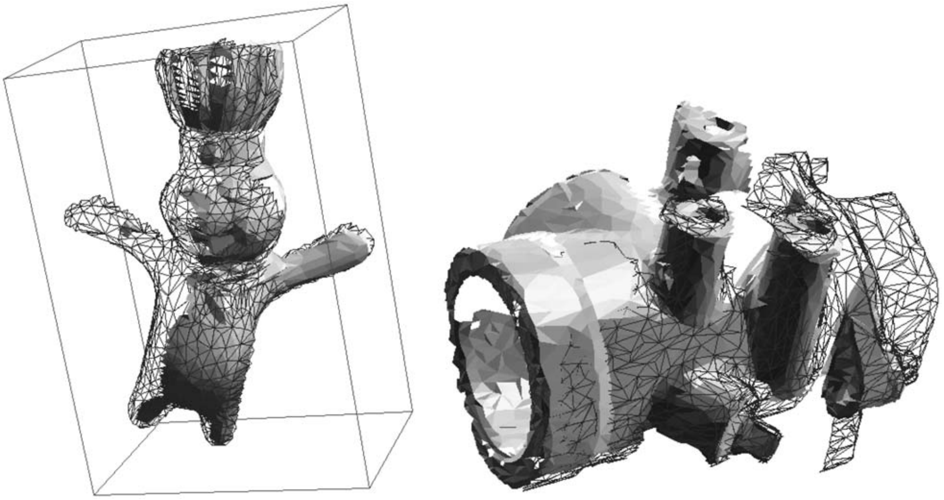

Fig. 7. Evaluation of the ICP methods according to the overlapping between the images (*left*) a sculptured object: *Dough* (40° difference between the two viewpoints); (*right*) a structured object: *Valve* (60° difference between the two viewpoints).

Table 2
Comparison of the two methods with respect to the overlapping rate

Registered images	0–10	0–20	0–30	0–40	0–50	0–60	0–70
Edge registration							
Nb. iter. at conv.	31	42	12	84	29	52	150
Nb. matched points	906	874	801	518	478	433	256
% matched points	67%	65%	59%	39%	35%	32%	19%
ICP err. at mesh conv.	0.0467	0.0456	—	0.1792	—	0.1645	—
ICP err. at conv.	0.0465	0.0463	0.0655	0.1074	0.1534	0.1613	0.4755
Mesh registration							
Nb. iter. at conv.	23	18	33	66	149	43	150
Nb. matched points	954	915	848	722	537	498	314
% matched points	70%	68%	63%	53%	40%	37%	23%
ICP err. at edge conv.	—	—	0.0688	—	0.1795	—	—
ICP err. at conv.	0.0488	0.0552	0.0690	0.1034	0.1808	0.1910	0.5295
Comparison							
Nb. iterations	mesh	mesh	edge	mesh	edge!!	mesh	=
Accuracy	edge	edge	edge	mesh	edge	edge	edge

ICP err. at edge conv. and *ICP err. at mesh conv.* show, if applicable, the ICP error of each registration at the final iteration of the corresponding opposite registration.

4.3. Global evaluation

Finally, at this stage the CPU time versus the registration error, when the previous criteria are considered together (e.g., different resolutions, initial conditions), is studied over both representations: meshes and edges.

In addition to the examples presented in the previous sections, other sets of range images were considered (4 sets of sculptured objects and 4 sets

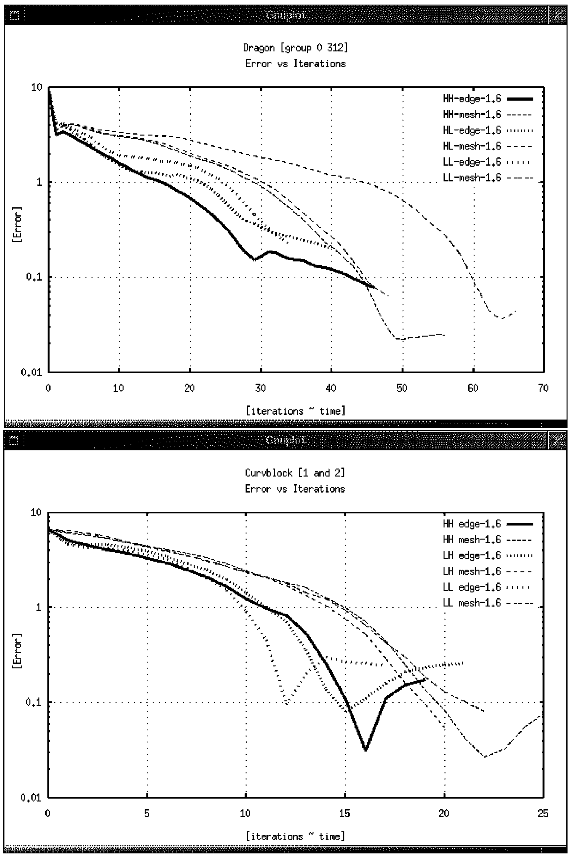

Fig. 8. Reference frame error evolution along the iterations: (*top*) a sculptured object (*Dragon*); (*bottom*) a structured object (*Curvblock*).

of polygonal objects). More than one hundred ICP results have been compared, using images from different objects, with different resolutions or different initial conditions. The global statistical results for each set of range images are shown in Table 3.

We could conclude that the ICP convergence is easier with edge points than mesh vertices when structured objects are considered (e.g., *Block2*). Otherwise, mesh vertices are the best choice if sculptured objects need to be registered (e.g., *HappyBuddha*). The same conclusion is also valid for the final 3D error (comparison with the ground truth, when it is available). In other words, the final result is closer to the ground truth with edge points than mesh vertices when structured objects are considered. On the contrary, mesh vertices give the best result for sculptured objects.

The aforementioned conclusion can be easily explained by the following statement. Low convergence ratio, when structured objects are registered by using mesh vertices, is originated by vertices placed inside the objects' faces. These vertices, left by the decimation algorithm, are not located at the same position in the different views; so the matchings with these points are either discarded by the statistical filter (and, for some images, ICP cannot converge, or converge with more iterations), or are preserved, but make less accurate the final ICP result.

On the contrary, for nonstructured shapes like sculptured objects, the vertices are located on curvature discontinuities, and their positions are stable enough between different views, so that ICP converges faster with mesh vertices than with edge points.

Finally, regarding the CPU time comparisons, we have noticed that, as an average, edge registration (including edge point extraction) is almost one hundred times faster than mesh registration (including mesh generation). We conclude that it is the most important advantage for the edge registration, whatever the nature of the scenes to be registered and whatever the image configurations.

Another advantage of the edge based representations, comes from the labeling of the edge points. Without using like in [3], an attribute vector to describe the local shape around the edge point, the ICP result can be improved using the nature of the discontinuity: depth (jump edge), orientation (crease) or visibility (image boundaries or occluding lines):

- according to our experiments, for nonpolyhedral objects, it is better to remove jump edges before the execution of an edge registration. For instance, Fig. 10(*left*) shows two images (*Wye* object) registered only using crease edges. On the contrary, Fig. 10(*right*) shows the result of the same registration, when all edges have been considered. In this example (we can generalize for all scenes containing nonpolyhedral objects like cylindrical surfaces for example) the convergence is faster and more accurate only using crease edges. It is because jump edges

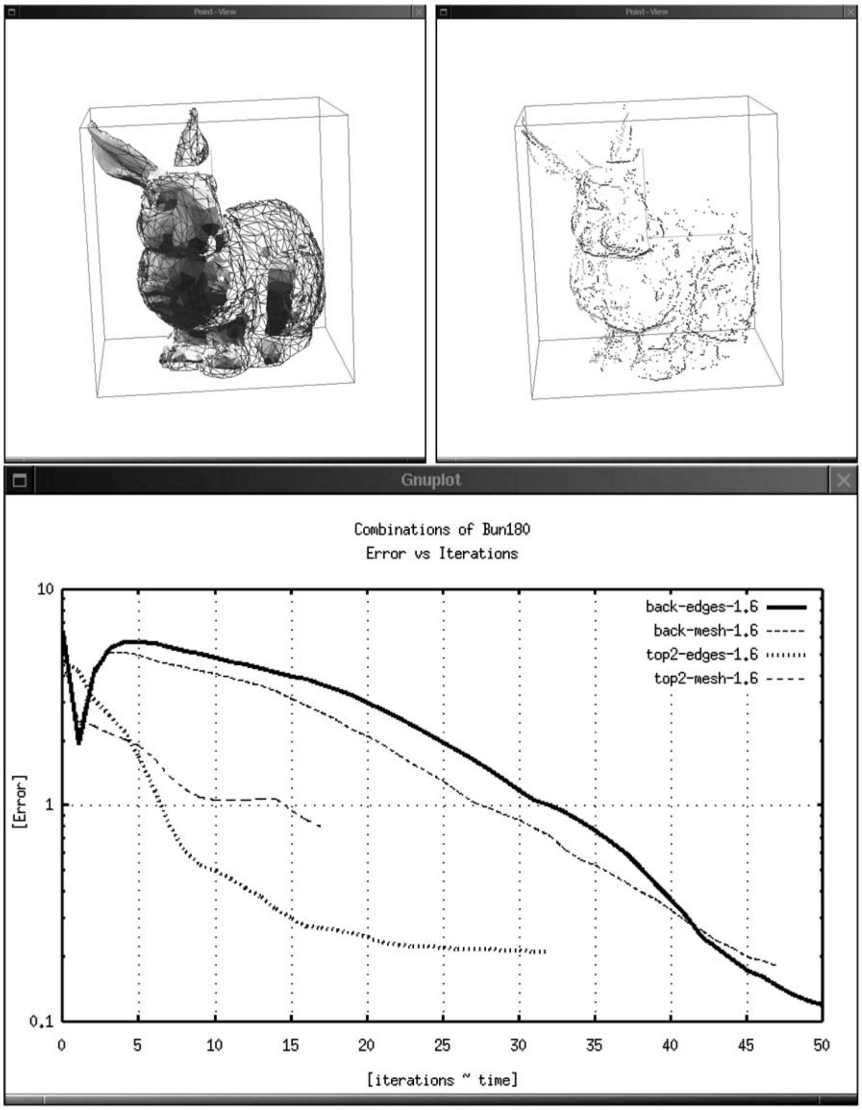

Fig. 9. Evaluation of the ICP methods on several *Bunny* images: (*top left*) mesh registrations, (*top right*) edge registrations, (*bottom*) ICP errors decreases.

on a cylindrical object belong to the surface boundaries defined only from that point of view, but they are not really jump edges (discontinuity on the surfaces).

- such a filtering is not possible for the mesh vertices, even if the mesh registration succeeds also for these two images (Fig. 11(*left*)).
- for polyhedral objects, jump edges are kept for the registration, because they could match with crease edges in other views. Fig. 11(*right*) shows *Block2* images registered by using all edge points, even the points that correspond to the image boundaries.

Table 3
Global results on all objects

Object	*Curvblock*	*Block2*	*Valve*	*Wye*	*Bunny*	*Happy*	*Dragon*	*Dough*
Number of tests	23	10	7	2	8	20	28	7
Number of resolutions	2	2	1	1	1	2	8	1
Nonconvergence with mesh vertices	1	4	1	0	0	1	3	1
Nonconvergence with edge points	1	0	1	0	0	7	7	1
Mean errors for edge points	0.2104	0.1738	na	na	0.3809	0.4721	0.5991	na
Mean errors for mesh points	0.4336	2.7862	na	na	0.3657	0.4136	0.3793	na
CPU time for edge registration including the edge extraction [seg]	0.559	0.536	0.573	0.513	1.145	2.634	1.37	0.500
CPU time for mesh registration including the mesh generation [seg]	27.75	39.87	30.16	58.99	101.31	181.32	90.06	21.07

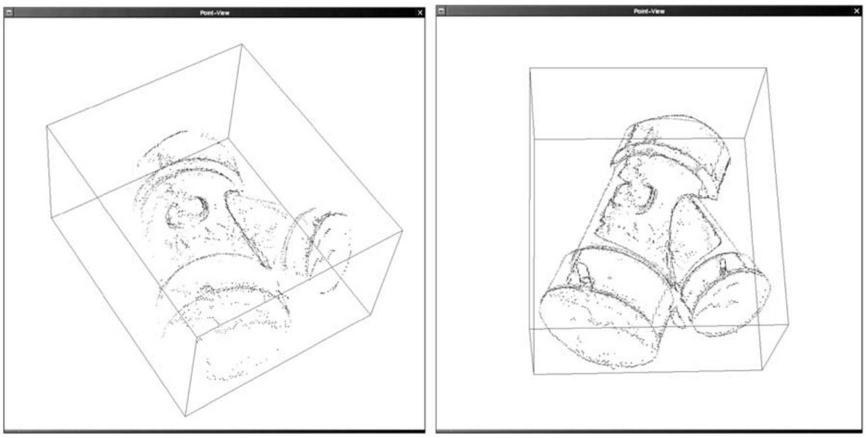

Fig. 10. Evaluation of the ICP methods on the *Wye* cylindrical object: (*left*) by using only the crease edges; (*right*) by using jump and crease edges.

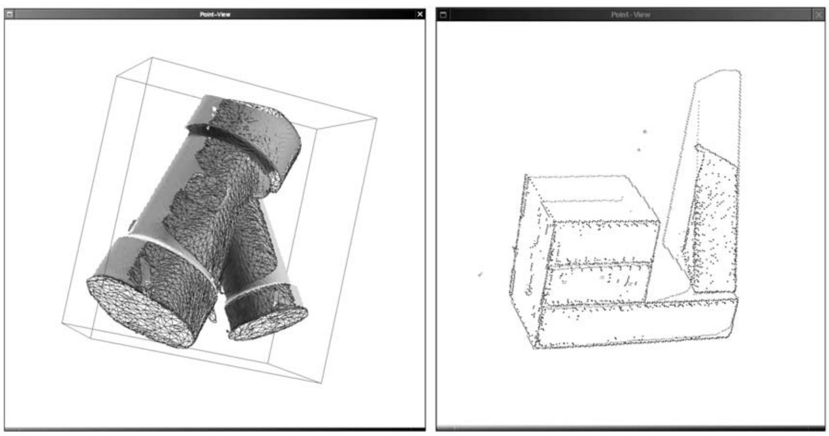

Fig. 11. Evaluation of the ICP methods: (*left*) two *Wye* images registered using mesh vertices; (*right*) two *Block2* images registered using edge points.

5. Conclusion

This paper presents an experimental comparison of two registration strategies. After testing several sequences of range images acquired from different type of objects, it is difficult to say which one of them is the best; it depends on the shapes contained in the given range images. However, we conclude the following: (a) edge based registration some times is less accurate than triangular mesh registration for the nonstructured scenes, but considering all tests, it is slightly better (less iterations, more monotonic decrease of the error); (b) the robustness to register representations at different resolutions is the same in both cases; (c) the most important point that deserve to be highlighted, is that the edge based registration is considerably faster than triangular mesh registration (in average we can say 100 times faster), counting up from the generation of each representation.

A future work includes to merge both approaches, thus in a first stage an edge registration approach will be applied, then as soon as the representations are near to the final position (it could be detected by studying the error gradient) the edge registration algorithm will be switched to another strategy which considers more points (vertices of an adaptive triangular mesh or, if it is possible, all data points contained in the given range images).

References

[1] K. Arun, T. Huang, S. Blostein, Least-squares fitting of two 3-d points sets, IEEE Trans. Pattern Anal. Mach. Intell. 9 (5) (September 1987) 698–700.
[2] J. Bozier, M. Devy, A. Sappa, A geometrical approach for the incremental modelling of free form surface by triangular meshes, in: Proceedings of the International Symposium on Intelligent Robotic Systems (SIRS'2000), Reading (UK), July 2000.
[3] O. Carmichael, M. Hebert, Unconstrained registration of large 3d points sets for complex model building, in: Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS'98), October 1998, pp. 360–367.
[4] R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973.
[5] L. De Floriani, A pyramidal data structure for triangle-based surface description, IEEE Comput. Graph. Appl. 9 (2) (March 1989) 67–78.
[6] M. Garland, P. Heckber, Surface simplification using quadric error metrics, in: ACM SIGGRAPH 97 Proceedings, Los Angeles, USA, August 1997.
[7] J. Hancock and al, Active laser radar for high-performance measurement, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'98), Leuven, Belgium, May 1998.
[8] A. Hoover, D. Goldgof, K. Bowyer, Extracting a valid boundary representation from a segmented range image, IEEE Trans. Pattern Anal. Mach. Intell. 17 (9) (September 1995) 920–924.
[9] H. Hoppe, View-dependent refinement of progressive meshes, in: The Proceedings of the Computer Graphics, June 1997.
[10] X. Jiang, H. Bunke, Edge detection in range images based on scan line approximation, Comput. Vision Image Understanding 73 (2) (February 1999) 183–199.
[11] The GNU Triangulated Surface Library, http://sourceforge.net/projects/gts/, Web site.
[12] P. Lindstrom, Out-of-core simplification of large polygonal models, in: ACM SIGGRAPH 2000 Proceedings, New Orleans, USA, July 2000, pp. 259–262.
[13] G. Medioni, Y. Chen, Object modeling by registration of multiple range images, in: Proceedings of the IEEE Conference on Robotics and Automation (ICRA'91), Sacremento (USA), April 1991.
[14] V. Murino, L. Ronchetti, U. Castellani, A. Fusiello, Reconstruction of complex environments by robust prealigned ic, in: Proceedings of the IEEE Third International Conference on 3-D Digital Imaging and Modeling (3DIM'2001), Quebec City, Canada, May 2001.
[15] Ohio State University (MSU/WSU) Range Image Database, http://sampl.eng.ohio-state.edu/sampl/data/3ddb/rid/index.htm, Web site.
[16] O. Pereira Bellon, A. Direne, L. Silva, Edge detection to guide range image segmentation by clustering techniques, in: Proceedings of the IEEE International Conference on Image Processing (ICIP'99), Kobe, Japan, October 1999.
[17] The CAMERA project, http://homepages.inf.ed.ac.uk/rbf/camera/camera.htm, Web site.
[18] A. Sappa, M. Devy, Fast range image segmentation by an edge detection strategy, in: Proceedings of the Third International Conference on 3D Digital Imaging and Modeling (3DIM'2001), Quebec (Canada), May 2001.
[19] A. Sappa, M.A. García, Modeling range images with bounded error triangular meshes without optimization, in: Proceedings of the 15th IAPR International Conference on Pattern Recognition (ICPR'2000), Barcelona, Spain, September 2000.
[20] A. Sappa, A. Restrepo Specht, M. Devy, Range image registration by using an edge-based representation, in: Proceedings of the Ninth International Symposium on Intelligent Robotic Systems (SIRS'2001), Toulouse (France), July 2001.

[21] V. Sequeira, K. Ng, E. Wolfart, J.G.M. Goncalves, D. Hogg, Automated reconstruction of 3d models from real environments, ISPRS Journal of Photogrammetry and Remote Sensing (1999) 1–22.

[22] E. Shaffer, M. Garland, Efficient adaptive simplification of massive meshes, in: IEEE Visualization, June 2001.

[23] Stanford, 3D Scanning Repository, http://graphics.stanford.edu/data/3dscanrep, Web site.

[24] G. Taubin, J. Rossignac, Geometric compression through topological surgery, Technical Report, January 1996.

[25] G. Turk, M. Levoy, Zippered polygon meshes from range images, in: ACM SIGGRAPH 94 Proceedings, Orlando, 1994, pp. 311–318.

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