ELSEVIER

Image and Vision Computing 15 (1997) 925–935

image AND vision COMPUTING

Efficient registration of NURBS geometry

Mihailo Ristic*, Djordje Brujic

Imperial College of Science, Technology and Medicine, Mechanical Engineering Department, Exhibition Road, London SW7 2BX, UK

Received 17 June 1996; revised 27 May 1997; accepted 29 May 1997

Abstract

The paper presents an implementation of free-form surface registration in relation to inspection of engineering components, defined as NURBS. Registration is principally performed through the Iterative Closest Point (ICP) method. The time-critical step in ICP was found to be the determination of the closest points on NURBS to a given point in space. Significant speed improvements were achieved through the adoption of a dual surface representation, involving approximation of NURBS entities by a polyhedral mesh. A criterion for sufficient polyhedral approximation was derived and implemented, producing encouraging results. Original solutions are suggested in order to further improve the computational speed. Extensive testing has been carried out, showing that the proposed registration method handles a full six degrees of freedom and achieves global convergence. Performance of the implemented algorithms is discussed with reference to registration of a turbine blade airfoil. © 1997 Elsevier Science B.V.

Keywords: registration; inspection; free-form surfaces; NURBS; best-fitting

1. Introduction

Accurate registration of free-form surfaces is an essential requirement for dimensional inspection and, as such, it is relevant to many branches of manufacturing industry. The requirement arises at several stages of the product life cycle, such as during product and manufacturing process development, production, and also in the repair of broken or worn-out parts [1]. A common feature of components containing free-form surfaces is the absence of clearly defined reference features. The prime examples are the parts produced using forming processes (casting, forging, pressing), such as aero engine compressor and turbine blades, car body panels and others. Typically, registration relies on employing special-purpose inspection jigs which provide the registration features that are not available on the part itself.

With the advent of various non-contact sensors, it is now possible to rapidly collect a large number of point measurements so that the actual surface may be characterised in full. A review of the most relevant non-contact measurement techniques is provided by Newman and Jain [2], while the accepted view [3] is that range sensing, such as laser triangulation, is the most appropriate choice for accurate dimensional inspection. In such situations software best-fitting is the essential registration tool to be employed in order to enable accurate measurement and to eliminate the systematic component of the error due to misalignment.

The main difficulty with best-fitting has been in dealing with large data sets in relation to parametric surfaces such as NURBS (Non-Uniform Rational B-Spline). The most suitable registration technique in such situations is widely accepted to be the Iterative Closest Point (ICP) method [4]. It is based on the idea that, at each iteration, the points on the nominal model that are nearest to the measured ones may be taken as corresponding and the aligning transformation is calculated by least-squares minimisation of the collective point-to-point distances. The process is then repeated until satisfactory convergence is achieved.

Although the ICP method is well known, several fundamental difficulties have not been adequately resolved to date. It was found that the time required to perform the best-fitting task by the published algorithms is so excessive that, in practical applications, they can handle only a limited number of measurements. The computational burden is largely associated with finding the nearest point on NURBS, which demands the use of numerical methods because the equations have no known analytical solution. Furthermore, the initial guess in finding the closest point was found to be of critical importance, in order to guarantee a correct result of this computation. However, none of the published methods were found to address this issue

* Tel: 0171 594 7048; fax: 0171 584 7239; e-mail: m.ristic, d.brujic@ic.ac.uk

PII S0262-8856(97)00049-8

adequately. As a consequence, in many situations, the accuracy and robustness of the ICP method as a whole were observed to be limited.

This paper presents an implementation of ICP registration for free-form surfaces that aims to overcome these problems. The main objective was to achieve high computational efficiency, without sacrificing the accuracy, when dealing with large measurement data sets and nominal geometry defined as NURBS. An objective was also to realise a robust method that could be readily and reliably applied to a variety of shapes.

The paper is organised in the following way. The next section deals with techniques for geometric modelling as the basis of the proposed registration method. It presents the most important functions related to NURBS and the approximation criteria that must be satisfied for registration purposes. The subsequent section presents the implementation of the ICP registration algorithm, showing how a number of important improvements in the computational speed were achieved. This is followed by a discussion of the achieved registration performance with reference to actual and simulated measurement of a turbine blade airfoil. The final section summarises the main conclusions drawn from this work.

2. Geometric modelling for inspection

The basis for ICP registration is the calculation of the closest point on the surface to a given point in space. Calculation of the nearest point on a parametric surface demands the use of iterative search techniques, since the equations in general have no closed form solutions. The initial guess for this search may be provided by the vertices of a suitable polygonal approximation of the parametric surface model.

In this work, NURBS surfaces were chosen to be the main modelling entity for the nominal geometry because they offer good local shape control and are rapidly becoming the principal building blocks of modern CAD/CAM systems. Importantly, NURBS are supported by IGES (entity 128), and almost all surface geometry exchange in heterogeneous environments is done via surface entities in IGES files. Since NURBS can exactly represent the natural quadric surfaces, software complexity can be drastically reduced if all surfaces are converted to NURBS. However, the overhead created by the unequal weighting is significant and NURBS formulation can incur considerable performance and reliability penalties. This issue was central to the design of the geometric modeller.

Following this, a number of geometric functions have been implemented and these are described below.

2.1. Computing a point on NURBS

The function which calculates x, y, z coordinates of a point on the surface defined by parameters u, v is implemented as suggested by Bartels et al. [5]. The algorithm was found to be very efficient. For example, on a SUN-SPARC2 workstation it computes approximately 5000 point coordinates per second for a fourth-order NURBS comprising 13 $\times$ 89 knots. (Source code for the benchmark is available via anonymous ftp on robot-gw.me.ic.ac.uk in pub/inspection/nurbs.zip)

2.2. Polyhedral approximation of NURBS surfaces

Polyhedral approximation is produced by subdivision sampling of the parametric surface. In its simplest form it generates a parametrically uniform grid of $m \times n$ points ordered as 'sections' in space. However, our implementation involved adaptive subdivision sampling in order to avoid over-sampling of the regions with low curvature, thus reducing the number of triangles needed to accurately describe the surface. The surface is then approximated by a mesh of straight line segments joining each pair of adjacent grid points. Triangular polyhedra are obtained by splitting each rectangle into two triangles. Topological relations encoding the connectivity structure are stored in corresponding linked lists.

The method is essentially that presented by Von Herzen and Barr [6], in which the sampling is performed recursively in the following steps until certain subdivision termination criteria have been satisfied:

1. The surface is sampled on a uniform parametric grid at some initial resolution.
2. Each region is evaluated using several acceptance criteria.
3. If the region is not acceptable, then it is split into two or four smaller regions, since the criteria may not be fulfilled with respect to one or both dimensions.
4. Steps 2 and 3 are repeated until the entire surface is sampled adequately.
5. The regions are broken into triangles.

The acceptance criteria for each region in adaptive sampling is a problem that requires special attention. Clearly, the first requirement is to guarantee that the triangulation is accurate to within a given tolerance. Consequently, we have implemented an algorithm in which one of the subdivision termination criteria is based on flatness [7], where the quality measure is the distance from the surface to its planar approximation. Most of the published literature assumes that satisfaction of this criterion is sufficient, but our investigation has revealed that, for registration purposes, an additional acceptance criterion is required. This is based on the object thickness and is presented below.

2.2.1. Object thickness and the required point density

A frequent practical problem when calculating the nearest point on a surface to a point in space is illustrated in Fig. 1, where P is the measured point and V_i, V_j and V_p are the

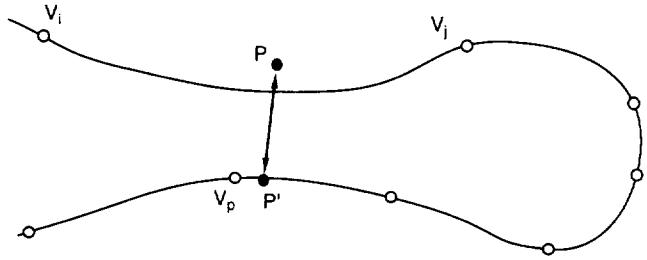

Fig. 1. Initial guess for MDSM search in finding the closest point.

vertices on the grid. This calculation starts with an exhaustive search for the nearest vertex, which is then used as the initial guess for the subsequent Multi-Dimensional Simplex minimisation (see the next section). The nearest point clearly lies in the segment between V_i and V_j, either of which is a correct initial guess. However, if the point density is too low, then the nearest found vertex may be V_p, which is on the wrong side of the model. The search for the nearest point would then minimise the distance PP', ultimately yielding a wrong result. This is particularly apparent when dealing with highly folded surfaces, as well as with components that are relatively flat but slender. Therefore the solution requires selection of a point density that is appropriate for the thickness of the given geometry.

Formally, object thickness (Fig. 2) and point density are defined as follows [8]:

> Let P be a set of points P_i, $i = 1,..,n$, on the surface. P is said to have density $1/\epsilon$ if and only if, for all points on the surface, there is a j, $1 \leq j \leq n$, such that P_j is within the open ball of centre P_i and radius ϵ.
> The thickness e of an object is the real positive number such that any maximal ball included either inside or outside the object has a radius larger than or equal to e.

The acceptance criterion in subdivision sampling based on thickness is explained with reference to Fig. 3, which shows the limiting case when $1/\epsilon = 1/2e$ and the given spatial point P_m is on the surface. Points V_i and V_j are its closest vertices on the near side of the object, while the point V_p is the nearest vertex on the far side of the object. Clearly, the situation where density $1/\epsilon > 1/2e$ will lead to vertex V_i or V_j being found as the nearest, both of which provide correct starting points in the subsequent MDSM minimisation. However, if this condition is not satisfied, then point V_p may be chosen, leading to a wrong point being found as the nearest.

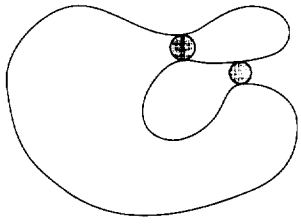

Fig. 2. Object thickness.

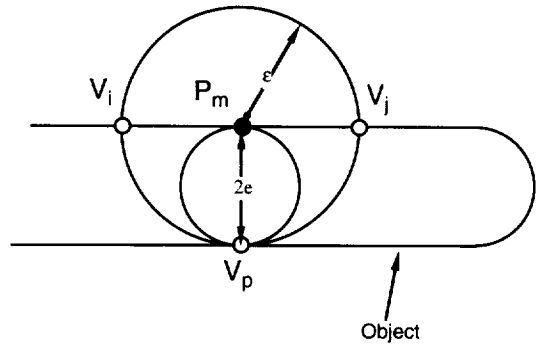

Fig. 3. Point density in relation to object thickness.

The thickness criterion therefore states that the density of the points on the surface must be greater than $1/2e$. The approximation obtained under this condition is homeomorphic. Note that this argument was derived bearing in mind accurate dimensional inspection, where the measurement noise size is assumed to be much smaller than the size of the smallest inspected feature on the object.

The implemented algorithm ensures that each region complies with this condition by considering the thickness of the NURBS surface in the directions of its parameters u and v. Thus, for each region, this results in different point densities in the two directions, according to the object geometry. This is also consistent with most modern CAD systems (such as ProEngineer) which do provide thickness as one of the surface interrogation functions.

2.3. Distance from a point in space to a NURBS surface

Computation of the closest point to a NURBS surface to a point in space P_m (x_m, y_m, z_m) involves minimisation of a scalar objective function Q, defined as the distance from that point to the surface:

$$Q = (x_m - x(u, v))^2 + (y_m - y(u, v))^2 + (z_m - z(u, v))^2 \quad (1)$$

The Multidimensional Downhill Simplex Method [9] (MDSM) has been chosen for this minimisation because it requires only evaluation of the function and not of its derivatives.

As with all iterative methods, the correct choice of an initial guess for MDSM minimisation is crucial, since the search would very often lead to a local minimum if the initial guess is not sufficiently close to the real nearest point. This problem was solved in the following manner. Prior to starting the determination of a closest point, a polygonal mesh which covers the parametric surface is calculated according to the subdivision sampling method described above. Each vertex is tagged with the corresponding parameter values (u, v) and both the vertex coordinates (x, y, z) and parameter values (u, v), are stored in a data structure. When the closest point is required we search for the nearest vertex on a polygonal mesh. Parameter values corresponding to the nearest vertex are then supplied to

MDSM as one point for the initial guess. Two other points are easily calculated as $(u + d, v)$ and $(u, v + d)$, where the constant 'd' is calculated from the mesh density.

In a number of situations in practice, the surface being manipulated is closed, such that points corresponding to a parameter value $u = u_{start}$ and $u = u_{end}$ coincide. In that case it may happen that the initial guess leads to a closest point on one edge of the surface while the real closest point is close to the other edge. The implemented solution solves the above problem by introducing a second initial guess. When the result of the first initial guess leads to a solution close to the surface edge, then the point on the opposite side is tried while the other parameter value is retained. Two cost functions are obtained and the smaller is selected. Its corresponding point is then accepted as the closest point on the surface.

2.4. Distance from a point in space to a polyhedron

The function of finding the nearest point on the polyhedral surface approximation to a given point in space was implemented through an algorithm which promotes the vertices (readily available) to be the representatives of their surroundings. This was made possible by the adoption of the correct termination criteria in subdivision sampling in order to produce a sufficiently dense set of vertices. The algorithm considers the vertex nearest to a given spatial point and the region which is formed by the facets in the vicinity of that vertex. It examines whether the projection of the given point falls on any of the facets surrounding the vertex and, if not, then the region under consideration is expanded to include the adjoining facets. The process may then be repeated until the true nearest point has been found.

3. Iterative closest point registration

The Best Fitting procedure, as applicable to dimensional inspection, can be defined as follows [4]:

> Given 3D data in a sensor coordinate system, which describes a data shape that may correspond to a model shape, and given a model shape in a model coordinate system in a different geometric representation, estimate the optimal rotation and translation that aligns the model shape and the data shape, minimising the distance between the shapes and thereby allowing determination of the equivalence of the shapes via a mean-square distance metric.

It has been proved [4] that, by using the collective distance between measured points and their closest points on a surface as the corresponding points, it is possible to build a best-fitting algorithm which will always converge to the nearest local minimum. Based on this, and on the above definition, the cost function to be minimised in best-fitting is the model–part distance, which can be expressed as:

$$F = \sum_{i=1}^{N} |\boldsymbol{q}_i - \mathbf{R}\boldsymbol{p}_i - \boldsymbol{t}|^2 \tag{2}$$

where $\boldsymbol{t}$ is the translation matrix, $\mathbf{R}$ is the rotation matrix, $\boldsymbol{p}_i$ is the ith measurement point, $\boldsymbol{q}_i$ is the closest point on the model, taken as the corresponding point, and N is the number of measured points.

This is the basis of the Iterative Closest Point (ICP) algorithm, which consists of the following steps:

1. For all the measured points, compute their closest points on the model shape.
2. Compute the transformation matrix which minimises a defined cost function.
3. Apply the transformation matrix to the measured points set.
4. If the change in the cost function is greater than a preset value, go to step 1.
5. Else stop.

Computationally, the most critical step was found to be the first, namely the calculation of the nearest points on the NURBS surface. For this reason, significant speed gains can be realised if the polyhedral approximation is used in the initial phase of best-fitting whereas the full NURBS model is used at the subsequent stages. This principle has led to the adoption of the dual representation of model geometry for ICP registration.

3.1. Search for the nearest vertex

With the adoption of the dual geometry representation, and following from the described methods for geometric modelling, the search for the nearest vertex is performed as a part of two tasks. Firstly, it provides an initial guess for MSDM search in finding the nearest point on NURBS. Secondly, it is performed when finding the nearest point on the approximation. Bearing in mind the large number of vertices typically contained in a polyhedral surface model and the lack of spatial information encoded in the topological structure, the search for the nearest vertex became the time critical task. Three major improvements have been implemented in order to increase the efficiency.

3.1.1. Adaptive window search

The first improvement is the *adaptive window search*. It is based on the original idea that for each measured point an exhaustive search for the closest vertex can be substituted by a search in the surrounding of some appropriately chosen vertex. Having in mind the iterative nature of the fitting algorithm, the search may be performed in the surrounding of the closest vertex found in the previous iteration. The search procedure is explained with reference to Fig. 4, in which $P_{m,k}$ denotes the position in relation to the model of the mth measured point, P_m, at the kth iteration, whereas

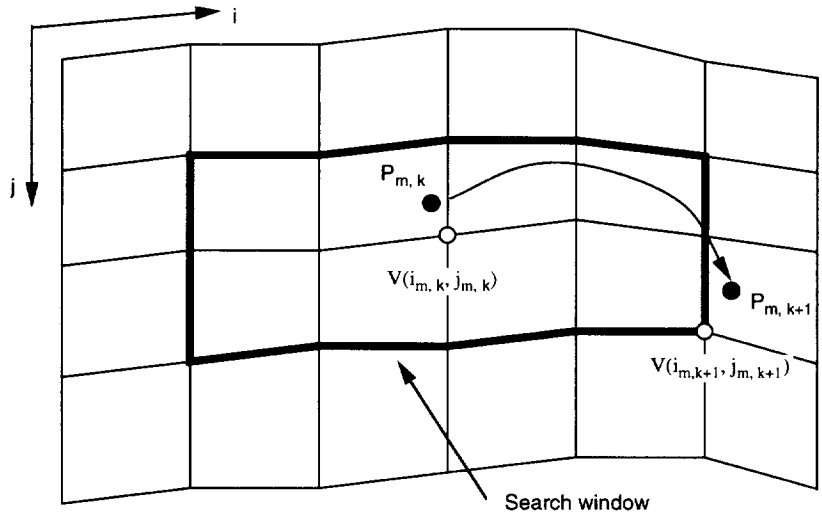

Fig. 4. Adaptive window search.

$V(i_{m,k}, j_{m,k})$ denotes the vertex on the grid which is found to be the closest to the point $P_{m,k}$.

Step 1: For each measured point $P_{m,1}$, $1 \leq m \leq N$, perform exhaustive search to find the nearest vertex $V(i_{m,1}, j_{m,1})$. Compute and apply the fitting transformation to move the point $P_{m,1}$ to position $P_{m,2}$ relative to the model.

Step 2: For each measured point $P_{m,2}$, again perform exhaustive search to find the nearest vertex $V(i_{m,2}, j_{m,2})$. Here, we notice that the same point would be found by searching in the window $\{i_{m,1} \pm |i_{m,2} - i_{m,1}|, j_{m,1} \pm |j_{m,2} - j_{m,1}|\}$, although this window is of course known only at the end of the search. Having in mind that the two objects are expected to move closer together at each fitting step and that the positional increments become smaller, it is therefore expected that this now known window will be sufficient for the next step.

Steps k ($k \geq 3$): For each measured point $P_{m,k}$, search can now be performed in the window

$$\{i_{m,k-1} \pm |i_{m,k-2} - i_{m,k-1}|, j_{m,k-1} \pm |j_{m,k-2} - j_{m,k-1}|\}$$

However, this has the drawback that for each point the closest vertices in the previous two steps have to be stored. Thus, in order to minimise the storage requirements, only the largest of the windows is used for all the points, the size of which is given by

$$\left\{i_{m,k-1} \pm \max_{m=1}^{N} |i_{m,k-2} - i_{m,k-1}|, j_{m,k-1} \pm \max_{m=1}^{N} |j_{m,k-2} - j_{m,k-1}|\right\}$$

The search window is adaptive because its size corresponds to the change in position at each step, and it rapidly reduces to include only eight adjacent vertices.

3.1.2. Multi-scale search

Unfortunately, the above solution still demands exhaustive search in the first two iteration steps in order to establish the required adjacency information. This is where the second improvement was introduced, which we call *multi-scale search* because it employs a coarse approximation of the object in the first two steps. The coarse approximation greatly reduces search time and, at the same time, solves the adjacency problem mentioned above. Adjacency information is computed (but not used) in the first two steps and is directly applied when switching to the 'window-based' method in the third step.

The number of vertices in the coarse approximation was chosen to be $C \log(N_v)$, where N_v is the original number of vertices and C is a suitable constant. This formula was adopted because it provides a suitable and reasonably uniform number of vertices for a wide range of N_v. Bearing in mind the computing speed of our hardware platform (SPARC-2), we used $C = 300$, so for the range $1000 < N_v < 100\,000$ we obtain $900 < 300 \log(N_v) < 1500$, while for faster machines a larger value of C may be more suitable. Of course, for $N_v < 1000$ multi-scale search brings little benefit.

3.1.3. Selective use of measured points

A further improvement in speed is achieved by allowing the user to choose the number of measured points to be used for each iteration. This is very important in the initial phases of alignment when, due to a considerable discrepancy, the use of large data sets does not improve the accuracy. Several thousand experiments were conducted [10] in which 10 000 points were measured. In each case, fitting was performed using a subset comprising 200–300 points. The subsequent

fitting of the full data set required only 1–2 further iterations.

3.2. Rigid body transformation

The second step in ICP, to find a rigid body transformation that minimises the least-square distance between the point pairs, is the essence of the best-fitting procedure. The method used to compute the transformation is Singular Value Decomposition (SVD), as suggested by Haralick [11]. The main reasons for this are that the SVD method is computationally very efficient and that it can be easily generalised to more than 3-D. A detailed proof of the SVD algorithm is provided by Haralick [11] and Arun et al. [12]. The SVD algorithm, as a method to solve constrained least-squares problems and its implementation, is presented by Golub and Van Loan [13] and Press et al. [9].

In some cases SVD will compute a reflection matrix, but this is easy to detect as the determinant of the rotation matrix becomes -1. In this case an alternative method is applied, in which the minimisation of the cost function is computed by solving the system of six nonlinear equations [14], all other steps in ICP remaining the same. Thus, in this case, the transformation is modelled using a function of six variables:

$$\boldsymbol{T} = f(t_x, t_y, t_z, \alpha, \beta, \gamma)$$

where t_x, t_y, t_z are translations along the x, y, z axes and a, b, g are the rotations. The system of six nonlinear equations to find the unknown variables is obtained from:

$$\frac{\partial F}{\partial t_i}(t_1, t_2, t_3, t_4, t_5, t_6) = 0 \quad \text{for } i = 1 \text{ to } 6 \tag{3}$$

where:

$$F = \sum_{i=1}^{N} |\boldsymbol{q}_i - \boldsymbol{T}\boldsymbol{p}_i|^2$$

and

$$(t_1, t_2, t_3, t_4, t_5, t_6) = (t_x, t_y, t_z, \alpha, \beta, \gamma)$$

The above derivatives of the cost function F can be obtained analytically. To solve the system of equations obtained in this way, the Variable Metric method [9] was applied. The results are the same as those obtained using SVD but, due to the iterative nature of the algorithm, the computational time is much longer. As a result, we primarily use the SVD algorithm and only when the reflection matrix is detected do we switch to the Variable Metric method.

3.3. Global convergence and termination criteria of ICP

ICP registration minimises the systematic component of the error while retaining the random component [15]. This means that the mean square error (MSE) after registration approaches the sample standard deviation of the measurement data set, which in turn (for reasonably large data sets) approaches the standard deviation of the measurement noise, σ. Furthermore, for inspection purposes it is readily assumed that the measuring instrument is sufficiently well characterised such that the value of σ is known. This argument allows that both detection of convergence to a local minimum and the ICP termination criterion be based on the known value of σ.

Convergence to a minimum is detected by comparing the change in MSE with a pre-specified value. Local minima are then detected as the situations when the MSE value is considerably larger than the known value of σ. In such situations a small local perturbation in position is generated and the subsequent convergence is examined.

The ICP procedure was set to terminate when the change in MSE became less than $10^{-2}\sigma$. This value is based on experience and in several thousand conducted experiments it produced good results [10]. Of course, this does not preclude using some other suitably chosen value in the early stages of registration, when a subset of measured points and the approximate model are used.

4. Registration performance

The most important aspects of the presented registration method will now be illustrated in relation to measurement of a particular aero engine turbine blade airfoil, Fig. 5. This component was considered to provide a useful registration performance benchmark for a number of reasons. Firstly, the airfoil shape is poorly conditioned for registration due to its strong symmetry in the z direction. Secondly, its

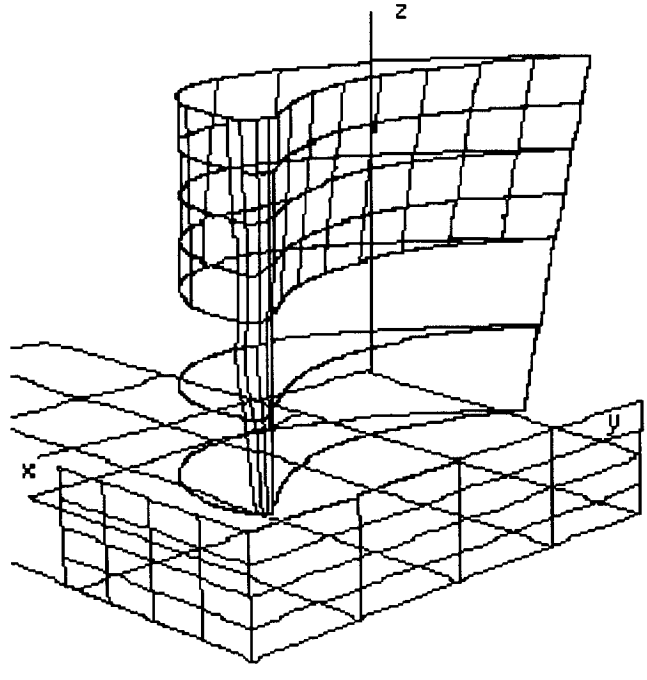

Fig. 5. CAD model of a turbine blade.

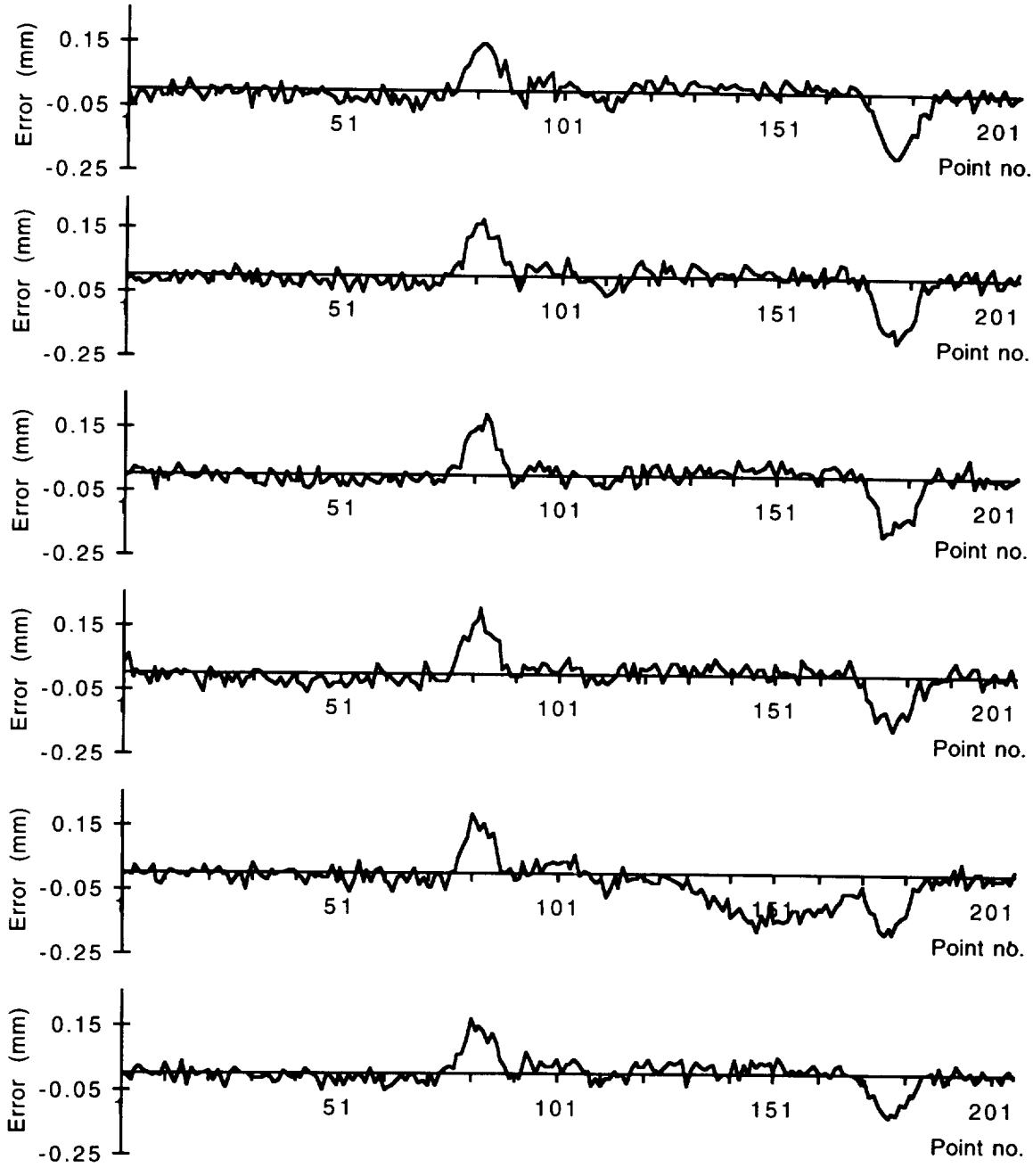

Fig. 6. Experimental results: error after ICP registration for each measured blade section.

surface is highly folded and the thinness of its shape potentially gives rise to problems in calculating the corresponding points, which the proposed method aims to overcome. Thirdly, the small radius of 0.3 mm at the trailing edge represents a fine feature which in practice needs to be measured and analysed. Finally, accurate measurement of turbine blades is a good example of an engineering application where the proposed method would bring immense practical benefits by eliminating the need for special tooling.

The turbine blade in question was measured using a conventional 3-axis coordinate measuring machine (CMM). The CMM was equipped with a laser triangulation probe (Matsushita LM200) which was mounted via a 2-axis indexable head. Special probe path planning software was also developed for this system, allowing accurate control of probe position relative to the object for each measured point. The measurements were performed in a number of parallel sections over the airfoil surface. Fig. 6 shows typical results obtained using this system, where the error magnitude after registration is displayed for each measured section. The regions of imperfection of the actual shape are clearly visible.

However, as direct knowledge of factors such as the amount of initial misalignment and the measurement error

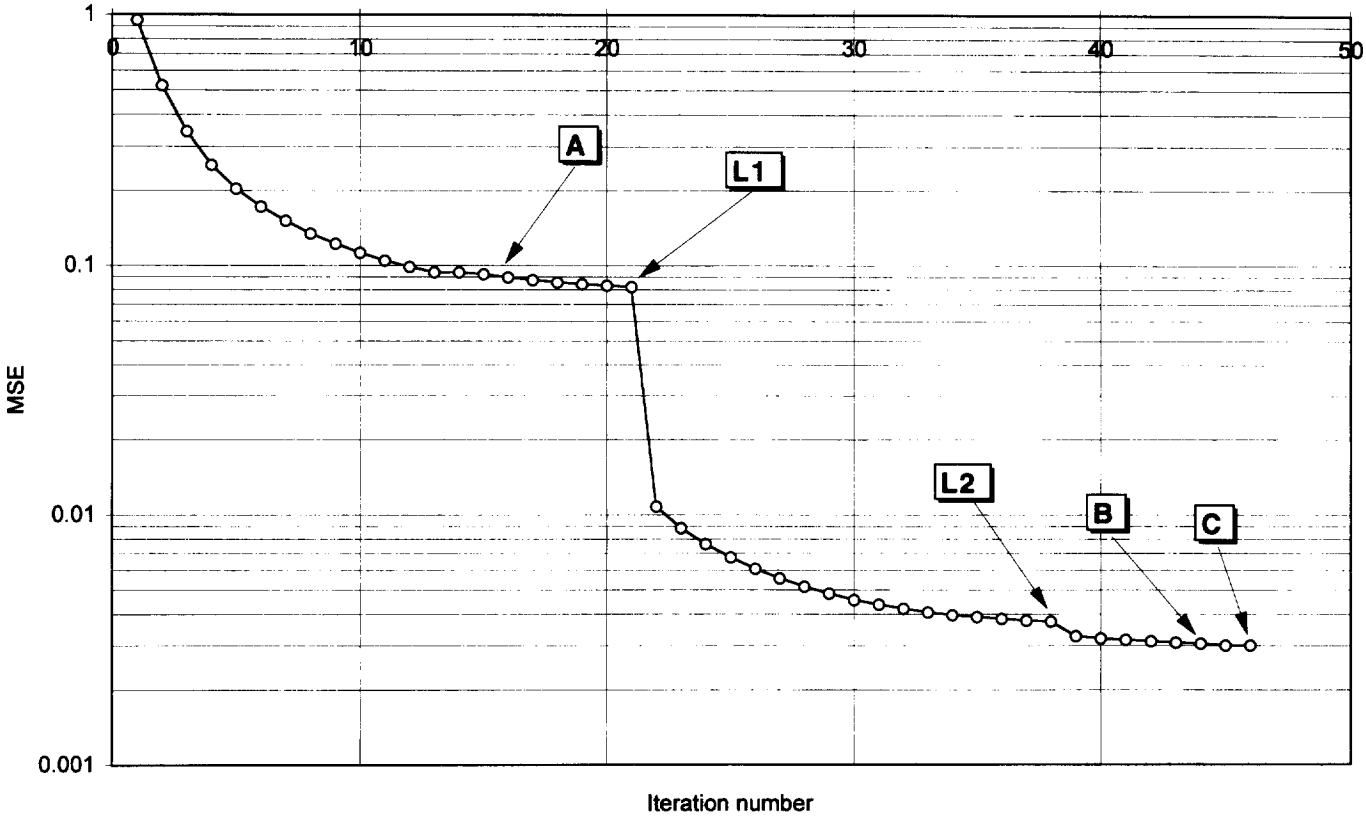

Fig. 7. Typical reduction of MSE vs iteration step number for different noise values.

at each point is not readily available with a physical system, an extensive simulation of the measurement process was also developed in order to better analyse the registration performance. The conduct of the simulation studies has been reported in detail by the authors [10], presenting a generic analysis of the influence of various factors on ICP registration. In this paper, on the other hand, we present the results that are considered to provide a validation of the proposed algorithms.

The measuring system accuracy was characterised by using a calibration sphere of known diameter. Application of the χ-test on calibration data proved that the measurement noise in this system was very closely represented as Gaussian. On this basis the measurements were simulated by introducing randomly generated Gaussian noise of known standard deviation σ. Initial misalignment, on the other hand, was simulated by applying a uniformly distributed random transformation in a certain range to the simulated measurements.

The graph in Fig. 7 shows typical progress of the ICP registration, in terms of the reduction in mean square error (MSE) at each iteration step for noise value $\sigma = 55$ μm. In this case the approximate polyhedral model comprised 5200 vertices; the measurement data set comprised 10 000 points. According to the described method, registration was performed by initially fitting a subset of 300 randomly chosen measurements to the approximate model, until convergence was detected such that the reduction in MSE was less than 10^{-2} (point A on the graph). Subsequently, the same point subset was fitted to the NURBS model until convergence was detected such that the reduction in MSE was less than $10^{-2}\sigma$ (point B). The final set of iterations was then performed by fitting the full set of measurements to the NURBS model, and the iterations were again terminated when reduction of MSE was detected to be less than $10^{-2}\sigma$ (point C). Points L1 and L2 represent local minima, which were overcome by applying a perturbation described previously.

Simulation studies [10] have shown that the ability of the ICP method to reach the global minimum depends on the number of measured points in relation to the object geometry. This is supported by the graph in Fig. 8, showing that the mean error in transformation parameter estimates tends to zero as the number of measured points increases. Each point in the graph is the average of 100 simulations with different initial misalignment. In this example, it is primarily the poor determinacy of the airfoil position in the z-direction that demands the use of a larger number of points.

A useful measure of registration performance may be provided by comparing the maximum measurement error in a data set with the maximum error identified after registration. This is particularly relevant to inspection, where the objective is that any errors introduced by best-fitting are minimised. This analysis was performed through simulation by sampling the nominal surface, introducing the random

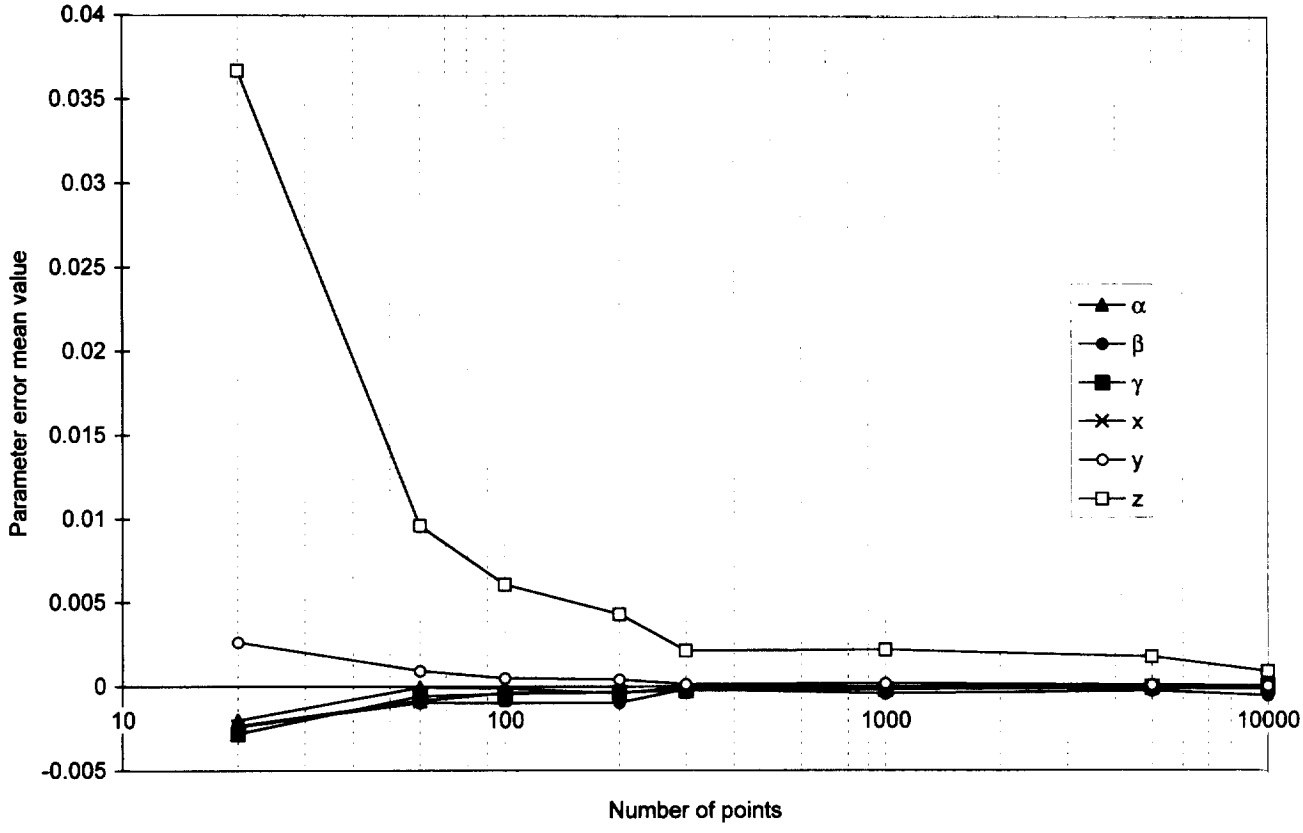

Fig. 8. Mean error in transformation parameter estimation vs number of measured points (noise $\sigma = 8$ μm).

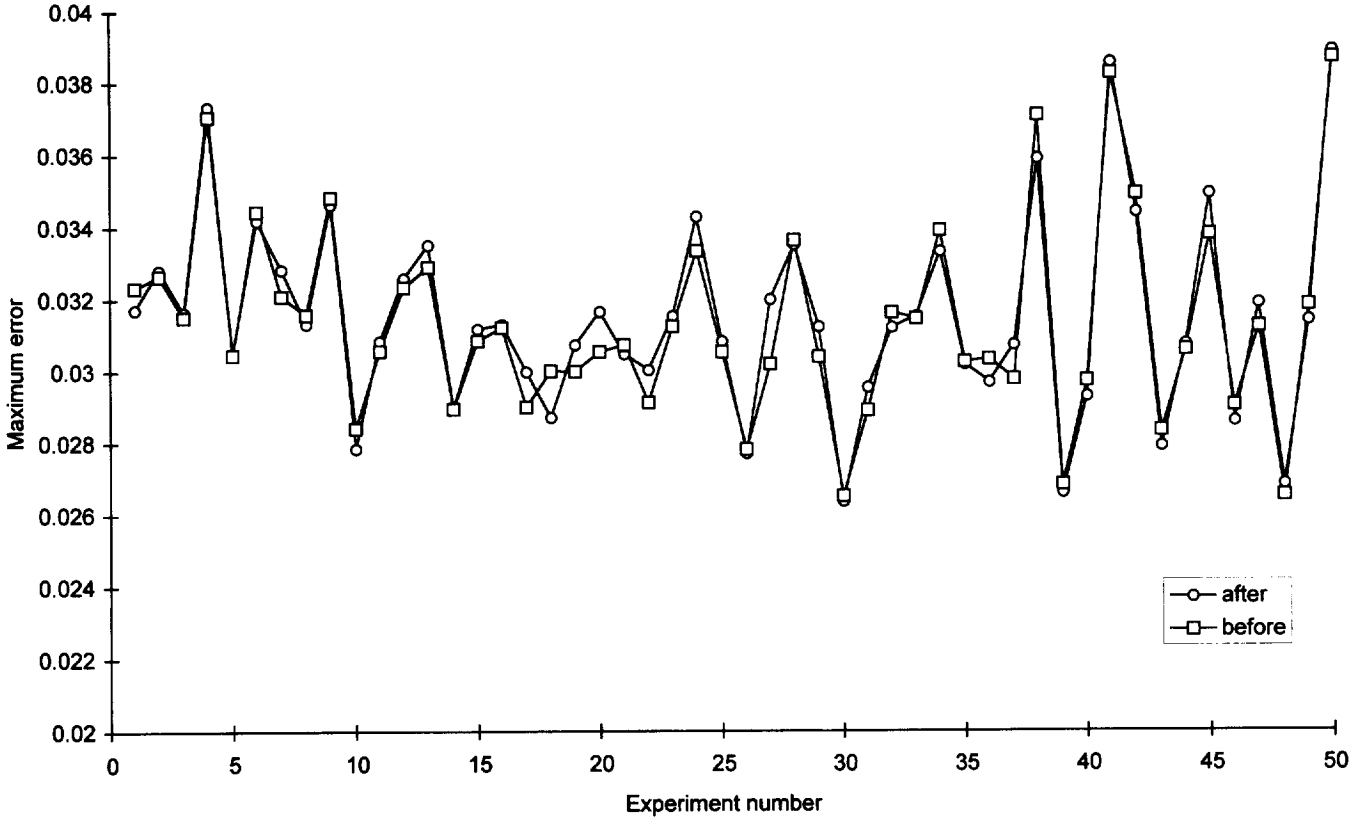

Fig. 9. Maximum error before misalignment and after ICP fitting in 50 simulations.

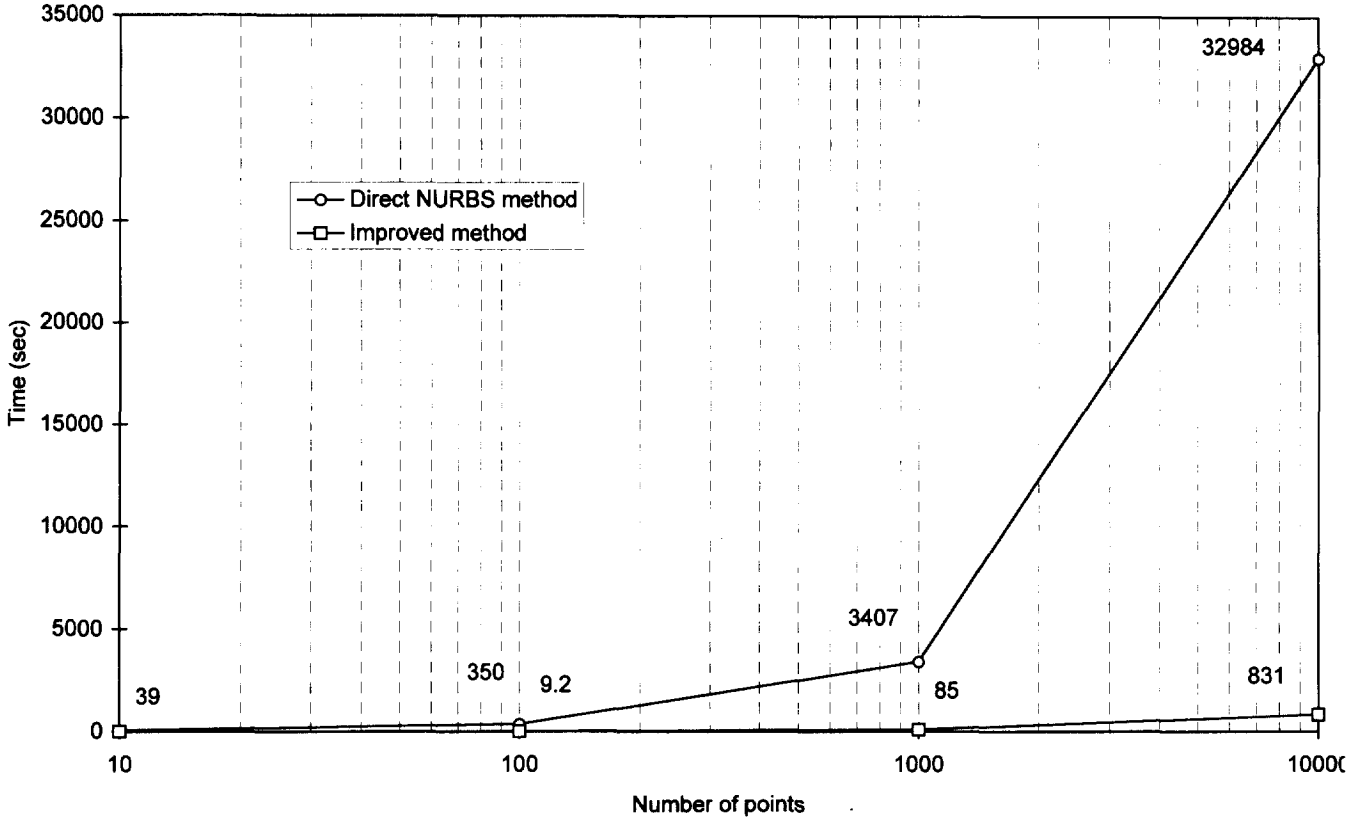

Fig. 10. Average computing time on SUN-SPARC2 workstation vs number of measured points for the implemented method and the direct method operating on NURBS.

measurement error at each point and recording the maximum error value in the set. After applying a random transformation to the points and the subsequent ICP registration, the maximum error was again recorded and compared to the original one. Sample results are shown in Fig. 9, which for clarity includes only 50 simulations, each with 10 000 measured points, noise $\sigma = 20\ \mu m$ and a misalignment range of $\pm$ 3 mm and $\pm$ 3°. The closeness of the two graphs indicates that the point measured with the largest error still results in the largest error after registration. Experiments involving misalignments in the range of $\pm$15 mm and $\pm$ 15° showed identical results.

Naturally, the question arises as to how the proposed registration method performs in the presence of gross outliers in the data set. In the context of ICP registration outliers may result from two causes, namely from wrongly determined point correspondences and from gross measurement error at certain points. As the results show, the former cause has been eliminated by adherence to the proposed thickness criterion in subdivision sampling, producing encouraging results even for a highly folded surface such as the blade airfoil. Further, the presence of outliers resulting from gross measurement error may be easily detected after fitting. We suggest that any point lying outside the range of $\pm 3\sigma$ be rejected and the fitting process repeated. With the system used for this work gross outliers were never detected, owing to the strict control of the measurement process at each point, but with other types of measuring system they may occur frequently.

Reports in the literature [4] suggest that more than 30 iterations are necessary to achieve accurate registration, and this was confirmed by our experiments. In both the experimental and the simulation studies involving the turbine blade, about 40 iterations were required on average to achieve the desired accuracy. This strongly underlines the time-critical task of finding the closest points.

The effect of the presented improvements on the computing speed is illustrated by the graph in Fig. 10. It shows the average computing time to perform registration against the number of measured points. Each point in the graph is the average result of 50 simulations involving random misalignment. Performance of the described method is presented in comparison with the direct method that operates on NURBS and uses the full data set at all steps, the registration accuracy being the same in both cases.

5. Conclusions

The paper has presented an implementation of NURBS surface registration using the Iterative Closest Point method. The original ICP method suggested by Besl was improved by fitting it to an approximate polyhedral model before switching to NURBS. The fitting speed was further

improved by using a subset of the measurement data at all but the final stages of the iteration.

The implementation also incorporates important computational improvements, which were realised in connection with finding the nearest vertex. These are the adaptive window search and the multi-scale search.

The paper has also suggested several original improvements for ICP registration. Firstly, the appropriate criteria for approximation of free-form surfaces have been derived. Secondly, it was suggested that the standard deviation of the measurement noise, if known, should be used as the basis for the ICP termination criteria. Finally, it is proposed that the knowledge of the measurement noise also be used for the detection of local minima, which are then overcome by the introduction of a local perturbation.

Experimental studies have shown that the method handles a full six degrees of freedom and that, for a reasonable initial misalignment, it converges to the global minimum.

Acknowledgements

This work has been partly supported by the EPSRC grant No. GR/L15449

References

[1] F. Mason, Ways to inspect jet-engine blades, American Machinist (August) (1989) 42–50.

[2] T.S. Newman, A.K. Jain, A survey of automated visual inspection, Computer Vision and Image Understanding 61 (2) (1995) 231–262.

[3] R. Jain, A. Jain, Report on Range Image Understanding. Workshop held in East Lancing, Michigan, 21–23 March 1988, in Machine Vision and Applications, Vol. 2 (1989) 45–60.

[4] P.J. Besl, N.D. McKay, A method for registration of 3-D shapes, IEEE Trans. Patt. Ann. and Mach. Intell. 14 (2) (1992) 239–255.

[5] R.H. Bartels, J. Beatty, B. Barsky, An Introduction to SPLINES for Use in Computer Graphics and Geometric Modelling, Morgan Kaufman, Los Altos, CA, 1987.

[6] B. Von Herzen, R.M. Barr, Accurate triangulations of deformed, intersecting surfaces, Computer Graphics 21 (4) (1987) 103–110.

[7] J.M. Lane, L.C. Carpenter, A generalised scan line algorithm for the computer display of parametrically defined surfaces, Computer Graphics and Image Processing 11 (3) (1979) 290–297.

[8] O. Faugeras, Three-dimensional Computer Vision: A Geometric Viewpoint, MIT Press, Cambridge, MA, 1993.

[9] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C, 2nd ed., Cambridge University Press, 1992.

[10] D. Brujic, M. Ristic, Analysis of free-form surface registration, Proc. IEEE Int. Conf. on Image Processing, Lausanne, Switzerland, 16–19 September 1996, Vol. 2, pp. 393–396.

[11] V.B. Haralick, Pose estimation from corresponding point data, in: H. Freeman (Ed.), Machine Vision for Inspection and Measurement, Academic Press, New York, 1989, pp. 1–41.

[12] K.S. Arun, T.S. Huang, S.D. Blostein, Least square fitting of two 3-D point sets, IEEE Trans. Patt. Anal. Mach. Intell. PAMI-9 (5) (1987) 698–700.

[13] G.H. Golub, C.F. Van Loan, Matrix Computations, 2nd ed., Johns Hopkins University Press, Baltimore, 1989.

[14] C.H. Menq, H.T. Yau, G.Y. Lai, Automated precision measurement of surface profile in CAD-directed inspection, IEEE Trans. Robotics and Automation 8 (2) (1992) 268–278.

[15] C.F. Dietrich, Uncertainty, Calibration and Probability: the Statistics of Scientific and Industrial Measurement, 2nd edn., Adam Higler, Bristol, 1991.