Algorithms for optimal partial matching of free-form objects with scaling effects

K.H. Ko,[a] T. Maekawa,[a,b] and N.M. Patrikalakis[a,*]

[a] Massachusetts Institute of Technology, Cambridge, MA 02139-4307, USA
[b] Yokohama National University, Yokohama 240-8501, Japan

Abstract

A free-form object matching problem is addressed in this paper. Two methods are proposed to solve a partial matching problem with scaling effects and no prior information on correspondence or the rigid body transformation involved. The first method uses umbilical points, which behave as fingerprints of a surface and their qualitative properties can be used for matching purposes. The second method uses an optimization scheme based on the extension of the KH curvature matching method [Comput. Aided Design 35 (2003) 913], first introduced in the context of a matching problem without scaling effects. Two types of curvatures, the Gaussian and the mean curvatures, are used to establish correspondences between two objects. The curvature matching method is formulated in terms of minimization of an objective function depending on the unknown scaling factor, and the rigid body transformation parameters. The accuracy and complexity of the proposed methods as well as the convergence for the optimization approach are analyzed. Examples illustrate the two methods.
© 2004 Elsevier Inc. All rights reserved.

1. Introduction

Matching of 3D objects is an important topic in computer graphics and modeling and several significant contributions have been made in developing matching

---

* Corresponding author. Fax: 1-617-253-8125.
 E-mail address: nmp@mit.edu (N.M. Patrikalakis).

methods of 3D *free-form* objects. Free-form can be interpreted in various ways. Campbell and Flynn [12] regarded free-form as "a general characterization of an object whose surfaces are not of a more easily recognized class such as planar and/or natural quadric surfaces." On the other hand, Besl [8] explained that "a free-form surface has a well defined surface normal that is continuous almost everywhere except at vertices, edges, and cusps." With these characteristics free-form enables us to represent various surfaces in digital form such as ship hulls, aircraft, and automobile bodies using non-uniform rational B-spline (NURBS) surface patches, polyhedra, and range data. However, matching for free-form objects still remains a challenging problem. It is referred to in terms of two different names depending on the situation. When matching is used in the context of computer-aided inspection, it is referred to as *localization*, whereas when it is used in the context of computer vision, it is referred to as *registration*.

Object matching can be achieved by various approaches. This paper focuses on object matching through a process determining a rigid body transformation (translation and rotation) and a scaling factor, which make two objects match as closely as possible. Such a matching problem can be formulated in terms of a minimization of an objective function and the choice of such an objective function can be quite flexible. In this work, the sum of the squared minimum distances between two objects is chosen for the objective function.

Correspondence search between two surfaces is a key issue in minimizing the objective function for determining the best transformation for matching. Correspondence can be established by calculating distinct features of one surface and then finding the same ones on the other. In that case, such features need to be extracted robustly and accurately, and carefully chosen for matching purposes. There are many candidates that may be used as distinct features among which shape intrinsic properties are frequently used for matching. They only depend on the geometric shape of the object, and are independent of parametrization and representation methods. Moreover, they are invariant under any rigid body transformation such as rotation and translation. When such distinct features cannot be obtained, then one has to search correspondence using methods based on iteration.

Matching problems can be classified based on several criteria. First, two types of matching can be considered: *global* and *partial* matching. Simply, the global matching is regarded as the matching for whole objects, while the partial matching is considered as the matching of part of objects. Matching problems can be further categorized based on the availability of correspondence or initial transformation information between two objects and the application of scaling. The classification of matching problems is summarized in Table 1. In the table, acronyms are used for simplification as follows:

- C: Correspondence information is provided.
- I, Initial information on correspondence is provided.
- N, No correspondence information is available.
- P, Partial matching.
- G, Global matching.
- WOS, Without scaling.
- WS, With scaling.

Table 1
Classification of matching problems

Criteria	Global matching		Partial Matching	
	Without scaling	With scaling	Without scaling	With scaling
Correspondence information	CGWOS	CGWS	CPWOS	CPWS
Initial information	IGWOS	IGWS	IPWOS	IPWS
No information	NGWOS	NGWS	NPWOS	NPWS

When correspondence information is provided, which is one of the types CGWOS or CPWOS, then the matching problem simply is reduced to calculation of a rigid body transformation [20,21]. If no correspondence is known, but a good initial approximation for the transformation is available (IGWOS or IPWOS), then popular iterative schemes such as the iterative closest point (ICP) algorithm [9] can be employed. However, when no prior clue for correspondence or transformation is given (NGWOS or NPWOS), the matching problem becomes more complicated. In this case, the solution process must provide a means to establish such correspondence information such as in [14].

Scaling is another factor that needs to be considered separately. If a matching problem involves scaling effects, then direct comparison of quantitative measures cannot be used any longer. For the global matching case, a scaling factor can be estimated by the ratio of surface areas and applied to resolve the scaling transformation. However, when it comes to partial matching, such area information becomes useless for the scaling factor estimation. When the correspondence information between two objects is known (CGWS or CPWS), the scaling factor between the objects can be easily obtained by using the ratio of Euclidean distances between two sets of corresponding points or areas. If an initial scaling value as well as a good initial approximation is provided (IGWS or IPWS), the ICP algorithm [9] or other optimization schemes such as the *quasi-Newton method* [42] which are modified to handle scaling, can be employed. The problem of NGWS type can be solved by the moment method using the principal moment of inertia or ratio of areas or volumes. However, we emphasize that few attempts have been made to solve problems of the NPWS type.

Our main contribution in this paper is the development of algorithms to solve a matching problem of the NPWS type. To the best of the authors' knowledge, this type of matching problem has not been discussed in the literature so far and this paper serves as the first step to address such a matching problem and provide appropriate matching algorithms. This work was motivated by the *Shape Intrinsic Watermarks for 3D Solids* project, which focuses on providing a method for protection of intellectual property of designed NURBS surfaces or solids bounded by NURBS surfaces in digital form [28,29]. Suppose that someone has inappropriately acquired part of a design surface and used it after transformation including uniform scaling without the consent of its owner. In this situation, identifying which part of the design has been stolen becomes a critical step for copyright protection, which can

be achieved by finding the best rigid body transformation and scaling factor which align the design surface, and the stolen piece as closely as possible. Other than such an application to ownership protection of CAD models, the solution methods to this matching problem can be also used to various applications such as inspection and model integration.

In this paper, we propose two methods to solve problems of the NPWS type for matching of points with a NURBS surface or a NURBS surface with a NURBS surface. When an object represented by data points is provided as input, a least squares surface fitting method is used to obtain a NURBS surface patch for input to the proposed methods. In this paper, we are generally focusing on 3D CAD models. So it is assumed that we have a design model defined by NURBS surface patches and try to match a surface patch against this design model. One method involves using umbilical points. There have been several papers addressing the possibility of using umbilical points for matching purposes [8,31]. But few results have been reported. The other method is an optimization scheme using the KH method [27]. The KH method is incorporated into an optimization process to deal with scaling. Since the umbilical point method requires a NURBS surface representation for extraction of umbilical points and the optimization method needs curvatures at three points, the proposed algorithms can deal with both matching cases in the same manner.

This paper is structured as follows: In Section 2, previous work is reviewed and in Section 3 mathematical definitions and concepts are summarized. Two correspondence search methods are introduced in Section 4. Full matching algorithms with scaling effects are presented in Section 5. Complexity, accuracy, and convergence of the algorithms are analyzed in Section 6 and several examples are presented in Section 7. Section 8 concludes this paper.

2. Literature review

Moment theories for 2D or 3D objects have been widely used for object matching and recognition. Invariant and non-invariant properties of moments have been studied and extensive literature exists on this topic. First order moment (center of mass) and second order moment (moment of inertia) are used for matching and pose estimation [34]. Also effort has been made to investigate various moment invariants [23,44]. For a comprehensive review of moment based approaches, see Prokop and Reeves [43]. Moment methods are simple and useful for various purposes but they cannot be used for partial matching.

Two-dimensional contours extracted from 3D objects are also used for matching and recognition. From the silhouettes of a 3D object, Mokhtarian [32] computes curvature scale space (CSS), and uses CSS maxima to find the best matching. Recently, Belongie et al. [5] proposed an algorithm to find correspondences between shapes. They sample points from the shape contour and associate each point with a shape descriptor, called the *shape context*, which describes the shape relative to each point. Establishing correspondence is reduced to finding the most similar shape contexts

from each shape under consideration. Their approach estimates the rigid body transformation which aligns two shapes as closely as possible. But it requires extraction of contours of a shape and does not deal with scaling effects.

Osada et al. [36,37] proposed the idea to reduce the shape matching problem to the comparison of 2D probability distributions which may capture the global shape of objects. Different *shape functions* based on global geometric properties are proposed to compute the unique signatures for each object, and the similarity between the distributions is investigated. Scaling can also be considered by this approach. However, such approach does not handle partial matching problems. Johnson and Hebert [25,26] proposed a new representation scheme for efficient object matching. 2D images which are associated with each point on the surface of an object are created with respect to a local basis (3D point with surface normal) on the surface point. These images, called *spin images* are obtained by accumulating two parameters which are describing other points on the surface of the objects. Correspondence is established by comparing the spin images. By doing this, they reduce 3D surface matching to 2D image matching, which can assist in handling the problems of clutter and occlusion. But scaling effects in matching are not taken into account. Another set of new representation schemes include the *splash* by Stein and Medioni [51], the *point signature* by Chua and Jarvis [15] and *COSMOS* by Dorai and Jain [18]. Barequet and Sharir [3] proposed an algorithm for partial surface and volume matching of 3D images represented as a set of points. Their approach deals with the rotation and translation separately. First, the best rotation is searched iteratively through a sequence of rotation in a steepest-descent style by scoring the ''goodness'' of rotation based on the *footprints* assigned to all points and advancing in the direction of the highest score. The best translation associated with the optimum rotation is computed by using a correlation function. Their approach also was tested with a new footprint, *directed footprint* in [4]. Their algorithm handles various types of matching problems efficiently but scaling effects are again not considered in their approach.

Registration/localization is another way to achieve matching of two objects. Depending on the type of matching problems introduced in Section 1, we can find suitable solution methods to find the best match between two objects. For CGWOS or CPWOS type of matching, we can use methods proposed in [2,20,21,53]. Since the correspondence between two objects is known, the calculation of the transformation is a relatively easy step and does not require iteration. On the other hand, with a good initial transformation, i.e., for IGWOS or IPWOS type problems, an optimum transformation can be found through iteration by minimizing an objective function of six independent variables (three translations and three rotations). Besl and McKay [9] used the iterative closest point (ICP) algorithm for registration of 3D shapes. Several alternative minimization approaches are discussed for improvement of convergence. The ICP registration is generalized to include Euclidean invariant features to provide a more effective algorithm by Sharp et al. [47]. A similar idea was proposed by Zhang [54]. He dealt with outliers, occlusion, appearance, and disappearance using a statistical method based on distance distribution. Patrikalakis and Bardis [38] provided an efficient method for accurate localization of free-form rational B-spline surfaces given

an initial estimate of the localization parameters. These iteration approaches, however, cannot guarantee optimal results in a global sense. Bergevin et al. [6] proposed a method to estimate the 3D rigid transformation between two range views of a complex object. They used a hierarchical surface triangulation representation through an iterative process, and performed the iterative least-squares computation of an incremental transformation proposed by Chen and Medioni [13] to find an optimum transformation based on the estimated hypothetical transformations. Chua and Jarvis [14] developed an object recognition method through registration. Their approach aligns two objects through registration assuming no prior knowledge of correspondence between two range data sets. They used the principal curvatures and Darboux frames and constructed a list of possible correspondences from which the best matching is sorted out through various searching algorithms. The method proposed in their paper extends the above work to cover a problem of partial correspondence search.

A scaling factor adds one more degree of freedom to matching problems so that solution schemes have to be modified to deal with it. By using a scaling parameter as is done in [24] with the offset parameter, it may be possible to find the transformation which may yield better results in matching. However, partial matching problems including scaling effects become more complicated. The general formulation of matching with scaling is presented in [8], and the ICP [9] can be extended to handle the matching problems of IGWS or IPWS type as long as a good initial transformation and a scaling value are provided.

All the algorithms developed so far can solve various types of matching problems as in Table 1. However, we wish to point out that no literature addresses partial matching problems with scaling effects and no prior clue for correspondence or transformation.

3. Mathematical preliminaries

3.1. Distance metric

The Euclidean distance between two points $\mathbf{p}_1$ and $\mathbf{p}_2$ is defined as

$$d_e(\mathbf{p}_1, \mathbf{p}_2) = |\mathbf{p}_1 - \mathbf{p}_2|. \tag{1}$$

We also define the minimum distance between a surface $\mathbf{r}$ and a point $\mathbf{p}$ as follows:

$$d_{sp}(\mathbf{r}, \mathbf{p}) = \min \{d_e(\mathbf{p}, \mathbf{p}_i), \forall \mathbf{p}_i \in \mathbf{r}\}. \tag{2}$$

3.1.1. Distance between a point and a parametric surface

Let us assume that we have a point $\mathbf{p}$ and a parametric surface $\mathbf{r} = \mathbf{r}(u, v)$, $0 \leqslant u, v \leqslant 1$. Then the squared distance function is defined as follows:

$$D(u, v) = |\mathbf{p} - \mathbf{r}(u, v)|^2 = (\mathbf{p} - \mathbf{r}(u, v)) \cdot (\mathbf{p} - \mathbf{r}(u, v)). \tag{3}$$

Finding the minimum distance between $\mathbf{p}$ and $\mathbf{r}$ reduces to minimizing (3) within the square $0 \leqslant u, v \leqslant 1$. Therefore, the problem needs to be broken up into several sub-problems which consider the behavior of the distance function at the boundary

and in the interior of the bound [39]. The sub-problems are summarized as follows: Find the minimum distances (1) in the interior domain, (2) along the boundary curves and (3) from the corner points. Among those minimum distances, the smallest one is chosen as the minimum distance between the point **p** and the surface **r**. A robust calculation of the minima of the distance function (3) can be achieved by the interval projected polyhedron (IPP) algorithm [39,48,55].

3.1.2. Distance metric function

A function can be defined using the distance metric (3) to formulate a localization problem. Suppose that we have a NURBS surface $\mathbf{r}_2$ and an object $\mathbf{r}_1$ represented by discrete points or surfaces. Then, the localization problem can be stated as finding a rigid body transformation (a translation vector **t** and a rotation matrix **R**) so that a global distance metric function

$$\Phi = \sum_{\forall \mathbf{p} \in \mathbf{r}_1} d_{\text{sp}}(\mathbf{r}_2, (\sigma \mathbf{R} \mathbf{p} + \mathbf{t})) \tag{4}$$

becomes minimal, where σ is a scaling factor.

3.2. Review of differential geometry

There are many textbooks available for theoretical treatment of the differential geometry of surfaces such as [17], and what follows is a summary of the relevant definitions and derivations used in this work.

Suppose we have a regular parametric surface

$$\mathbf{r}(u, v) = [x(u, v), y(u, v), z(u, v)]^{\text{T}}. \tag{5}$$

From the theory of differential geometry on surfaces, the first (*I*) and the second (*II*) fundamental forms [52] are defined by

$$I = \mathrm{d}\mathbf{r} \cdot \mathrm{d}\mathbf{r} = E\mathrm{d}u^2 + 2F\,\mathrm{d}u\,\mathrm{d}v + G\mathrm{d}v^2, \tag{6}$$

$$II = -\mathrm{d}\mathbf{r} \cdot \mathrm{d}\mathbf{N} = L\,\mathrm{d}u^2 + 2M\,\mathrm{d}u\,\mathrm{d}v + N\mathrm{d}v^2, \tag{7}$$

where **N** is the surface unit normal vector, E, F, and G the first fundamental form coefficients, and L, M, and N the second fundamental form coefficients. Here, we follow the sign convention that the positive normal curvature is defined such that the center of curvature of the normal section curve is on the opposite side of the surface normal [39]. The *Gaussian curvature (K)* and the *mean curvature (H)* of Eq. (5) are given by [39,52]

$$K = \frac{LN - M^2}{EG - F^2},$$

$$H = -\frac{1}{2}\left(\frac{EN + GL - 2FM}{EG - F^2}\right). \tag{8}$$

3.2.1. Umbilical points

An umbilic is a point on a surface where normal curvatures in all directions are equal. At that point, the principal directions are indeterminate, and the principal direction field shows a singular behavior there. The principal curvature functions can be expressed in terms of the Gaussian and mean curvature functions as follows [52]:

$$\kappa_{1,2}(u,v) = H(u,v) \pm \sqrt{H^2(u,v) - K(u,v)}, \tag{9}$$

where K and H are the Gaussian and the mean curvature functions, respectively.

Let $W(u,v) = H^2 - K$. The functions $\kappa_{1,2}(u,v)$ are real valued functions so that $W \geqslant 0$ must hold. From the definition of an umbilical point we have $W(u,v) = 0$. With these two conditions combined, we can infer that at an umbilical point, $W(u,v)$ has a global minimum [31]. Here, we assume that W is at least C^2 smooth. Then, the condition that W has a global minimum at an umbilic implies that $\nabla W = 0$. Therefore, at an umbilical point, the following three equations hold:

$$W(u,v) = 0, \quad \frac{\partial W(u,v)}{\partial u} = 0, \quad \frac{\partial W(u,v)}{\partial v} = 0. \tag{10}$$

Given a rational polynomial parametric surface patch such as a rational Bézier surface patch, we can set $W = P_N/P_D$, where P_N and P_D are polynomials in u and v. With the condition $W \geqslant 0$, $P_N \geqslant 0$ is assured since $P_D > 0$ is always true given the regularity condition of the surface. Then, Eq. (10) reduce to

$$P_N(u,v) = 0, \quad \frac{\partial P_N}{\partial u} = 0, \quad \frac{\partial P_N}{\partial v} = 0. \tag{11}$$

3.2.2. Classification of umbilical points

Umbilical points are classified into two types: *generic* and *non-generic*. Generic umbilical points maintain their properties under small perturbations of the surface, while non-generic umbilical points may lose their qualitative properties under small perturbations [7,31,39,46]. Generic umbilical points are further classified into three types: *star*, *monstar*, and *lemon* as shown in Fig. 1.
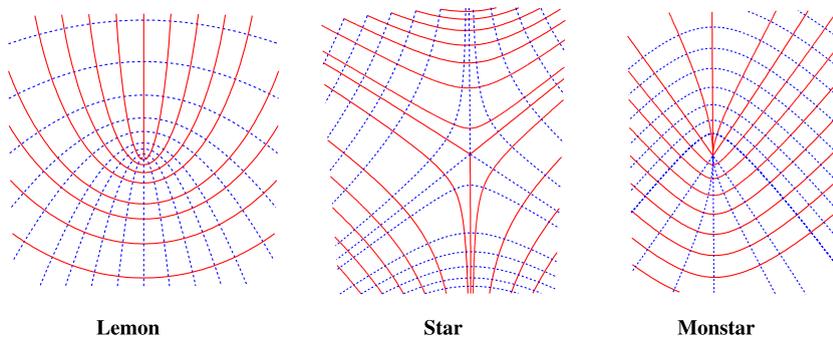

Lemon **Star** **Monstar**

Fig. 1. Three generic umbilics adapted from [39].

The umbilic diagram shown in Fig. 2 [41] is an easy way to distinguish the type of a generic umbilical point. In order to use this diagram, the local surface near the umbilical point has to be represented as a height function or the Monge form with respect to an local coordinate system as follows:

$$\mathbf{r} = (x, y, h(x, y)). \tag{12}$$

The height function $h(x, y)$ is Taylor expanded at the origin of the local coordinate system. Then we have

$$h(x, y) = -\frac{\kappa}{2}(x^2 + y^2) + \frac{1}{6}(ax^3 + 3bx^2y + 3cxy^2 + dy^3) + O(4). \tag{13}$$

Here, κ is the normal curvature at the umbilical point, see [31,39]. Let us set $C(x, y) = ax^3 + 3bx^2y + 3cxy^2 + dy^3$. This expression implies that the local structure is dominated by the coefficients of $C(x, y)$, i.e., by (a, b, c, d) and they determine the types of umbilical points [33,41]. It is convenient to represent the cubic polynomial $C(x, y)$ in the complex plane for analysis purposes. If we set $\zeta = x + iy$, then $C(x, y)$ becomes

$$\hat{C}(\zeta) = \alpha\zeta^3 + 3\overline{\beta}\zeta^2\overline{\zeta} + 3\beta\zeta\overline{\zeta}^2 + \overline{\alpha}\overline{\zeta}^3, \tag{14}$$

with

$$\begin{aligned}
\alpha &= \frac{1}{8}[(a - 3c) + i(d - 3b)], \\
\beta &= \frac{1}{8}[(a + c) + i(b + d)],
\end{aligned} \tag{15}$$

where $\alpha \neq 0$ and $\overline{\zeta}$ denotes the complex conjugate. We can express (14) in a coordinate system rotated about the normal vector without losing any essential features to make the coefficient of ζ^3 be equal to 1. Using $\xi = \alpha^{\frac{1}{3}}\zeta$, the Eq. (14) becomes
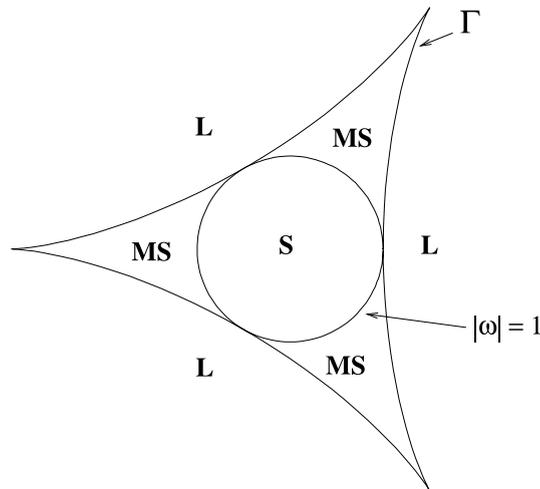

Fig. 2. The umbilic diagram adapted from [41].

$$\tilde{C}(\xi) = \xi^3 + 3\overline{\omega}\xi^2\overline{\xi} + 3\omega\xi\overline{\xi}^2 + \overline{\xi}^3, \tag{16}$$

where $\omega = \beta\alpha^{-\frac{1}{3}}\overline{\alpha}^{-\frac{2}{3}}$. This means that the cubic polynomial $C(x,y)$ can be parametrized with respect to a single complex variable ω [11]. Therefore, the variations of $C(x,y)$ can be mapped onto the complex plane [11,33,41]. Depending on the properties of $C(x,y)$, two characteristic lines are determined as follows:

- $|\omega| = 1$,
- $\Gamma : \theta \to -(2e^{i\theta} + e^{-2i\theta})$,

where Γ is a map from θ to the ω-plane. They divide the ω plane into sub-regions as shown in Fig. 2. Each sub-region corresponds to a specific type of an umbilical point. In Fig. 2, *S* means *star*, *MS monstar*, and *L lemon*. The graph Γ separates umbilical points of the lemon type from those of the monstar and star types, and the circle $|\omega| = 1$ umbilical points of the star type from those of the monstar and lemon types. If ω falls on a dividing curve, then the umbilical point is of the non-generic type. Using this diagram, the type of umbilical point is easily determined, see [11,33,41].

4. Correspondence search

The main objective of this section is to present methods for finding a correspondence between two objects (points and a NURBS surface or a NURBS surface and a NURBS surface) using surface intrinsic properties. Let us assume that we have two objects $\mathbf{r}_1$ and $\mathbf{r}_2$, and $\mathbf{r}_1$ is positioned in a different pose with no scaling effect. Recovery of the scaling factor is discussed in Section 5.

4.1. Surface fitting

Two cases are dealt with for matching in this paper: the point vs. surface and the surface vs. surface matching cases. Since the proposed algorithms are based on differential properties for matching, robust, and accurate extraction of them is an important step. When the surface vs. surface case is considered, the differential properties can be accurately calculated. However, estimation of umbilical points or curvatures from range data itself is a difficult problem. A surface fitting method such as a least squares fitting method or a method in [49] is preferred here because high order derivative properties can be calculated analytically from the fitted surface.

In this paper, we use a NURBS surface fitting method in the least squares sense [19,42] using the singular value decomposition method. When data points are arranged as a grid, then the chord length parametrization method can be adopted for parametrization of the data points, and the control points of a NURBS surface are obtained using the standard least squares method. If data points are unorganized, one can use the base surface method for parametrization proposed by Ma and Kruth [30]. A Gaussian (low pass) filter is used to eliminate high frequency noise in the data points to reduce the effects from the noise in the differential property calculation [35].

4.2. Umbilical Points

This method depends on the existence of umbilical points on an object. Several papers indicate the possibility of the use of umbilical points for matching and recognition [8,31]. An umbilical point is defined as a point, where normal curvatures in all directions are equal. Generic umbilical points are stable with respect to small perturbations so that they may act as fingerprints on a surface [7,31,39,45,50]. Besl [8] discussed the use of umbilical points as surface features for matching and suggested a possible matching method. Maekawa et al. [31] and Patrikalakis and Maekawa [39] proposed a robust method to calculate umbilical points and demonstrated their stability under small perturbations.

To locate umbilical points from range data, any type of surface fitting method may be used to obtain a NURBS surface patch, which is provided as input to the umbilical point detection algorithm [31,39]. Let us assume that we have an approximated surface r_1. When generic umbilical points exist on both surfaces r_1 and r_2, we can use the ω-plane to establish correspondence between the umbilical points. Let us assume that r_1 and r_2 have n_1, n_2 umbilical points, respectively. Then we calculate ω in Eq. (16) for each umbilical point, i.e., ω_{1j} $(j = 1, \ldots, n_1)$ for the umbilics on r_1 and ω_{2k} $(k = 1, \ldots, n_2)$ for the umbilics on r_2. Since the number of generic umbilical points on a surface is generally small, an exhaustive search scheme can be employed without loss of performance.

The searching process involves finding $(\omega_{1j}, \omega_{2k})$ which satisfy

$$\left| \omega_{1j} - \omega_{2k} \right| < \delta_\omega, \tag{17}$$

where δ_ω is a user-defined tolerance.

4.3. KH Method

The overall diagram of the KH method is shown in Fig. 3. The input of the process includes two objects and three pairs of the Gaussian and the mean curvatures at three different non-collinear locations. The algorithm yields the minimum value of Φ in the Eq. (4), and the corresponding rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$. Since no scaling effect is involved, we assume that the scaling factor $\sigma = 1$.

4.3.1. Step 10

Step 10 is to select three non-collinear points $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_3$ on r_1 away from the boundary of r_1 where each point has different, Gaussian K and mean curvature H values. At $\mathbf{m}_i$, we have K_i and H_i, where $i = 1, 2, 3$. Next, subdivide r_2 into rational Bézier surface patches $\mathbf{B}_j$ $(j = 1, \ldots, n)$ by inserting appropriate knots [22,40]. Then for each rational Bézier surface patch $\mathbf{B}_j$, we express K_j and H_j in the bivariate rational Bernstein polynomial basis from Eq. (8) using rounded interval arithmetic to formulate the problem. This allows us to use the interval projected polyhedron (IPP) algorithm [39,48] for solving nonlinear polynomial systems. For each pair K_i and H_i, we solve the following system of equations by the IPP technique.
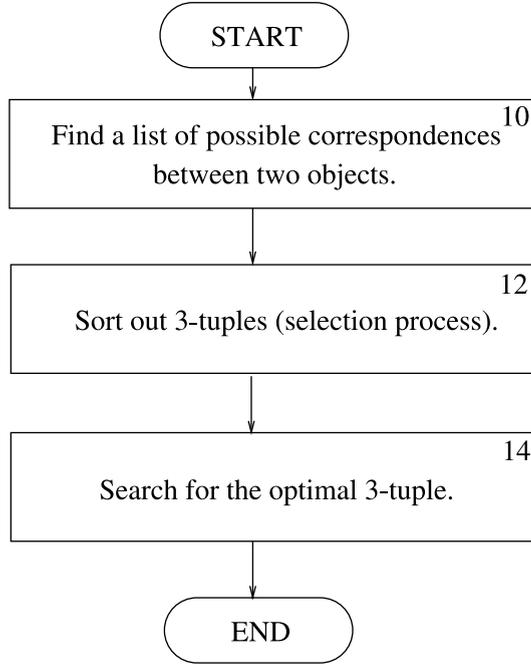
Fig. 3. A diagram of the KH method.

$$K_j(u,v) = K_i \pm \delta_K,$$
$$H_j(u,v) = H_i \pm \delta_H \quad (j = 1, \ldots, n \text{ and } i = 1, 2, 3), \tag{18}$$

where δ_K and δ_H represent the uncertainty of estimated curvatures from data points. For each pair of K_i and H_i, a list of roots $L_i = (u_{ik}, v_{ik})$, $(k = 1, \ldots, d_i)$ is obtained. Here, the integer value d_i varies depending on the tolerances δ_K and δ_H, and the shape of the surface for which Eq. (18) are formulated. However, typical values for different examples are from four to a few thousand [27].

4.3.2. Step 12 (selection process)

A simple pruning search based on the Euclidean distance can be applied to the selection process. We have three lists of candidate points, $L_i = (u_{ik}, v_{ik})$, $(k = 1, \ldots, d_i)$ from which one 3-tuple $(\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3)$

$$\mathbf{n}_1 = \mathbf{r}_2(u_{1k}, v_{1k}), \quad \mathbf{n}_2 = \mathbf{r}_2(u_{2k}, v_{2k}), \quad \mathbf{n}_3 = \mathbf{r}_2(u_{3k}, v_{3k}), \tag{19}$$

is selected to satisfy the following Euclidean distance constraints simultaneously

$$||\mathbf{m}_1 - \mathbf{m}_2| - |\mathbf{n}_1 - \mathbf{n}_2|| < \delta_{\text{select}},$$
$$||\mathbf{m}_2 - \mathbf{m}_3| - |\mathbf{n}_2 - \mathbf{n}_3|| < \delta_{\text{select}}, \tag{20}$$
$$||\mathbf{m}_3 - \mathbf{m}_1| - |\mathbf{n}_3 - \mathbf{n}_1|| < \delta_{\text{select}},$$

where δ_{select} is a user-defined tolerance.

4.3.3. Step 14

The correspondence information between each point $\mathbf{m}_i$ on $\mathbf{r}_1$ and $\mathbf{n}_i$ on $\mathbf{r}_2$ is established, from which a list of translation vectors and a rotation matrices can be obtained. We choose a translation vector and a rotation matrix which produces a minimum value of Eq. (4) with $s = 1$, see Ko et al. [27].

5. Algorithms with scaling effects

In this section, matching algorithms involving scaling are discussed. Two methods are proposed. One is to use umbilical points for matching and the other is to formulate a matching problem based on optimization.

The KH method described in Section 4.2 is designed to solve a matching problem when no scaling effect is included, and in this section it is extended to resolve scaling effects in the matching problem. Here, only uniform scaling is considered because non-uniform scaling may distort the geometry of a surface so that its functionality may be destroyed.

In case of global matching, a scaling value can be easily recovered by calculating the ratio of areas between two surfaces or volumes between two solids. However, for partial matching, comparison of any type of quantitative measures does not make sense. Only qualitative feature matching can be considered. The other possible approach is to search for a good match out of many possible solutions. Both methods are explained in the next subsections.

5.1. Use of umbilical points

The correspondence search explained in Section 4.2 only deals with qualitative aspects. Since the ω-plane is not affected by scaling, only qualitative correspondence can be established in the process. This implies that without a scaling factor applied, a rigid body transformation cannot be obtained for aligning two surfaces. Therefore, a scaling factor has to be estimated before any transformation is considered.

5.1.1. Method 1

Let us assume that we have two surfaces $\mathbf{r}_1$ and $\mathbf{r}_2$, where $\mathbf{r}_1$ is an approximated surface of input data points. The overall procedure is shown in Fig. 4.

In step 100, all generic umbilical points are located on both surfaces $\mathbf{r}_1$ and $\mathbf{r}_2$ using the IPP algorithm [31,39]. Non-generic umbilical points are not used in this process. If a generic umbilical point does not exist, this procedure cannot be applied.

In step 102, the correspondence search described in Section 4.2 is performed. The value ω in the complex plane is scale-independent so that qualitative correspondences can be built from this step. Suppose that matched pairs are denoted as m_k, $(k = 1, \ldots, n_k)$, where n_k is the number of matched pairs. Then when at least one pair is found, the next step 104 is performed. If no correspondence is established, then the algorithm stops, implying that the umbilical point method cannot be used in this case.

```
        ( START )
            |
            v
+---------------------------+
| Calculate umbilics    100 |
| for r₁ and r₂             |
+---------------------------+
            |
            v
+---------------------------+
| Correspondence pair search 102 |
| using w plane             |
| mₖ = (w₁ₖ , w₂ₖ)          |
+---------------------------+
            |
            v
+---------------------------+
| Estimate a scaling factor 104 |
| and scale r₁              |
+---------------------------+
            |
            v
         < mₖ < 3 >  --No-->  +---------------------------+
            |                 | Calculate a rigid body 106 |
           Yes                | transformation            |
            |                 +---------------------------+
            v                             |
+---------------------------+             v
| Translate r₁ and      108 |     < Transformation >
| match the normal vectors  |  No   < Exist? >
| for each pair             |             |
+---------------------------+            Yes
            |
            v
+---------------------------+
| Match the directions of 110 |
| lines of curvature        |
+---------------------------+
            |
            v
+---------------------------+
| Choose one that       112 |
| minimizes                 |
| the distance squared      |
| objective function        |
+---------------------------+
            |
            v
         ( END )
```
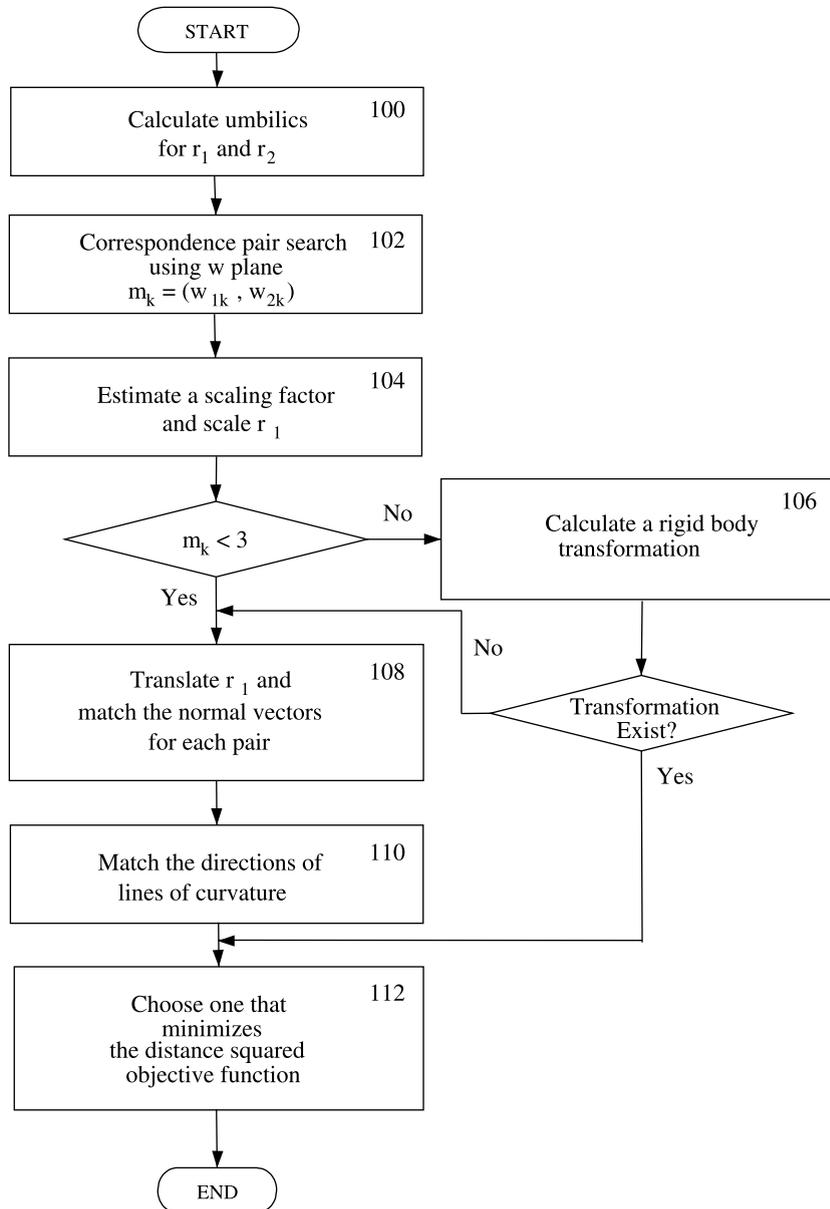
Fig. 4. A diagram for matching using umbilics.

Step 104 resolves the scaling transformation. To calculate a scaling factor, the normal curvatures are evaluated at the corresponding umbilical points on both surfaces $\mathbf{r}_1$ and $\mathbf{r}_2$. Then the ratio between them is obtained as a scaling factor. Suppose that a surface $\mathbf{r}$ is scaled with a scaling factor σ, denoted as $\mathbf{r}_\sigma$. Then the normal

curvature κ on $\mathbf{r}$ is scaled to be $\frac{\kappa}{\sigma}$ on $\mathbf{r}_\sigma$. Therefore using this relation, the scaling factor can be recovered.

In step 106, after sorting out candidate points, a rigid body transformation is estimated by using the unit quaternion method [20]. Since the number of matched pairs is more than three and if at least three pairs survive the selection process, the problem reduces to finding a rigid body transformation with three known corresponding pairs. Using the methods in [20] a rotation matrix and a translation vector can be calculated. If the matched pairs fail in the selection process, then the algorithm goes to step 108 which deals with the matching process with less than three matched pairs.

In step 108, the orientations of the normal vectors at the corresponding umbilical points are aligned. First, translate the scaled surface $\mathbf{r}_1$ by the difference between the positions of the matched umbilical points. Then, align the normal vectors at the umbilical points. The alignment of the normal vectors can be achieved by using the unit quaternion method [20]. Let us assume that we have two normal vectors $\mathbf{n}_1$ and $\mathbf{n}_2$ at the corresponding umbilical points for $\mathbf{r}_1$ and $\mathbf{r}_2$, respectively. The problem of matching the normal vectors can be stated as: rotate $\mathbf{n}_1$ around the vector $\mathbf{N}_n = \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\|\mathbf{n}_1 \times \mathbf{n}_2\|}$ by an angle θ formed by $\mathbf{n}_1$ and $\mathbf{n}_2$. The angle θ can be calculated by $\theta = \arccos(\mathbf{n}_1, \mathbf{n}_2)$, see [20] for details of rotation in the quaternion frame.

In step 110, matching of lines of curvature emanating from an umbilical point is performed. Depending on the type of the umbilical point, one (*lemon*) or three (*star and monstar*) lines of curvature pass through the umbilical point as shown in Fig. 1, and each direction can be determined by the structure of the cubic terms $C(x, y)$ as summarized in Section 3.2.2. The directions can be obtained by calculating angles of the lines of curvature with respect to the local coordinate system at the umbilical point [31,39]. Using the angles, vectors which indicate the directions of lines of curvature at the umbilical point can be obtained. These vectors are calculated at the matched umbilical points on $\mathbf{r}_1$ and $\mathbf{r}_2$. Suppose that the number of the direction vectors on $\mathbf{r}_1$ is n_{v1} and the number of the direction vectors on $\mathbf{r}_2$ is n_{v2}. Choose one vector from $\mathbf{r}_2$ and align all of the vectors on $\mathbf{r}_1$. This process produces n_{v1} matched cases among which one match is chosen that minimizes Eq. (4). This alignment is achieved by rotation around the normal vector in the tangent plane at the matched umbilical point. Therefore, the rotation method using the unit quaternion can be used in this process [20].

5.1.2. Method 2

The rigid body transformation can also be obtained by using the KH method described in Section 4.2 after the scaling transformation is resolved. The algorithm is the same as in Fig. 4 from step 100 through step 104. After the scaling transformation is resolved, the KH method can be used to find the rigid body transformation between two objects.

5.2. Optimization

The matching problem with scaling effects can be solved by an optimization technique. Since there is no quantitative measure that can be used to estimate a scaling

value, the solution to the matching problem with scaling effects has to resort to an optimum search method, which can narrow down the best estimate from a candidate set of solutions.

As shown in Section 4.3, the KH method finds the rotation matrix and the translation vector which minimize the objective function (4) under the assumption that a scaling factor is provided a priori. This implies that we can treat the KH method as a function of one variable, i.e., the scaling factor σ. Namely, steps 10, 12, and 14 in the diagram of Fig. 3 are grouped as a function f such that

$$f = \Phi(\sigma, \mathbf{R}, \mathbf{t}), \tag{21}$$

where Φ is the expression given in Eq. (4), σ the scaling factor, $\mathbf{R}$ the rotation matrix and $\mathbf{t}$ the translation vector. Since the rotation matrix and the translation vector are obtained from the KH method, we can reduce Eq. (21) to a function of a single variable σ, or $f = \Phi(\sigma)$. Hence, when σ is given as input, then f produces the best rigid body transformation (translation vector and rotation matrix) as well as the value of the objective function Φ for the input scaling factor σ. The behavior of the function f depends on the success of steps 10 and 12 in Fig. 3. When no solution to Eq. (18) is found in step 10, then we have no candidate points for step 12. In this case, the function f is penalized to yield a very large value. When we have candidate points in step 10, but the selection process fails under a given tolerance δ_{select}, then the tolerance is iteratively increased until any triplet is obtained or the following holds

$$\delta_{\text{select}} \leqslant \max(|\mathbf{m}_1 - \mathbf{m}_2|, |\mathbf{m}_2 - \mathbf{m}_3|, |\mathbf{m}_3 - \mathbf{m}_1|). \tag{22}$$

If there are more than two triplets, then, our method chooses one triplet which yields the smallest value of Φ, which can be performed effectively by using an efficient searching or sorting algorithm. When no triplet is found or the condition (22) is not satisfied, then the function f is penalized to yield a very large value.

Using the function $f(\sigma)$, the problem can be formulated as a one-dimensional optimization problem to find a scaling value which yields the minimum of f. A popular one dimension optimization scheme, the Golden section search [42] can be employed to solve it. An initial bracket $[a, b, c]$ of the scaling factor is provided which contains an optimum value, and satisfies $f(a) > f(b)$ and $f(c) > f(b)$. Suppose this bracket is I_0. The Golden section search starts with I_0 and continues while the size of an interval containing the optimum value σ is larger than a user defined tolerance which determines the size of the interval. Once the size of the interval becomes less than the tolerance, the search stops and the interval is reported to enclose the optimum value.

6. Analysis of algorithm

In this section, the proposed algorithms are analyzed in terms of time complexity and accuracy. The convergence of the optimization method is also discussed.

6.1. Complexity

6.1.1. Surface fitting

The least squares fitting method requires solving a system of linear equations. Suppose that there are r input data points and the number of the control points of an approximated NURBS surface patch is c. Then the solution algorithm of the singular value decomposition takes $O(rc^2 + c^3)$ multiplications [10,19]. In general, the singular value decomposition method is slower than solving the normal equations. However, it is more stable and reliable.

6.1.2. IPP algorithm

The total asymptotic computation time per step is $O(nlm^{l+1})$ [39,48], where n is the number of the nonlinear equations that need to be solved, l the number of the independent variables, m the maximum degree of the variables. Since the total number of steps depends on a user-defined tolerance, it is hard to predict how many steps will occur in advance. But for analysis purposes, we can use a constant α to indicate the total number of the steps performed in the algorithm. Then the asymptotic time complexity becomes $O(\alpha nlm^{l+1})$.

6.1.3. Calculation of umbilical points

Calculating isolated umbilical points from a Bézier surface patch depends on the degrees of the surface. The governing equations that have to be solved are Eq. (11), and the IPP algorithm is employed for the solution to a system of the nonlinear equations. Let us assume that the degrees of a Bézier surface are d_u and d_v in u and v directions. In this case, the maximum degrees of the each governing equation in u and v directions are proportional to the degrees of the input surface. The number of the governing equations is three and the number of independent variables is two. Therefore, the time complexity for the calculation of umbilical point reduces to $O(m^3)$ per step, where $m = \max(d_u, d_v)$.

In general, a surface is represented in NURBS form. Therefore, in order to apply the IPP algorithm to locate umbilical points, the surface needs to be subdivided into Bézier patches by the knot insertion algorithm [39,40]. Suppose, a surface has $c_u \times c_v$ control points. Then the total number of subdivided Bézier surface patches is proportional to $c_u c_v$. Therefore, it is concluded that given a NURBS surface of degree d_u and d_v in u and v directions with $c_u \times c_v$ control points, the total complexity reduces to $O(c_u c_v \alpha m^3)$, where $m = \max(d_u, d_v)$, and α is the number of iteration steps in the IPP algorithm.

6.1.4. Umbilical method

The time complexity of the matching process of the umbilical point method depends on the number of isolated umbilical points on the surfaces. Suppose that n_1 and n_2 are the number of umbilical points on the model and target surfaces $\mathbf{r}_1$ and $\mathbf{r}_2$. Then the search of correspondence takes $O(n_1 n_2)$ time in the worst case. However, in general, n_1 and n_2 are small integers. Therefore, the elapsed time is typically negligible.

6.1.5. Optimization method

The elapsed time of the optimization method relies on the number of iterations in the IPP algorithm and the tolerance of the Golden section search. Therefore, it is reasonable to consider the time complexity of one iteration of the proposed method. The complexity of the KH method consists of two parts: the IPP algorithm and the selection process. The IPP algorithm solves Eq. (18) using the auxiliary variable method [39]. Therefore, the number of equations and the number of independent variables are three. Suppose that the degrees of a NURBS surface are d_u and d_v and denote the total number of steps performed by the IPP algorithm as α, which depends on the tolerance provided by the user. Since the maximum degree of each equation is proportional to the degree of the surface, the total complexity for the IPP algorithm becomes $O(\alpha m^4)$, where $m = \max(d_u, d_v)$. The selection process takes $O(d_1 d_2 d_3)$, where d_i is the number of points of L_i in Section 4.3.2. Therefore, the overall time complexity is $O(\alpha l m^4 + d_1 d_2 d_3)$ per iteration of the Golden section search.

6.2. Accuracy

In this section, accuracy of both matching methods is discussed.

6.2.1. Umbilical point method

The accuracy of the umbilic method depends on that of the computation in locating umbilical points. The IPP algorithm requires a tolerance δ_{umb} which limits the size of intervals containing roots. Therefore, the tolerance δ_{umb} becomes the expected maximum accuracy bound of the umbilic method.

The quantitative estimation of the normal curvature at an umbilical point is important to recover a scaling factor. Since, in general, the curvature value is hard to be estimated from a set of data points, the umbilical matching method may not yield a satisfactory result. However, the result can be used as an initial estimate for an optimization method such as the ICP or the method proposed in this paper, see Section 5.2.

6.2.2. Optimization method

The accuracy of this method depends on various tolerances that need to be provided as input to the algorithm. First, the KH method requires two different tolerances δ_{IPP} and δ_{select}. The tolerance δ_{IPP} is provided as input to the IPP algorithm which limits the size of intervals of roots. The tolerance δ_{select} is used in the selection process of Section 4.3.2. However, the tolerance δ_{select} does not affect the accuracy of the final result because it is used as a value to sort out 3-tuples from a set of candidate points. Therefore, the tolerance δ_{select} can be a quite large number compared to δ_{IPP}. The additional tolerance δ_G, which is used by the Golden section search, is the one which affects the accuracy of the result such that it restricts the size of an interval which contains an optimum scaling factor.

The tolerance δ_G needs to be carefully chosen [42]. The algorithm is designed to stop when the size of a bracket becomes smaller than δ_G during the iteration. But how small the tolerance can be is an important issue that needs to be clarified. As

Press et al. [42] indicate, the smallest tolerance that can be used for the Golden section search is $\sqrt{\epsilon}$, where ϵ is the machine precision. The smaller tolerance than $\sqrt{\epsilon}$ results in unnecessary subdivisions of the bracket so that the overall performance deteriorates.

6.3. Convergence of the optimization method

The function $f = f(\sigma)$ is not necessarily smooth so that the function is not suitable for optimization methods such as the *parabolic interpolation in one dimension* [42], which can find an optimum value more efficiently when a function is smooth. Instead, a slow, but robust method, the Golden section search in one dimension is employed for this optimization problem. This optimization technique is designed to cope with the worst possible case [42] and narrow down the interval which surely contains an optimum value. It is known that the Golden section search converges linearly to an interval of user-defined size which surely contains an optimum (local minimum) value. The strategy of the proof is to show that the interval in the subsequent step which contains an optimum value decreases by a factor of γ (< 1). An assumption is made that three points a, b, and c in a bracket $[a, b, c]$ satisfy the following conditions.

$$f(a) > f(b), \quad f(c) > f(b). \tag{23}$$

In addition, suppose that w is $3 - \sqrt{5}/2$. This is called *the Golden ratio* which is used in the subdivision of the interval. We have an initial bracket $[a_0, b_0, c_0]$ which contains an optimum value and satisfies (23). At the nth step, suppose that we have an interval $[a_n, b_n, c_n]$ and at a_n, b_n, and c_n the conditions (23) are satisfied. Moreover, we have $(b_n - a_n)/(c_n - a_n) = w$. The interval l_n is calculated as $l_n = |c_n - a_n|$. At the next step, i.e., $(n + 1)$th step, a value x_n is selected which satisfies the ratio $(x_n - a_n)/(c_n - a_n) = 1 - w$ and then the function $f(x_n)$ is evaluated there. If $f(x_n) > f(b_n)$, then an interval $[a_n, b_n, x_n]$ is chosen. If $f(x_n) < f(b_n)$, then the other interval $[b_n, x_n, c_n]$ is taken. These two conditions guarantee that the selected interval encloses the optimum value all the time. First, let us assume that the interval $[a_n, b_n, x_n]$ is selected. The size of the interval is $l_{n+1} = |x_n - a_n|$ which is equivalent to $(1 - w)l_n$. Since $1 - w < 1$, l_n is decreased to l_{n+1} by a factor of $1 - w$. Next, consider the interval $[b_n, x_n, c_n]$. The size of the interval is $l_{n+1} = |c_n - b_n|$ which is also equal to $(1 - w)|c_n - a_n|$, i.e., $(1 - w)l_n$. Therefore, in both cases, the size of the interval l_n is reduced by a factor of $1 - w$ (< 1).

7. Examples

7.1. Matching with scaling

Two methods are tested with several numerical examples. The formulation for the Eqs. (11) and (18) are performed and solved by using the IPP algorithm in interval arithmetic for robustness reasons [1], and the rest of the calculation is done in double

precision. For simplicity all numbers are shown rounded at the fourth digit after the decimal point. A linux machine with 1.6 GHz CPU and 512 Mbytes was used for calculation.

7.1.1. Umbilical point matching

In this section, matching through umbilical points is demonstrated with an example. Suppose we have a set of data points $\mathbf{r}_1$ and a surface $\mathbf{r}_2$. The surface $\mathbf{r}_2$ shown in Fig. 5 is a bicubic B-spline surface with 64 (8×8) control points enclosed in a box of 25 mm $\times$ 23.48 mm $\times$ 11 mm. It has three star type umbilical points also as shown in Fig. 5, and the parametric values of the umbilical points in interval arithmetic from the IPP algorithm are summarized in Table 2. The elapsed time to calculate the umbilical points is 81 s. The center values of the interval roots representing umbilical points in Table 2 and the corresponding ω values are shown in Table 3. The point set $\mathbf{r}_1$ shown in Fig. 6 is approximated with a bicubic B-spline surface patch of 256 (16×16) control points. It takes 20 s to obtain the approximated surface. It has one umbilical point of star type as shown in Fig. 6. The root is ([0.207059775021701, 0.207059851944778], [0.684685549876914, 0.684685626799991]) in interval arithmetic and the elapsed time is 524 s. The center value of the interval and ω values are given in Table 4.
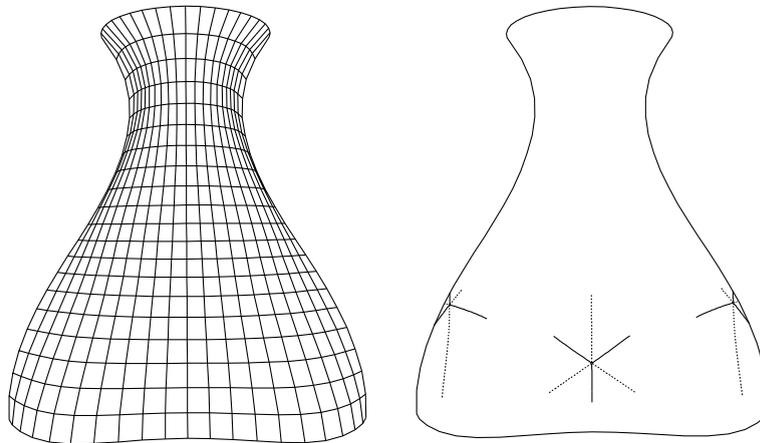

Fig. 5. Surface $\mathbf{r}_2$ and its umbilics.

Table 2
Umbilical points in interval arithmetic

Estimated umbilics for surface $\mathbf{r}_2$ using the IPP algorithm	
No.	(u, v)
1	([0.748157559043998, 0.748157794338109], [0.0280931620263166, 0.0280933620263166])
2	([0.860717315099411, 0.860717550393523], [0.4999999, 0.5000001])
3	([0.748157559043993, 0.748157794338105], [0.971906637973675, 0.971906837973676])

Table 3
Umbilics and ω values for $\mathbf{r}_2$

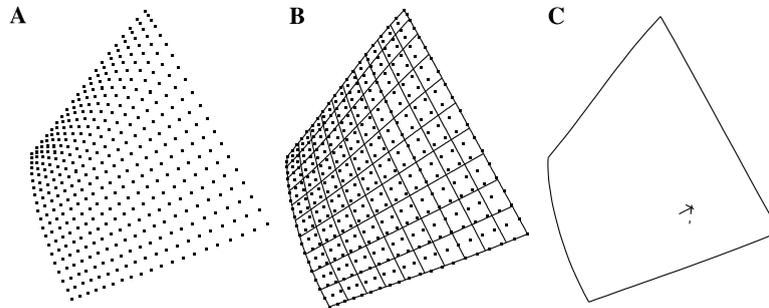| Surface $\mathbf{r}_2$		
No.	(u, v)	$\omega = (x + \mathrm{i}y)$
1	(0.748, 0.028)	$0.094 - 0.069\mathrm{i}$
2	(0.861, 0.5)	$0.151 - 0.261\mathrm{i}$
3	(0.748, 0.972)	$0.094 + 0.069\mathrm{i}$

Fig. 6. (A) Input point set $\mathbf{r}_1$. (B) Approximated surface of $\mathbf{r}_1$. (C) Located umbilical point (star type).

Table 4
An umbilic and ω value for $\mathbf{r}_1$

Surface $\mathbf{r}_1$		
No.	(u, v)	$\omega = (x + \mathrm{i}y)$
1	(0.207, 0.685)	$0.151 - 0.261\mathrm{i}$

Each ω value is plotted in the complex plane as shown in Fig. 7. We can find out that the umbilical point of $\mathbf{r}_2$ matches the number 2 umbilical point of $\mathbf{r}_1$ by comparing their complex ω values. Since a correspondence has been found, a scaling factor can be estimated by using the normal curvatures at the corresponding umbilical points on both surfaces. The normal curvatures at the corresponding umbilical points of $\mathbf{r}_1$ and $\mathbf{r}_2$ are $\kappa_1 = 0.334 \times 10^{-2}$ and $\kappa_2 = 0.113 \times 10^{-2}$, respectively. So the scaling factor σ can be calculated as $s = \frac{\kappa_1}{\kappa_2} = 2.941$. This scaling factor is applied to $\mathbf{r}_1$ and translated by the difference of the positions between the two corresponding umbilical points to get $\mathbf{r}_1'$. The next step is to align the normal vectors $\mathbf{n}_1$ for $\mathbf{r}_1'$ and $\mathbf{n}_2$ for $\mathbf{r}_2$. The alignment can be done by rotating $\mathbf{r}_1'$ by the angle between $\mathbf{n}_1$ and $\mathbf{n}_2$ around a vector $\mathbf{N_n} = \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\|\mathbf{n}_1 \times \mathbf{n}_2\|}$. The angle is 0.239 (rad) for this example. In order to match lines of curvature passing umbilical points, one direction of the lines of curvature from the surface $\mathbf{r}_2$ is selected. Let us denote the selected direction vector as $\mathbf{v}_2$
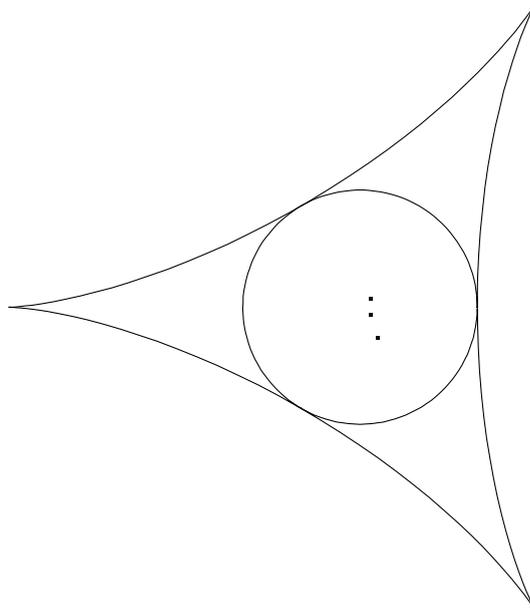
Fig. 7. Umbilical points on the ω-plane.

$$\mathbf{v}_2 = (-0.821, -0.021, -0.570). \tag{24}$$

Then three rotation angles between $\mathbf{v}_2$ and the directions in Table 5 at the umbilical points on $\mathbf{r}_1'$ are calculated as in Table 6. Here, the angles are measured from $\frac{\partial \mathbf{r}_1'}{\partial u}$ at the umbilical point. Matching the directions of lines of curvature is done by rotating $\mathbf{r}_1'$ around the normal vector in the tangent plane at the umbilical point. The rotation of $\mathbf{r}_1'$ by the first angle 1.105 rad yields the best match as shown in Fig. 8. The relative measure of the maximum error can be calculated by dividing the maximum distance error by a square root of the surface area of the surface, which is 0.011.

Table 5
Angles and directions of lines of curvatures

No.	Angle (rad)	Direction
1	0.964	(0.141, −0.001, −0.990)
2	−0.964	(0.879, 0.017, 0.477)
3	1.669×10^{-15}	(0.893, 0.014, −0.450)

Table 6
Rotation angles for matching lines of curvature

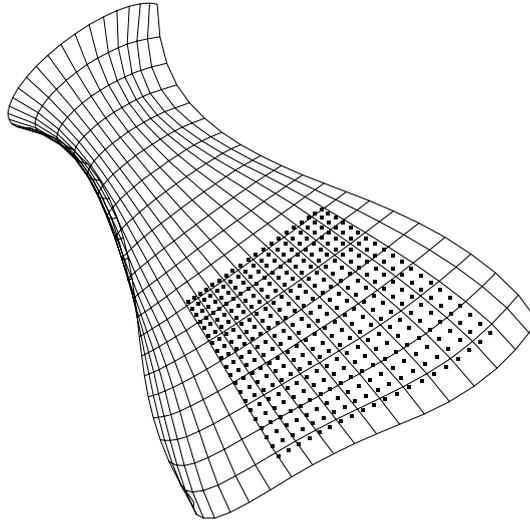|	1	2	3
Angle (rad)	1.105	3.033	2.069

Fig. 8. Localized points onto the surface.

7.1.2. Optimization

A few examples are presented for demonstration of the optimization method. The first example is to match two objects shown in Fig. 9. Data points in Fig. 9B have been scaled and transformed. A bicubic B-spline surface with 324 (18×18) control points is used to approximate the data points to calculate the Gaussian and the mean curvatures at three seed points shown as circles in the figure. The fitting step takes
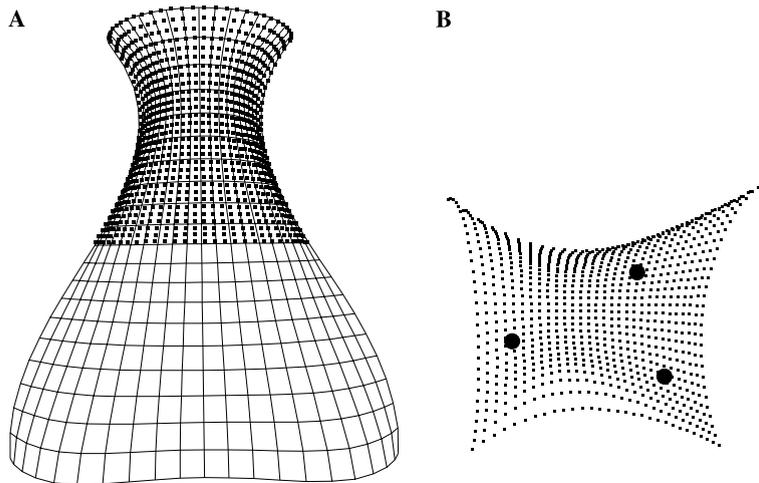

Fig. 9. Example 1 for the optimization method. (A) Matched surfaces. (B) Input points and three selected seed points for matching.

99 s. The bottle surface shown in Fig. 9A is a bicubic B-spline surface with 64 (8 × 8) control points, enclosed in a box of 25 mm × 23.48 mm × 11 mm. The problem here is finding a scaling factor and a rigid body transformation which make the two objects match as closely as possible. In this example, $\delta_{IPP} = 0.5$ was used for the IPP algorithm tolerance and $\delta_G = 0.001$ for the Golden section search tolerance. For an initial interval for the Golden section search, $[0.3, 1.0]$ was used. The Gaussian and the mean curvature functions are approximated by a bicubic B-spline surface with 256×256 control points and provided as input to the algorithm. The tolerances δ_K and δ_H of 0.0001 and 0.001 are used, respectively. After the optimization stops, the estimated scaling factor is 0.364, and the rotation matrix and the translation vector are summarized in Table 7. The matched surfaces are shown in Fig. 9A and the darker portion is surface B scaled and transformed. The relative measure of the maximum error can be calculated by dividing the maximum distance error by a square root of the surface area of the bottle, which is 0.0068.

The second example is an artificial surface shown in Fig. 10. The surface shown in Fig. 10A is a bicubic B-spline surface with 400 (20 × 20) control points enclosed in a box of 10 mm × 10 mm × 2.5 mm. The Gaussian and the mean curvature values are estimated at three selected seed points as shown in Fig. 10B. An interval of [0.3, 1.0] is used for an initial bracket for the optimization routine, and a value of 0.001 is used for the tolerance of the Golden section search. The recovered scale

Table 7
Estimated rigid body transformation for the first example

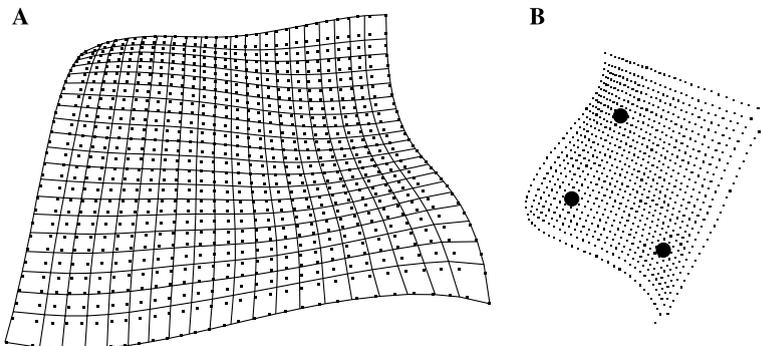| Rotation matrix	Translation vector
$\begin{bmatrix} 0.039 & 0.921 & 0.386 \\ -0.169 & 0.388 & -0.906 \\ -0.985 & -0.030 & 0.171 \end{bmatrix}$	$\begin{bmatrix} -32.090 & -53.595 & -12.391 \end{bmatrix}$

Fig. 10. Example 2 for the optimization method. (A) Matched surfaces. (B) Input points and three selected seed points for matching.

value is 0.652, and the relative maximum error is 0.002. The estimated rigid body transformation for this example can be found in Table 8.

The localized result is presented in Fig. 10A.

The third example is half of a fictitious automobile hood surface enclosed in a box of 13 mm × 12 mm × 6 mm. To imitate the behavior of a 3D scanner, the points **P** in Fig. 11B are disturbed by the following equation with $\xi = 0.01$ [31,39]:

$$\mathbf{P}' = \mathbf{P} + \xi \frac{(e_{ij}^x, e_{ij}^y, e_{ij}^z)^{\mathrm{T}}}{\sqrt{e_{ij}^{x2} + e_{ij}^{y2} + e_{ij}^{z2}}}, \tag{25}$$

where e_{ij}^x, e_{ij}^y, and e_{ij}^z $(i, j = 1, 2, 3)$ are randomly chosen numbers which vary from -1 to 1. Three seed points are selected, where the Gaussian and mean curvatures are evaluated as shown in Fig. 11B. An interval of [0.3, 1.0] is used for an initial bracket for the optimization routine, and a value of 0.01 is used for the tolerance of the Golden section search. The recovered scaling factor is 0.708, and the relative maximum error is 0.0005. The estimated rigid body transformation for this example can be found in Table 9 and the localized result is presented in Fig. 11A.

The elapsed time of the optimization method (excluding the surface fitting step) depends on the tolerances of the Golden section search and the IPP algorithm. The elapsed times for the examples under the given tolerances in this subsection are summarized in Table 10.

Table 8
Estimated rigid body transformation for the second example

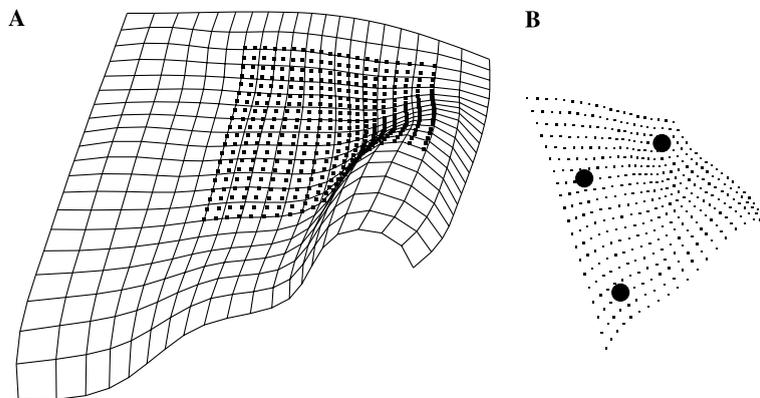| Rotation matrix			Translation vector
0.110	−0.906	−0.409	
−0.235	−0.423	0.875	[63.566 45.262 33.335]
−0.966	2.13×10^4	−0.259	

Fig. 11. A matching of a fictitious automobile hood surface. (A) Matched surfaces. (B) Input points and three selected seed points for matching.

Table 9
Estimated rigid body transformation for the third example

Rotation matrix	Translation vector
$\begin{bmatrix} 0.451 & 0.054 & -0.891 \\ -0.787 & 0.495 & -0.368 \\ 0.421 & 0.867 & 0.265 \end{bmatrix}$	$\begin{bmatrix} 4.28 & 11.463 & 0.137 \end{bmatrix}$

Table 10
Elapsed times for the examples

Examples	Times (s)
Bottle surface (Fig. 9)	3449
Test surface (Fig. 10)	1567
Automobile hood (Fig. 11)	907

8. Conclusions

We have addressed a problem of partial matching of free-form objects with scaling effects and no prior information on correspondence, and proposed two methods to solve it. With rough tolerances, the methods can be used to produce a good initial value for iterative algorithms such as the ICP algorithm and its variants. When tight tolerances and accurately estimated curvatures are used, the methods can yield accurate transformations. In the optimization method, an initial interval estimate of the scaling factor needs to be determined which includes the optimum value. However, unlike the ICP algorithm which requires an initial rigid body transformation as well as an initial scaling factor, our optimization method needs only an initial interval of the scaling factor because it incorporates the KH method which can handle a matching problem of NGWOS or NPWOS type. Therefore, estimation of an initial approximation becomes considerably simplified. In addition, the overall performance of the optimization method can be improved by combining the Golden section search and the secant method [16]. Near an optimum point where the size of the interval from the Golden section search is small, the secant method may be used to reduce execution time due to its faster convergence rate.

The proposed algorithms are well suited to copyright protection of CAD models represented by NURBS surface patches. Extension of the current algorithms to be used for various representation methods such as polyhedral and range data is recommended. In order to achieve this goal, estimation of differential properties from range data is important. In this paper, a least squares NURBS surface fitting method is used to obtain an approximated NURBS surface which is provided as input to the algorithms. However, this approximation itself requires further study. Since there is no general method which can cover all cases, a different method needs to be employed for the estimation of differential properties depending on the quality of input data. In conjunction with this estimation of differential properties, the overall effect

of noise in the input data on the matching result needs to be investigated. This analysis is important when real scanned data are provided as input.

Interval arithmetic is used for the formulation and solution of the systems of nonlinear equations in the calculation of umbilical points and the KH method using the IPP algorithm. After that, center values of each interval are calculated and floating point arithmetic is used in the subsequent calculation. All calculation for both proposed algorithms can be performed using interval arithmetic to enhance robustness. The method of classification of umbilical points and the Golden section search using interval arithmetic is a subject for future research topic.

References

[1] S.L. Abrams, W. Cho, C.-Y. Hu, T. Maekawa, N.M. Patrikalakis, E.C. Sherbrooke, X. Ye, Efficient and reliable methods for rounded-interval arithmetic, Comput. Aided Design 30 (8) (1998) 657–665.

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

[3] G. Barequet, S. Kumar, Repairing CAD models, in: R. Yagel, H. Hagen, (Eds.), Proceedings of IEEE Visualization Conference, Phoenix, Arizona, 1997, pp. 363–370.

[4] G. Barequet, M. Sharir, Partial surface matching by using directed footprints, Comput. Geometry 12 (1999) 45–62.

[5] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Trans. Pattern Anal. Mach. Intell. 24 (24) (2002) 509–522.

[6] R. Bergevin, D. Laurendeau, D. Poussart, Estimating the 3D rigid transformation between two range views of a complex object, in: Proceedings of 11th IAPR International Conference on Pattern Recognition. Conference A: Computer Vision and Applications The Hague, The Netherlands, vol. 1, IEEE Computer Society Press, Silver spring, MD, 1992, pp. 478–482.

[7] M.V. Berry, J.H. Hannay, Umbilic points on Gaussian random surfaces, J. Phys. A 10 (11) (1977) 1809–1821.

[8] P.J. Besl, The free-form surface matching problem, in: H. Freeman (Ed.), Machine Vision for Three-Dimensional Scenes, Academic Press, New York, 1990, pp. 25–71.

[9] P.J. Besl, N.D. McKay, A method for registration of 3D shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (2) (1992) 239–256.

[10] A. Bjorck, Numerical Methods for Least Squares Problems, Society for Industrial and Applied Mathematics, Philadelphia, 1996.

[11] J.W. Bruce, P.J. Giblin, F. Tari, Ridges, crests and sub-parabolic lines of evolving surfaces, Int. J. Comput. Vis. 18 (3) (1996) 195–210.

[12] R.J. Campbell, P.J. Flynn, A survey of free-form object representation and recognition techniques, Comput. Vis. Image Understand. 81 (2) (2001) 166–210.

[13] Y. Chen, G. Medioni, Object modeling by registration of multiple range images, in: The Proceedings of the 1991 IEEE International Conference on Robotics and Automation, Sacramento, California, April 1991, pp. 2724–2729.

[14] C.S. Chua, R. Jarvis, 3D free-form surface registration and object recognition, Int. J. Comput. Vis. 17 (1) (1996) 77–99.

[15] C.S. Chua, R. Jarvis, Point signatures: a new representation for 3D object recognition, Int. J. Comput. Vis. 25 (1) (1997) 63–85.

[16] G. Dahlquist, Å. Björck, Numerical Methods, Prentice-Hall, Englewood Cliffs, NJ, 1974.

[17] P.M. do Carmo, Differential Geometry of Curves and Surfaces, Prentice-Hall, Englewood Cliffs, NJ, 1976.

[18] C. Dorai, A.K. Jain, COSMOS-a representation scheme for 3D free-form objects, IEEE Trans. Pattern Anal. Mach. Intell. 19 (10) (1997) 1115–1130.

[19] G.H. Golub, C. Reinsch, Singular value decomposition and least squares solution, Numerische Mathematik 14 (1970) 403–420.

[20] B.K.P. Horn, Closed-form solution of absolute orientation using unit quaternions, J. Opt. Soc. Am. A 4 (4) (1987) 629–642.

[21] B.K.P. Horn, H.M. Hilden, S. Negahdaripour, Closed-form solution of absolute orientation using orthonormal matrices, J. Opt. Soc. Am. A 5 (7) (1988) 1127–1135.

[22] J. Hoschek, D. Lasser, Fundamentals of Computer Aided Geometric Design, A.K. Peters, Wellesley, MA, 1993. Translated by L.L. Schumaker.

[23] M. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inf. Theory IT-8 (1962) 179–187.

[24] R.A. Jinkerson, S.L. Abrams, L. Bardis, C. Chryssostomidis, A. Clement, N.M. Patrikalakis, F.-E. Wolter, Inspection and feature extraction of marine propellers, J. Ship Prod. 9 (2) (1993) 88–106.

[25] A.E. Johnson, M. Hebert, Recognizing objects by matching oriented points. in: CVPR'97, Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Juan, Puerto Rico, June 1997, pp. 684–689.

[26] A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3D scenes, IEEE Trans. Pattern Anal. Mach. Intell. 21 (5) (1999) 433–449.

[27] K.H. Ko, T. Maekawa, N.M. Patrikalakis, An algorithm for optimal free-form object matching, Comput. Aided Design 35 (10) (2003) 913–923.

[28] K.H. Ko, T. Maekawa, N.M. Patrikalakis, Shape intrinsic properties for free-form object matching, J. Comput. Inf. Sci. Eng. 3 (4) (2003) 325–333.

[29] K.H. Ko, T. Maekawa, N.M. Patrikalakis, H. Masuda, F.-E. Wolter, Shape intrinsic watermarks for free-form objects, in: Proceedings of the 2004 NSF Design, Service and Manufacturing Grantees and Research Conference, Dallas, Texas, January 2004.

[30] Y. Ma, J.P. Kruth, Parametrization of randomly measured points for least squares fitting of B-spline curves and surfaces, Comput. Aided Design 27 (9) (1995) 663–675.

[31] T. Maekawa, F.-E. Wolter, N.M. Patrikalakis, Umbilics and lines of curvature for shape interrogation, Comput. Aided Geometric Design 13 (2) (1996) 133–161.

[32] F. Mokhtarian, Silhouette-based isolated object recognition through curvature scale space, IEEE Trans. Pattern Anal. Mach. Intell. 17 (5) (1995) 539–544.

[33] R. Morris, The sub-parabolic lines of a surface, in: G. Mullineux (Ed.), The Mathematics of Surfaces VI, Proceedings of the 6th IMA Conference on Mathematics of Surfaces VI, Clarendon Press, Oxford, UK, 1996, pp. 79–102.

[34] M. Novotni, R. Klein, A geometric approach to 3D object comparison, in: International Conference on Shape Modeling and Applications, Shape Modeling International (SMI 2001), IEEE, Genova, Italy, 2001, pp. 167–175.

[35] A.V. Oppenheim, Applications of Digital Signal Processing, Prentice Hall, Englewood Cliffs, N.J, 1978.

[36] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Matching 3D models with shape distributions, in: International Conference on Shape Modeling and Applications, Shape Modeling International (SMI 2001), IEEE, Genova, Italy, 2001, pp. 154–166.

[37] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin, Shape distributions, ACM Trans. Graphics 21 (4) (2002) 807–832.

[38] N.M. Patrikalakis, L. Bardis, Localization of rational B-spline surfaces, Eng. Comput. 7 (4) (1991) 237–252.

[39] N.M. Patrikalakis, T. Maekawa, Shape Interrogation for Computer Aided Design and Manufacturing, Springer-Verlag, Heidelberg, 2002.

[40] L.A. Piegl, W. Tiller, The Nurbs Book, Springer, New York, 1995.

[41] I.R. Porteous, Geometric Differentiation for the Intelligence of Curves and Surfaces, Cambridge University Press, Cambridge, 1994.

[42] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C, Cambridge University Press, Cambridge, 1988.

[43] R.J. Prokop, A.P. Reeves, A survey of moment-based techniques for unoccluded object representation and recognition, Graph. Models Image Proc. 54 (2) (1992) 438–460.

[44] F.A. Sadjadi, E.L. Hall, Three-dimensional moment invariants, IEEE Trans. Pattern Anal. Mach. Intell. PAMI-2 (1980) 127–136.

[45] P.T. Sander and S.W. Zucker, Singularities of principal direction fields from 3-D images, in: IEEE Second International Conference on Computer Vision, Tampa Florida, 1988, pp. 666–670.

[46] P.T. Sander, S.W. Zucker, Inferring surface trace and differential structure from 3D images, IEEE Trans. Pattern Anal. Mach. Intell. 12 (9) (1990) 833–854.

[47] G.C. Sharp, S.W. Lee, D.K. Wehe, ICP registration using invariant features, IEEE Trans. Pattern Anal. Mach. Intell. 24 (1) (2002) 90–102.

[48] E.C. Sherbrooke, N.M. Patrikalakis, Computation of the solutions of nonlinear polynomial systems, Comput. Aided Geometric Design 10 (5) (1993) 379–405.

[49] S.S. Sinha, Differential properties from adaptive thin-plate splines, in: B.C. Vemuri (Ed.), Geometric Methods in Computer Vision, vol. 1570, SPIE, San Diego, CA, 1991, pp. 64–74.

[50] S.S. Sinha, P.J. Besl, Principal patches: a viewpoint-invariant surface description, in: IEEE International Robotics and Automation, Cincinnati, Ohio, May 1990, pp. 226–231.

[51] F. Stein, G. Medioni, Structural indexing: efficient 3-D object recognition, IEEE Trans. Pattern Anal. Mach. Intell. 14 (2) (1992) 125–145.

[52] D.J. Struik, Lectures on Classical Differential Geometry, Addison-Wesley, Cambridge, MA, 1950.

[53] M.W. Walker, L. Shao, R.A. Volz, Estimating 3-D location parameters using dual number quaternions, Comput. Vis. Graph. Image Proc. Image Understand. 54 (3) (1991) 358–367.

[54] Z. Zhang, Iterative point matching for registration of free-form curves and surfaces, Int. J. Comput. Vis. 13 (2) (1994) 119–152.

[55] J. Zhou, E.C. Sherbrooke, N.M. Patrikalakis, Computation of stationary points of distance functions, Eng. Comput. 9 (4) (1993) 231–246.