Shape Intrinsic Fingerprints for Free-Form Object Matching

K. H. Ko
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
khko@mit.edu

T. Maekawa
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
tmaekawa@mit.edu

N. M. Patrikalakis
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
nmp@mit.edu

H. Masuda
The University of Tokyo
Research into Artifacts, Center for Engineering
Tokyo 157-8904, Japan
masuda@race.u-tokyo.ac.jp

F.-E. Wolter
University of Hannover
Institute of Computer Science
D-30167 Hannover, Germany
few@informatik.uni-hannover.de

ABSTRACT

This paper presents matching and similarity evaluation methods between two NURBS surfaces, and their application to copyright protection of digital data representing solids or NURBS surfaces. Two methods are employed to match objects: the moment and the curvature methods. The moment method uses integral properties, i.e. the volume, the principal moments of inertia and directions, to find the rigid body transformation as well as the scaling factor. The curvature method is based on the Gaussian and the mean curvatures to establish correspondence between two objects. The matching algorithms are applied to problems of copyright protection. A suspect model is aligned to an original model through the matching methods so that similarity between two models can be assessed to determine if the suspect model contains part(s) of the original model, which may be stored in an independent repository. Three types of tests, the weak, intermediate and strong tests, are proposed for similarity assessment between two objects. The weak and intermediate tests are performed at node points obtained through shape intrinsic wireframing. The strong test relies on isolated umbilical points which can be used as fingerprints of an object for supporting an ownership claim to the original model. The three tests are organized in two decision algorithms such that they produce systematic and statistical measures for a similarity decision between two objects in a hierarchical manner. Based on the systematic and statistical evaluation of similarity, a decision can be reached whether the suspect model is an illegal copy of the original model.

SM'03, June 16–20, 2003, Seattle, Washington, USA.

Categories and Subject Descriptors

I.5.3 [**PATTERN RECOGNITION**]: Similarity measures; J.6 [**COMPUTER APPLICATIONS**]: Computer-Aided Engineering—*Computer-aided design (CAD)*; K.5.1 [**LEGAL ASPECTS OF COMPUTING**]: Hardware/Software Protection—*Copyrights*

General Terms

Algorithms

Keywords

NURBS, umbilics, matching, registration, localization, partial matching, similarity, fingerprints

1. INTRODUCTION

Free-form is a term used to describe an object which cannot be represented with simple surfaces such as planes, the natural quadrics and tori [6]. Using *Non-Uniform Rational B-Spline (NURBS)* surface patches, free-form shapes of great complexity can be represented in digital form, which are used to design many 3D objects such as ship hulls, automobile and aircraft bodies, turbines, propellers, and consumer products.

In the past, these 3D model descriptions had been represented with a fairly restrictive shape variety, for example, 2D drawings (blueprints). Today they are typically described with CAD systems in digital form. Here the richest shape variety can be modeled by free-form surfaces that are typically defined as NURBS. Hence, the most important and fundamental part of the value creation process for a 3D model consists in creating the digital 3D free-form model. As this digital data model is expensive and important part of the whole production process there exists the natural need to protect its ownership [16]. Two types of feasible protection methods can be considered: one is to embed *watermarks* in the object and check them for illegal duplication. The other method is to align two objects as accurately as possible and check them for similarity.

Several methods have been reported on digital watermarking for 3D polygonal models which are widely used for virtual reality and computer graphics. Most of the methods are designed for triangular meshes and embed watermark information by perturbing geometry or changing topological connectivity [22, 1] or using the frequency domain of 3D models [14, 26, 33]. Watermarking on Constructive Solid Geometry (CSG) models was developed by Fornaro and Sanna [10]. Despite the popularity of NURBS curves and surfaces, watermarking for the NURBS representation is relatively new in the CAD field. Ohbuchi *et al.* [23] proposed a new data embedding algorithm for NURBS curves and surfaces, which are reparameterized using rational linear functions whose coefficients are manipulated for encoding data. The watermark information, however, can be easily destroyed by reparameterization or reapproximation of the curves and surfaces. Since it is difficult to insert any form of robust user-defined watermarks in the NURBS representation, the similarity checking method is adopted in this work. Apart from a rigid body motion, the definition of every surface or solid provides the shape identity of the object. This shape identity might be called the *shape intrinsic fingerprint*, as it is not artificial but is related merely to the definition of the object's shape. Umbilics are a particular type of these shape intrinsic fingerprints. Therefore, the fundamental issue involved here is to compare two objects (solids or NURBS surface patches) using the shape intrinsic fingerprints and determine whether one object is a copy of the other. Since such fingerprints are intrinsic features of an object, extracting these fingerprints does not disturb the object as opposed to watermarking which embeds additional information resulting in some modification of the object. Digital fingerprints have been widely used for 2D images and multimedia data. Recognition, digital indexing and tracking of images or multimedia data [13, 9] are main applications of the digital fingerprints. Some researchers refer to digital fingerprints as *passive watermarks*. They can also be used as an alternative to digital watermarks. However, digital fingerprints for 3D CAD models have not been discussed so far in the extant literature.

Before comparing two free-form objects, they have to be aligned so that the effects of a rigid body transformation (rotation and translation) and possibly scaling are minimized. This alignment problem is equivalent to *localization* or *registration*. 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*.

The moment based approach is a simple and useful method and has been extensively employed for aligning two objects, especially two solids [28]. Iterative optimization approaches are widely used in matching problems [4, 32, 24]. An optimal transformation can be found by minimizing a mean square distance metric objective function which involves six degrees of freedom for matching (three for translation and three for rotation). A matching problem with no prior clue for correspondence or transformation belongs to another category of matching discussed in the literature. Bergevin *et al.* [2] used a hierarchical surface triangulation representation to estimate the 3D rigid body transformation. A matching method for two range data sets using the principal curvatures and Darboux frames was proposed in [7]. Most of the matching approaches developed so far, however, deal with a case of sets of points vs. a surface. When two NURBS surface patches are given, points are sampled from one of the surfaces, and then any of the matching methods can be employed. But, in this case, no assurance can be made that the best transformation, i.e. the global optimum, has been found, as some of the surface information is lost during the digitization of the surface. Therefore, accurate alignment cannot always be expected.

In this paper, we review matching methods using the principal moments and surface intrinsic properties. Similarity checking algorithms are proposed along with a new detection method for umbilical points and a surface intrinsic wireframing method. We demonstrate the proposed algorithms for copyright protection of objects in NURBS form.

This paper is structured as follows: Mathematical concepts from differential geometry are briefly reviewed in Section 2 and a method to extract umbilical points is presented with a few examples in Section 3. An algorithm to create a surface intrinsic wireframe for a given NURBS surface is described in Section 4 and two matching methods are presented in Section 5. Section 6 describes three tests for matching and assessment of similarity, followed by two similarity decision algorithms in Section 7. Several examples demonstrating the proposed matching methods and similarity decision algorithms are presented in Section 8. Finally Section 9 concludes the paper, including suggestions for future work.

2. REVIEW OF DIFFERENTIAL GEOMETRY

Suppose we have a regular parametric surface

$$\mathbf{r}(u,v) = [x(u,v), y(u,v), z(u,v)]^T . \tag{1}$$

From the theory of differential geometry on surfaces, the first (I) and the second (II) fundamental forms [31] are defined by

$$I = d\mathbf{r}\cdot d\mathbf{r} = Edu^2 + 2Fdudv + Gdv^2, \tag{2}$$

$$II = -d\mathbf{r}\cdot d\mathbf{N} = Ldu^2 + 2Mdudv + Ndv^2, \tag{3}$$

where $\mathbf{N}$ 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. The *Gaussian (K)* and the *mean (H) curvatures* of (1) are given by

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

$$H = \frac{1}{2}\left(\frac{2FM - EN - GL}{EG - F^2}\right). \tag{4}$$

Lines of Curvature

A *line of curvature* is a curve on a surface whose tangents are the principal directions at all its points. At a point on a surface away from umbilical points, two principal directions are uniquely determined and are orthogonal [31].

Employing the arc length parameter s, i.e. $u = u(s)$ and $v = v(s)$, we obtain the ordinary differential equations for lines of curvature as follows [31]:

$$\frac{du}{ds} = \eta(M + \kappa F), \quad \frac{dv}{ds} = -\eta(L + \kappa E), \tag{5}$$

or

$$\frac{du}{ds} = \mu(N + \kappa G), \quad \frac{dv}{ds} = -\mu(M + \kappa F), \tag{6}$$

where κ is the principal curvature, η and μ are factors determined through the normalization condition using the first fundamental form [21, 25]. Depending on the size of the coefficients, $(L + \kappa E)$ and $(N + \kappa G)$, either (5) or (6) are selectively used. Namely, if $|(L + \kappa E)| \leq |(N + \kappa G)|$, we solve (5). Otherwise, solve (6). See [21, 25] for details.

Geodesics

Let us define a unit vector $\mathbf{u} = \mathbf{N} \times \mathbf{t}$ at a point $\mathbf{p}$ on a surface, where $\mathbf{t}$ is the unit tangent vector of a curve $\mathbf{c}$ on the surface at $\mathbf{p}$. Then $\mathbf{u}$ is perpendicular to $\mathbf{N}$ and $\mathbf{t}$, and is contained in the tangent plane of the surface at $\mathbf{p}$. The $\mathbf{u}$ component of the curvature vector $\boldsymbol{\kappa}$ of $\mathbf{c}$, which is obtained by

$$\boldsymbol{\kappa}_g = (\boldsymbol{\kappa} \cdot \mathbf{u})\,\mathbf{u}, \tag{7}$$

is called the *geodesic curvature vector*, and the magnitude of $\boldsymbol{\kappa}_g$ is the *geodesic curvature* in the direction of $\mathbf{t}$ at $\mathbf{p}$ [31]. Considering that the surface normal $\mathbf{N}$ has the direction of a normal to the geodesic curve $\pm\mathbf{n}$, a set of differential equations for geodesic curves can be derived from two equations $\mathbf{n} \cdot \mathbf{r}_u = 0$ and $\mathbf{n} \cdot \mathbf{r}_v = 0$, using the Christoffel symbols Γ^i_{jk}, $(i, j, k = 1, 2)$ [31] as follows:

$$\frac{du}{ds} = p, \tag{8}$$

$$\frac{dv}{ds} = q, \tag{9}$$

$$\frac{dp}{ds} = -\Gamma^1_{11}p^2 - 2\Gamma^1_{12}pq - \Gamma^1_{22}q^2, \tag{10}$$

$$\frac{dq}{ds} = -\Gamma^2_{11}p^2 - 2\Gamma^2_{12}pq - \Gamma^2_{22}q^2. \tag{11}$$

Umbilics

An *umbilic* is a point on a surface where the normal curvatures in all directions are equal and the principal directions are indeterminate. The principal curvature functions are represented in terms of the Gaussian (K) and the mean (H) curvature functions as follows [31]:

$$\kappa_{1,2}(u, v) = H(u, v) \pm \sqrt{H^2(u, v) - K(u, v)}. \tag{12}$$

Let $W(u, v) = H^2 - K$. The principal curvatures, $\kappa_{1,2}$ are real valued functions so that $W \geq 0$ must hold. From the definition of the 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 [19, 21]. 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 umbilic the following equations hold [21]:

$$W(u, v) = 0, \quad \frac{\partial W(u, v)}{\partial u} = 0, \quad \frac{\partial W(u, v)}{\partial v} = 0. \tag{13}$$

Given a polynomial parametric surface patch such as a rational Bézier surface patch, we can set $W = \frac{P_N}{P_D}$, where P_N and P_D are polynomials in u and v. With the condition $W \geq 0$, $P_N \geq 0$ is assured since $P_D > 0$ is always true under the regularity condition of the surface [31]. The equation $W = 0$ is equivalent to $P_N = 0$. The first derivative of W is $\frac{\partial W}{\partial x_i} = (\frac{\partial P_N}{\partial x_i} P_D - P_N \frac{\partial P_D}{\partial x_i})/P_D^2 (i = 1, 2)$, where $x_1 = u$ and $x_2 = v$, which is reduced to $\frac{\partial W}{\partial x_i} = (\frac{\partial P_N}{\partial x_i})/P_D$ using $P_N = 0$. Therefore, equations (13) are reduced to [21]

$$P_N(u, v) = 0, \quad \frac{\partial P_N}{\partial u} = 0, \quad \frac{\partial P_N}{\partial v} = 0. \tag{14}$$

Umbilical points can be isolated or form lines or regions. They are classified into two types based on the stability with respect to small perturbations: *generic* and *non-generic*. Generic umbilical points are stable with respect to small perturbations. Isolated generic umbilical points are further categorized into three types: *lemon*, *star* and *monstar* as shown in Figure 1 [3]. The type of the isolated generic umbilical points can be determined by the *index* and the number of lines of curvature passing through the umbilical point. The index is the amount of rotation that a straight line tangent to lines of curvature experiences when rotating in the counterclockwise direction along a small closed path around the umbilic [21]. The index can distinguish the star type umbilical point from the monstar or lemon type umbilical point, and the number of lines of curvature passing through the umbilical point can discriminate between the lemon type, and the monstar or star type umbilical points. One line of curvature passes for the lemon type umbilical point whereas three pass through the umbilic of monstar and star types. The criterion distinguishing monstar from star type is that all three directions of lines of curvature through a monstar umbilic are contained in a right angle, whereas in the star type umbilical point case, they are not enclosed in a right angle [21]. For details, see [3, 21, 25].

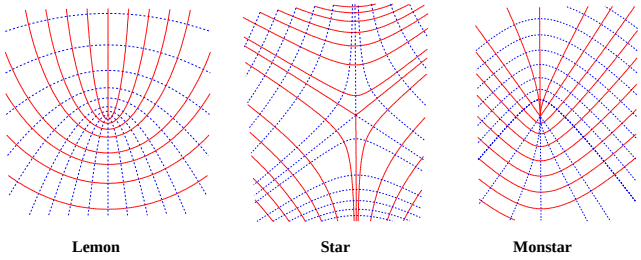

Figure 1: Three generic umbilics

3. EXTRACTION OF UMBILICS

Umbilical points appear in various forms on a free-form surface such as isolated points, lines or regions. Because of the singular behavior of the principal direction field around the umbilics, information on their locations is important in surface-intrinsic-wireframing (see Section 4), and in selection of reference points for matching. Moreover, isolated generic umbilical points need to be located and classified for the strong test of decision algorithms for similarity (see Section 7). Therefore, a robust and efficient algorithm for extraction of umbilical points and regions is necessary, and is presented below.

The quadtree decomposition is used to extract umbilical points or regions from a free-form NURBS surface. Such decomposition has been used for various purposes such as intersection problems [8] and computer graphics [29]. Combining the *convex hull property* of the Bernstein polynomials, the quadtree decomposition provides an efficient method for extraction of umbilical points for the free-form NURBS surface, especially when the umbilics are not isolated.

A NURBS surface is first subdivided into rational Bézier surface patches by knot insertion and the governing equations (13) are formulated for each resulting rational Bézier surface patch, which are reduced to equations (14). The graph of $z = P_N(u, v)$ $(0 \le u, v \le 1)$ is represented over the uv parametric plane. The condition $P_N(u, v) \ge 0$ assures that no portion of P_N lies below the uv plane, i.e. P_N has no negative value. Suppose that $\mathbf{P}_N = (u, v, P_N)$. Given an integral Bézier surface patch of degree m and n in u and v, $\mathbf{P}_N(u, v)$ in the bivariate Bernstein form is given by [21]:

$$\mathbf{P}_N(u,v) = \sum_{i=0}^{10m-6} \sum_{j=0}^{10n-6} \mathbf{p}_{ij} B_{i,10m-6}(u) B_{j,10n-6}(v), \quad (15)$$

and for a rational Bézier surface patch of degree m and n in u and v, $\mathbf{P}_N$ is given by [21]:

$$\mathbf{P}_N(u,v) = \sum_{i=0}^{24m-6} \sum_{j=0}^{24n-6} \mathbf{p}_{ij} B_{i,24m-6}(u) B_{j,24n-6}(v), \quad (16)$$

where $\mathbf{p}_{ij}$ are Bernstein coefficients. Then the detection problem is reduced to find a set of (u, v) which satisfy $P_N(u, v) = 0$, or the zero-set of a bivariate Bernstein polynomial (15) or (16). An adaptive quadtree decomposition on the uv domain is used to narrow down regions possibly containing umbilical points. A rectangular domain is subdivided into four rectangular domains at the mid points of u and v using the de Casteljau algorithm. This algorithm can be robustly executed in rounded interval arithmetic. At a depth d, there are at most 4^d nodes and each node has a domain of size $r2^{-d} \le u \le (r+1)2^{-d}$ and $s2^{-d} \le v \le (s+1)2^{-d}$ where r and s are integers of $0 \le r \le 4^{d-1}$ and $0 \le s \le 4^{d-1}$, respectively. Figure 2 shows an example of the quadtree decomposition. The depth of the quadtree depends on the user-specified size of subdivided regions.

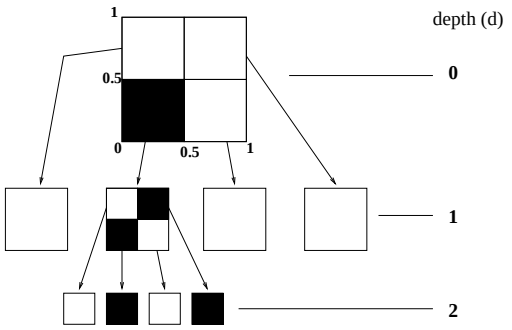

Figure 2: An example of the adaptive quadtree decomposition (The marked dark domains indicate those which possibly contain umbilical points.)

The strategy of the adaptive quadtree decomposition algorithm for extraction of umbilical points is to eliminate the regions which do not include any umbilical points using the convex hull property. The convex hull property determines if a subdivided region does not contain roots of $P_N(u, v) = 0$ or not. This property assures that if all Bernstein coefficients in a subdivided region are positive, then the Bézier surface patch in that region should lie above the uv plane, which means that no umbilical point exists in the region. Note that the reverse is not necessarily true. The algorithm checks the signs of the Bernstein coefficients in a subdivided region. If all of them are positive, the node of the corresponding region is marked as a non-umbilic region. The algorithm stops at a depth d_f where the size of the regions of the nodes at d_f is less than the user defined tolerance. We traverse the tree and collect the unmarked nodes to produce the regions which possibly contain umbilical points.

It is not known a priori how many nodes will be generated in general. But because we can estimate the maximum depth required to achieve a given accuracy, we can perform the worst case analysis. Given a tolerance δ_{umb}, the algorithm stops when the size of a subdivided domain at a depth d, or 2^{-d} is less than δ_{umb}. Let us assume that $2^{-d} < \delta_{umb}$ and the algorithm has stopped. Subdivision happens twice at every node up to depth $d-1$. Therefore, the total number of applications of the de Casteljau algorithm becomes $2\sum_{j=0}^{d-1} 4^j$. For a Bézier surface patch of degree m and n in the u and v, the application of the de Casteljau algorithm in u parameter requires $O(m^2)$. Hence, the complexity of the worst case is reduced to $O(4^d(m^2+n^2))$.

The IPP algorithm [21, 25, 30] can also be used for extraction of umbilics. But when the IPP algorithm encounters regions of umbilical points, it slows down dramatically. The quadtree decomposition method is efficient in dealing with lines or regions of umbilical points because it employs an adaptive subdivision scheme and only requires subdivision using the de Casteljau algorithm, whereas the IPP algorithm needs not only subdivision but also calculation of convex hulls and projection onto a hyperplane. However, since it belongs to the subdivision class of methods, the proposed method is unable to differentiate multiple roots [30, 25].

Examples

An integral bi-cubic Bézier surface patch [21] is used for extraction of isolated umbilical points as in Figure 3. The patch contains five isolated umbilical points. Table 1 summarizes the exact parametric values of each umbilical point adapted from [21] and the boxes from the subdivision extraction algorithm. The calculation is performed with a tolerance of $\delta_{umb} = 0.005$. Table 1 shows that the estimated

No.	Exact Location (u, v)	Estimation $[u_a, v_a] \times [u_b, v_b]$
1	$(0.211, 0.052)$	$[0.2109, 0.0508] \times [0.2148, 0.0547]$
2	$(0.211, 0.984)$	$[0.2109, 0.9807] \times [0.2148, 0.9844]$
3	$(0.789, 0.052)$	$[0.7852, 0.0508] \times [0.7891, 0.0547]$
4	$(0.789, 0.984)$	$[0.7852, 0.9807] \times [0.7891, 0.9844]$
5	$(0.500, 0.440)$	$[0.4961, 0.4375] \times [0.5039, 0.4414]$

Table 1: Comparison of positions of isolated umbilical points

boxes contain the exact location of the umbilical points computed via the IPP algorithm solving (14) at a higher precision of 10^{-12}. The next example shows a line of umbilical points on a surface as in Figure 4. The input surface is a developable cubic-linear surface adapted from [18]. The surface has an inflection line at $u = 0.5754$ computed via the IPP algorithm with a precision of 10^{-12}. The proposed algorithm with a tolerance $\delta_{umb} = 0.005$ produces a series

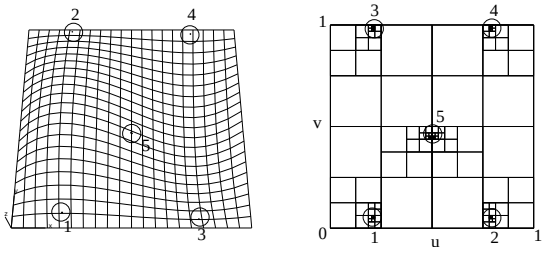

Figure 3: An example of isolated umbilical points on *uv* domain and the surface

of boxes which contain 0.5754 as follows:

$$[0.5742, 0.0] \times [0.5781, 0.0039] ,$$
$$[0.5742, 0.0039] \times [0.5781, 0.0078] ,$$
$$\vdots$$
$$[0.5742, 0.9961] \times [0.5781, 1] .$$

Those boxes are mapped onto the surface in a series of small boxes as shown in Figure 4.

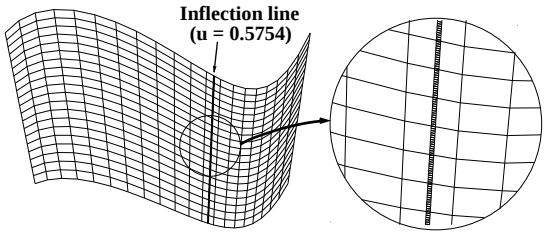

Figure 4: An example of a line of umbilical points

An example of extraction of a planar region is presented in Figure 5. It is a bicubic B-spline surface which is partially planar, with 20×20 control points in the u and v parametric directions. The extracted planar region of the surface is shown in Figure 5, and the tolerance used is $\delta_{umb} = 0.05$.

4. INTRINSIC WIREFRAMING

In this section, a method for surface-intrinsic-wireframing using lines of curvature and geodesic curves is explained, which will not be affected by parametrization, any rigid body transformation and representation methods. From this wireframe, reference points are selected for comparison and surface matching.

4.1 Overall Structure

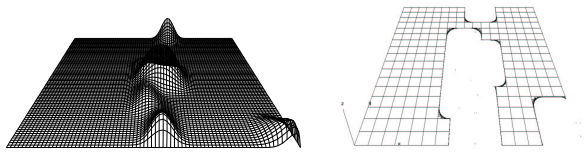

Figure 5: Extraction of planar region

An algorithm for the construction of intrinsic wireframing is shown in Figure 6. A NURBS surface is provided as input. As the principal directions are indeterminate at umbilical points, we need to locate such points prior to wireframing. Next a starting point is selected, from which wireframe creation begins. It continues until lines of curvature cannot be traced any longer. The remaining area is covered with geodesic curves. If the entire surface has been wireframed sufficiently densely according to appropriate user-specified thresholds, the algorithm stops.

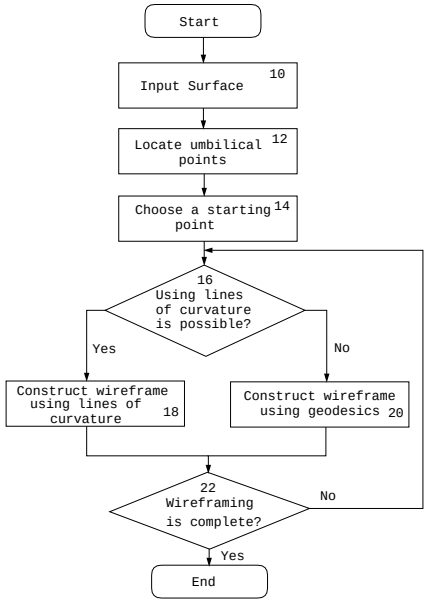

Figure 6: A diagram of the algorithm

Input Surfaces and Umbilics (Steps 10 and 12)

Information on exact locations of umbilical points is essential for wireframing using lines of curvature because the principal directions cannot be uniquely defined there so that the lines of curvature cannot be traced properly. Therefore, surfaces need to be provided as input to the algorithm with complete information of isolated umbilical points, spherical or planar regions, or lines of umbilical points extracted by using the quadtree decomposition method of Section 3.

Starting Points (Step 14)

Either a star type umbilical point or a non-umbilical point can be chosen as a starting point for wireframing because the maximum and minimum lines of curvature radiate from the point in an alternating pattern so that a simple algorithm is sufficient and the resulting mesh is more well-proportioned. The other two types of umbilics, i.e. lemon and monstar, are not appropriate for this purpose. Three lines of curvature pass through a star umbilical point, each of which changes its attribute from the maximum line of curvature to the minimum line of curvature or *vice versa*. Therefore, at the star type umbilical point, we can imagine that six lines of curvature radiate from the umbilical point so that we can use up to six initial directions for tracing lines of curvature. When there is no umbilical point on the surface, a non-umbilical point is chosen as a starting point. The non-umbilical point

has the maximum and minimum lines of curvature intersecting orthogonally. Thus, we can choose up to four directions for tracing the lines of curvature.

Wireframing with Lines of Curvature (Step 18)

Lines of curvature are calculated by solving equations (5) or (6) using for example the fourth order Runge-Kutta method.

An intersection point of the maximum and minimum lines of curvature can be calculated accurately using Newton's method. Let us assume that a maximum line of curvature L_1 and a minimum line of curvature L_2 intersect as shown in Figure 7. Using the arc length s_1 as a parameter, L_1 can be represented as

$$u_1 = u_1(s_1), \quad v_1 = v_1(s_1). \tag{17}$$

Similarly, L_2 is

$$u_2 = u_2(s_2), \quad v_2 = v_2(s_2), \tag{18}$$

where s_2 is the arc length parameter. The problem can be stated to find s_1 and s_2 so that

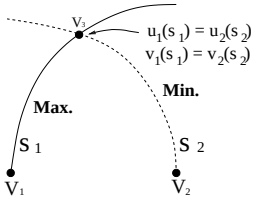

Figure 7: Intersection of lines of curvature

$$\mathbf{F}(\mathbf{x}) = \begin{pmatrix} f(s_1, s_2) \\ g(s_1, s_2) \end{pmatrix} = \begin{pmatrix} u_1(s_1) - u_2(s_2) \\ v_1(s_1) - v_2(s_2) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{19}$$

where $\mathbf{x} = (s_1, s_2)^T$. The multivariate Newton's method can be adopted as a solution method. Equation (19) is Taylor expanded to obtain $\mathbf{F}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{F}(\mathbf{x}) + \mathbf{J} \cdot \delta\mathbf{x}$, where $\delta\mathbf{x} = (\delta_1, \delta_2)^T$ and $\mathbf{J}$ is the Jacobian matrix of $\mathbf{F}$ which can be calculated numerically using the equations (5) or (6). Two tolerances δ_1 and δ_2 for the variables s_1 and s_2 are provided for termination of the iteration of Newton's method.

Geodesic Wireframing (Step 20)

In a region where the algorithm using lines of curvature fails, i.e. in the neighborhood of an umbilical point (except the umbilical point used as a starting point) and near a boundary or in an umbilical region, a geodesic curve can be used to complete wireframing. A geodesic problem comes in the form of a boundary value or an initial value problem. However, the problem arising in wireframing is always formulated as a boundary value problem (BVP) and a solution method for BVPs is adopted. Two points are selected and a geodesic line is calculated which connects them. As an initial approximation, a straight line connecting two boundary points is used, from which a solution is obtained iteratively by using the *relaxation method*, see [25, 27, 17].

4.2 Methods for Constructing Quadrilateral Meshes

The proposed method is semi-automatic so that in order to complete a wireframe of a free-form surface the user needs to work interactively. An automatic procedure handles wireframing using lines of curvature. After a starting point has been selected, the routine constructs quadrilateral meshes (mostly orthogonal meshes) radiating from the starting point. An outline of this procedure follows:

1. Advance L_1 and L_2 from V_0 by a user defined distance to reach V_1 and V_2 points as shown in Figure 8. Here, we assume that L_1 and L_2 are the maximum and the minimum lines of curvature, respectively.
2. Choose the minimum line of curvature (L_1') at V_1 and the maximum lines of curvature (L_2') at V_2.
3. Find an intersection point V_3 of L_1' and L_2'.

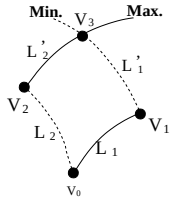

Figure 8: A diagram for meshing algorithm

This procedure continues until no more intersection point can be obtained, i.e. when lines of curvature hit either the boundary or umbilical points. In a region that lines of curvature cannot cover, the user inserts additional nodes and connects them using geodesic curves. Figure 9 shows an example of intrinsic wireframing. This example has 3 star umbilics and consists of 413 nodes, 356 quadrilateral and 14 triangular elements. A computer program was developed to

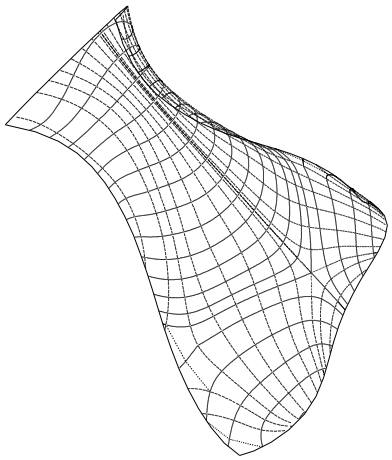

Figure 9: Intrinsic wireframe

integrate all functions that are used to generate an intrinsic wireframe of a NURBS surface. It consists of two windows. One is to visualize a surface and wireframe in 3D space and the other, which is denoted as the control window, is to provide an interface for a user to interact with the wireframe when user input is needed. The visualization window uses OpenGL. It provides functions for rotation and scaling for better visualization. The control window shows the current status of the parametric domain during wireframing. Several options are provided for interactive operations. The

language used in the development is C++. A set of window managing libraries, Qt-2.3.1, are used for the GUI.

5. MATCHING METHODS

Matching is a process determining a rigid body motion (translation and rotation) which makes two objects match as closely as possible [12]. Two methods for matching are used. Using the integral properties such as the moments of inertia, volume and principal moments of inertia, a rigid body motion can be found to align two objects. The other approach is to use intrinsic properties on the surface of objects. This method, denoted as the *KH method* can be applied not only to global matching but also to partial matching.

5.1 Moment Method

Matching via integral properties is used for solids. The integral properties for solids A and B, i.e. centroids (centers of volume) and moments of inertia assuming uniform density equal to one, are calculated using Gauss's theorem or the divergence theorem which reduces volume integrals to surface integrals. The inertia tensors of solids A and B are constructed. The inertia tensor is a 3×3 square symmetric matrix, whose diagonal terms I_{xx}, I_{yy} and I_{zz} are called the moments of inertia and the remaining three terms are called products of inertia. Principal moments of inertia and their directions are obtained via an eigenvalue problem [28].

Once the centroids and principal directions of both solids are calculated, solids A and B are translated and rotated so that their centroids and principal axes of inertia coincide. If necessary, solid B is uniformly scaled based on the relative volumes of the two solids.

There are cases where ambiguity in matching arises when the principal moments of inertia are used for matching. Since for each principal axis, two opposite directions are possible so that in total eight matching cases can be obtained. A right-handed coordinate frame assumption can reduce the eight matching cases to four [11]. Such ambiguity in matching the directions of the principal moments of inertia can be resolved by evaluating the sum of the squared distances calculated at reference points for comparison and choosing one direction which yields the minimum value.

5.2 KH Method

The KH method [15] establishes correspondence between two surfaces using the surface intrinsic properties and finds a rigid body transformation to match the two surfaces as closely as possible. Three points are selected on surface B and three pairs of the Gaussian (K) and the mean (H) curvatures are evaluated at these three points. Then the corresponding points on A which have the same curvature values as those on surface B are located by solving a system of equations for the Gaussian and the mean curvatures on A using the *Interval Projected Polyhedron* algorithm [30, 25]. After the correspondence has been established, a rigid body motion is calculated. This method relies only on the surface. Therefore, it can be applied to solids bounded by NURBS surfaces. Moreover, it can be used for partial surface localization.

6. SIMILARITY EVALUATION

Suppose we have two surfaces A and B. For solids, the bounding surfaces are considered. A matching method is used to align B with A. Then, the matched surfaces are compared for similarity with the following tests.

6.1 Tests of Matching

The ϵ-offset and the principal curvature tests are performed at the node points obtained from the surface intrinsic wireframe.

ϵ-Offset *Test (Weak Test)*

The objective of this test is to determine how close B is to A in terms of the Euclidean distance. The squared minimum distances between A and B are calculated and checked if all of them are within an ϵ-distance bound or one of the surfaces is within an **ϵ-offset** of the other. In [34], the squared distance function and its stationary points between two variable points located on two different geometric entities are investigated. Based on this technique, the distances are calculated between two surfaces A and B.

Principal Curvature Test (Intermediate Test)

The principal curvatures and their directions are used in this similarity test. The differences of the principal curvatures and directions between two surfaces are calculated and used for a similarity decision.

Umbilic Test (Strong Test)

Every closed orientable surface (of differentiability class C^3) with a genus different from one (hence being topologically different from the torus) has at least one umbilic [5] and various free form surfaces may contain umbilical points. However, the availability of this test depends on the existence of isolated generic umbilics. This test is based on the fact that generic isolated umbilics and the patterns of lines of curvature around them are stable to perturbations so that their qualitative properties are preserved and checks whether their locations and patterns for surface A match those for surface B.

6.2 Assessment of Matching

Let us denote k node points of the surface intrinsic wireframe from surface B as $\mathbf{P}_i$ $(i = 1, 2, \cdots, k)$. Here surface intrinsic sampling methods using geodesics or lines of curvature are preferred because they are independent of parametrization. Next, find the minimum distance footpoints $\mathbf{Q}_i$ on surface A of $\mathbf{P}_i$. The IPP algorithm can be used to find these minimum distance footpoints robustly as in [34]. After finding the footpoints $\mathbf{Q}_i$ on A, calculate the following quantities between $\mathbf{P}_i$ and $\mathbf{Q}_i$ $(i = 1, 2, \cdots, k)$.

- Euclidean distance of $|\mathbf{P}_i - \mathbf{Q}_i|$: ϵ_{0i}
- The second derivative properties
  - Difference of principal curvatures : $\epsilon_{1i}, \epsilon'_{1i}$
  - Difference of principal directions : ϵ_{2i}

Maximum values, average values and standard deviations can be calculated for each ϵ_{0i}, ϵ_{1i}, ϵ'_{1i} and ϵ_{2i} to provide quantitative statistical measures to determine how similar the two surfaces are in a global manner.

Local similarity can be assessed with ϵ_{ji} at corresponding positions. Each ϵ_{ji} $(j = 0, 1, 2)$ is normalized with respect to the maximum value of $max_i(\epsilon_{ji})$. Tolerances δ_j $(j = 0, 1, 2)$, corresponding to ϵ_{ji}, are used to extract the regions

of interest. Namely, the regions in which $\epsilon_{ji} > \delta_j$ are those where the two surfaces are different. As an extension of this idea, the similarity between two surfaces can be provided as a percentage value. First, the difference values ϵ_{ji} are located over the uv plane. Then the uv plane is subdivided into a set of square grids of size $(\delta_s \times \delta_s)$ where δ_s is a user defined value. The total number of the square meshes is denoted as D_T. Given a tolerance δ_j, the number of the squares D_ϵ, which contain at least one point satisfying $\epsilon_{ji} > \delta_j$, is found. Then, $\left(1 - \frac{D_\epsilon}{D_T}\right) \times 100$ becomes a percentage value of similarity. The squares which do not contain points satisfying the condition indicate the regions where the two surfaces are equivalent under a given test with a tolerance δ_j.

7. DECISION ALGORITHMS

Two similarity decision algorithms are proposed in this section. They consist of three tests as described in Section 6.1 and provide quantitative results with which one can determine whether one surface is a copy of another surface or not. Algorithm 1 uses the maximum deviation value at each test for a decision, while Algorithm 2 employs statistical methods for a decision. Each algorithm produces hierarchical results for similarity between two surfaces. In the subsequent sub-sections, it is assumed that surfaces A and B are matched, wireframed and all umbilical points are detected.

7.1 Algorithm 1

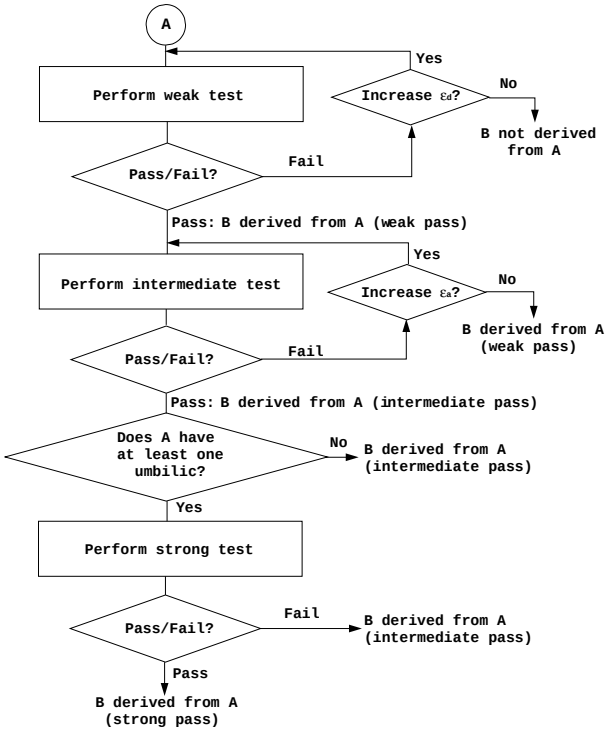

Figure 10: Algorithm 1

Two surfaces are provided as input to the first test or a weak test (ϵ-offset test) as shown in Figure 10. Then a decision is made that surfaces A and B are within or out of tolerance ϵ_d based on the Euclidean distance. If the maximum distance between corresponding points on surfaces A and B is within ϵ_d, then surface B is considered to have passed the weak test and determined to be a copy of A under the weak test. On the other hand, if the distance is greater than tolerance ϵ_d, the test fails. In such case, there are two possible courses of action. If ϵ_d is not large with respect to the size of surfaces, the user may decide to increase it and retry the weak test. If ϵ_d is large, then the user may decide to stop the process and decide that B is not derived from A.

If the weak test is passed, then an intermediate test (principal curvature and direction test) may be performed. The procedure is similar to that of the weak test. If the test succeeds, B is considered to be a copy of A under the intermediate test. If it fails, the user may decide to stop the process and conclude that B is derived from A with respect only to the weak test or try the test again with a new ϵ_a.

If no isolated generic umbilical point exist, the process stops and it is concluded that B is derived from A under the intermediate test. If an umbilic exists, the strong test (umbilic test) may be performed. If the test succeeds, it is concluded that B is derived from A under the strong test. Otherwise, it is decided that B is a copy of A under the intermediate test.

7.2 Algorithm 2

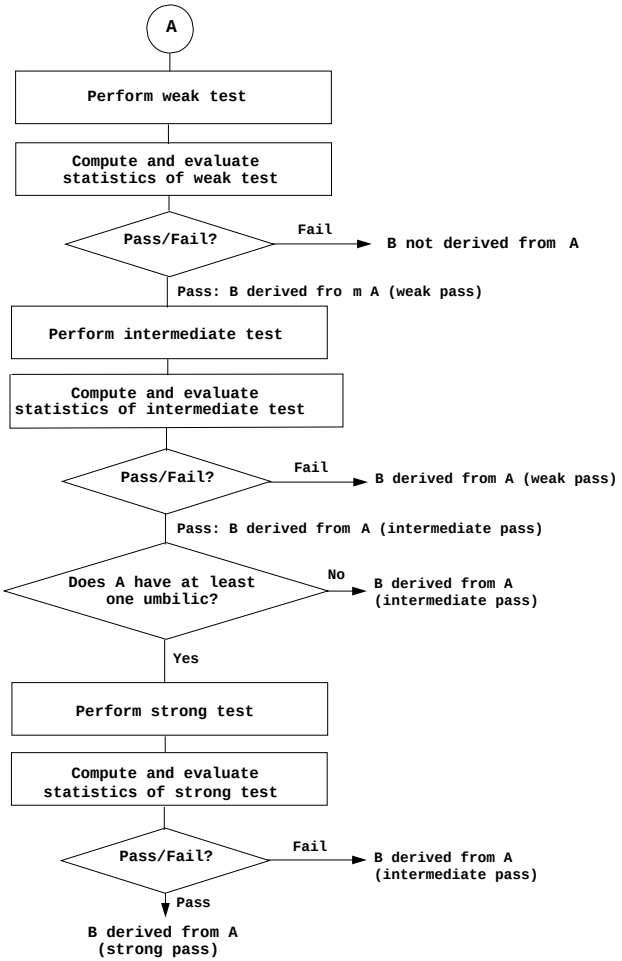

Figure 11: Algorithm 2

Items	Solid A	Solid B
Volume (mm^3)	83.794	18.007
Center of Volume (mm)	$(2.689 \times 10^{-5}, 7.016 \times 16^{-5}, 16.907)$	$(12.628, -8.354, 17.216)$
Principal Moments	1951.228, $(0.0, 0.0, 1.0)$	150.437, $(0.259, -0.330, 0.908)$
of Inertia (mm^5)	4867.277, $(1.0, 0.0, 0.0)$	373.973, $(0.951, -0.076, -0.299)$
and direction cosines (x,y,z)	6046.463, $(0.0, 1.0, 0.0)$	464.746, $(0.168, 0.941, 0.295)$

Table 2: Integral properties of solids A and B

The overall procedure is the same as Algorithm 1 except that no iteration is involved, see Figure 11. In this algorithm, a decision is made based on statistical information obtained in each test. From the weak test, statistics of the distance function are computed and evaluated by the user or a computer program. If the statistics pass a set of threshold tests, then it is concluded that B is derived from A under the weak test, and then the intermediate test begins. Otherwise, it is concluded that B is not derived from A.

The intermediate test constructs statistics of intrinsic properties, such as angle differences of the principal directions. A determination is made as to whether the statistics pass a set of threshold tests. If the threshold tests are negative, it is concluded that B is derived from A under the weak test. Otherwise, it is concluded that B is derived from A under the intermediate test.

Similarly, depending on the existence of umbilical points, the strong test may be performed. Statistics of position differences of the locations between corresponding isolated generic umbilics are considered in this test. A decision is made as to whether the statistics pass a set of threshold tests. If the tests are negative, B is concluded to be derived from A under the intermediate test. Otherwise, it is decided that B is derived from A under the strong test.

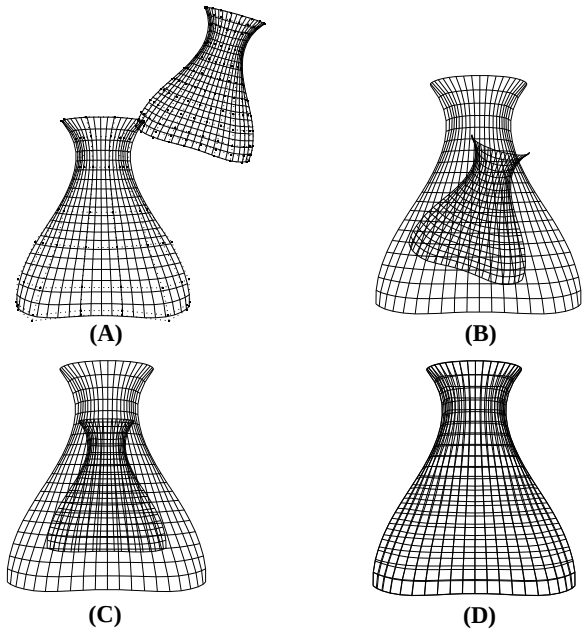

Figure 12: Matching via integral properties

8. EXAMPLES

8.1 Matching

Moment Method

Examples are presented to demonstrate the proposed algorithms. Solids bounded by bicubic integral B-spline surface patches A and B are used. Solid A is enclosed in a rectangular box of $25mm \times 23.48mm \times 11mm$. Here, the height of solid A is $25mm$. Figure 12 shows a sequence of operations for matching of the two surfaces using the principal moments of inertia of input solids. In this example, for clarity, only part of the boundary surfaces of the solids are displayed. The smaller solid has been translated, rotated, uniformly scaled and reparameterized. In Figure 12-(A), two boundary surfaces of the input solids are shown with their control points. Those two surfaces have similar shape but different numbers of control points and parametrization. Matching the centroids of the two solids is performed by translating the small solid by the position difference between the centroids, which is demonstrated in Figure 12-(B). The orientation of the largest principal moment of inertia of the solid A is aligned to that of the largest one of the solid B. Similarly, the remaining two orientations are aligned based on the values of the principal moments of inertia. After matching the orientations of the principal moment of inertia, the two solids are aligned in their orientations as shown in Figure 12-(C). Figure 12-(D) shows that the two solids match after uniform scaling obtained from the ratio between the volumes of the two solids, 4.651, is applied to the small solid. The centroids, volumes, principal moments and their directions are given in Table 2.

KH Method

A partial matching example using the KH method is presented in Figure 13. Figure 13 shows half of a car hood. It is represented by bicubic B-splines. The hood has 64 (8×8) control points (enclosed in a rectangular box of $13mm \times 12mm \times 6mm$) The relative error, i.e. the maximum distance divided by the square root of the surface area, is 0.00484.

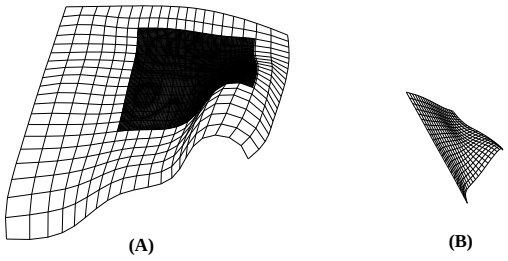

Figure 13: A partial matching example of a hood

Criteria	Max	Average	Standard Deviation
ϵ-offset (mm)	0.03456	0.00814	0.00665
Maximum principal curvature (mm^{-1})	0.07872	0.01572	0.01530
Minimum principal curvature (mm^{-1})	0.10577	0.01411	0.02165
Principal direction (rad)	0.70052	0.05657	0.11385

Table 3: Statistics of the quantities defined in Section 6.2 for each test

8.2 Copyright Protection

In this section, the two proposed similarity decision algorithms are demonstrated with the bottle example used for the moment method. After aligning two solids A and B shown in Figure 12, we are ready to assess the similarity between them. Here, part of the bounding surfaces are used for similarity checking. The surfaces are represented as bicubic B-splines and one surface has 64 (8×8) and the other 144 (12×12) control points with different parametrization. Both surfaces are enclosed in a rectangular box of $25mm \times 23.48mm \times 11mm$. The 413 node points are used from the wireframe given in Figure 9. The quantities provided in Section 6.2 are calculated and summarized in Table 3. All umbilical points for the two surfaces are located as shown in Figure 14. The Euclidean distances of the corresponding umbilical points are summarized in Table 4.

	1	2	3
Distances (mm)	0.08099	0.02954	0.08115

Table 4: Euclidean distances between the corresponding umbilics

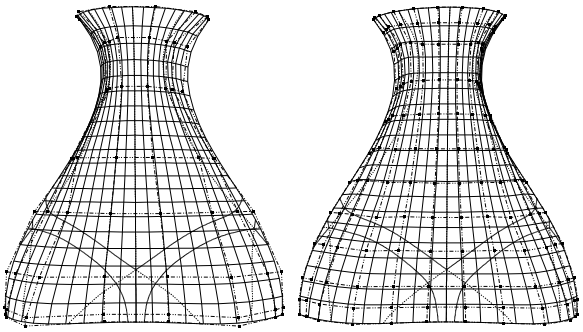

Figure 14: Comparison of lines of curvatures and umbilical points

In order to use Algorithm 1, we have to specify tolerances for ϵ_0, ϵ_1, ϵ_1' and ϵ_2. Depending on each tolerance, we can determine which test has passed or failed. Statistical information given in Table 3 is obtained for Algorithm 2. Suppose we have 0.01 as a tolerance for the weak test and subdivide the uv region into 400 square sub-regions (each box size of 0.05×0.05). The total number of sub-regions which contain footpoints P_i satisfying $\epsilon_i > 0.01$ is 31. Therefore, we can conclude that two surfaces are similar by 92.25% under the weak test with tolerance 0.01 and sub-region of size 0.05×0.05. This can be visualized as in Figure 15-(A). Here, the boxes indicate the regions which have at least one point with deviation larger than the tolerance 0.01. The results of the intermediate test using the maximum principal curvature is visualized in Figure 15-(B). Under the intermediate test for the maximum principal curvature with a tolerance 0.03, the similarity value between two surfaces is 91.25%. The strong test can also be performed based on the umbilical points for both surfaces as shown in Figure 14. Three star type umbilical points are identified for each surface, and the Euclidean distances between the corresponding umbilical points are calculated in Table 4. The types of the corresponding umbilical points match, and the position differences are small compared to the size of the object. Therefore, we may conclude that the strong test has passed.

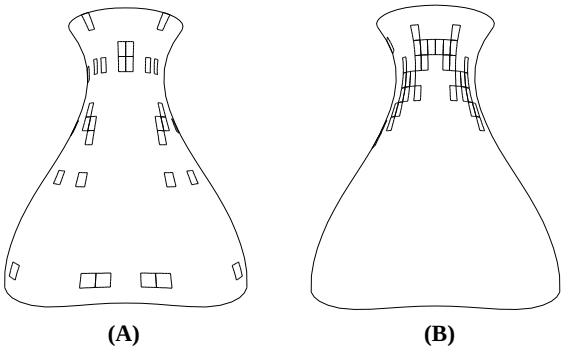

Figure 15: (A) Weak test (ϵ-offset) and (B) intermediate test (maximum principal curvature) based on Algorithm 2

The next example shows a case that one surface has been deformed so significantly that the strong test fails. Two surfaces in Figure 16 are represented as bicubic B-spline surface patches with 64 (8×8) control points, and are enclosed in a rectangular box of size $25mm \times 23.48mm \times 11mm$. The control points of surface A have been changed such that the difference of the bottom portion of surfaces A and B is noticeable. The surfaces are matched using the KH method

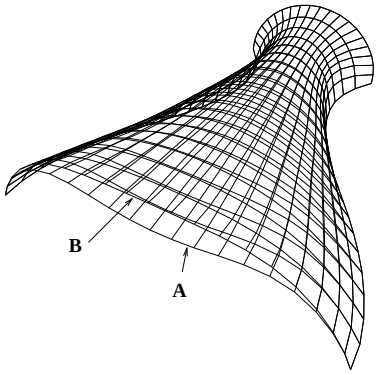

Figure 16: Surfaces for the failure case

Criteria	Max	Average	Standard Deviation
ϵ-offset (mm)	2.13435	0.26950	0.47164
Maximum principal curvature (mm^{-1})	0.19392	0.00222	0.03114
Minimum principal curvature (mm^{-1})	0.10979	0.01265	0.02135
Principal direction (rad)	1.54548	0.15166	0.27892

Table 5: Statistics of the quantities defined in Section 6.2 for the failure case

Test	Tolerance	Similarity (%)	Figure
ϵ-offset (mm)	0.48	86.50	17-(A)
Maximum principal curvature (mm^{-1})	0.04	88.50	17-(B)
Minimum principal curvature (mm^{-1})	0.02	89.00	
Principal direction (rad)	0.30	92.50	

Table 6: Quantitative similarity values for the failure case

with three seed points selected around the neck of surface A. A number of 366 node points are used for similarity tests. The statistical information for the similarity tests is summarized in Table 5. To assess local similarity, the uv parameter space is subdivided into 400 equal square boxes. Under the user-specified tolerances for each test, the quantitative similarity measures are summarized in Table 6. Figure 17 illustrates each test. The boxes indicate the regions where the condition for each test is not satisfied. All umbilical points have been calculated using the IPP algorithm as shown in Figure 18, which are provided as input to the strong test. Surface B has three star type umbilical points, whereas surface A has two umbilics of star type. Because this is a global matching case and the number of umbilical points is different, it is concluded that the strong test fails.

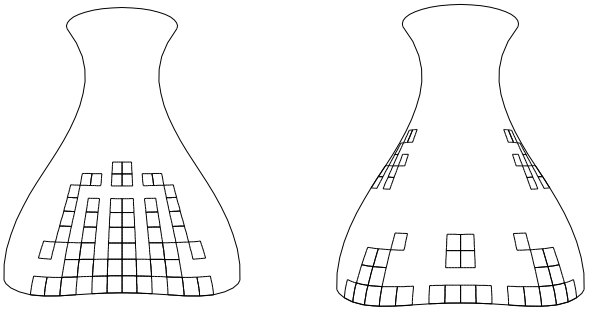

Figure 17: (A) ϵ-offset (B) Maximum principal curvature

9. CONCLUSIONS

We have addressed a problem of matching NURBS surface patches and solids bounded by NURBS surface patches, and introduced algorithms for a similarity decision. As auxiliary steps, methods of detection of umbilical points and construction of an intrinsic wireframe have been also proposed.

Quantitative assessment of matching is another issue discussed in this paper. Three hierarchical tests are proposed, and two decision algorithms are developed which provide systematic and statistical measures for a user to determine the similarity between two geometric objects.

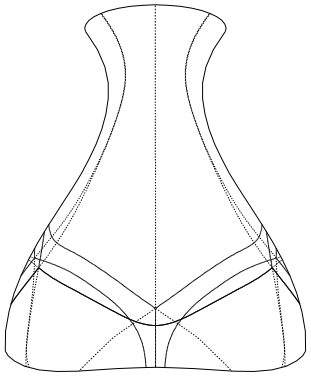

Figure 18: Umbilical points and lines of curvature

The proposed matching and similarity checking techniques can be used for copyright protection of NURBS surfaces. A user can compare a suspicious surface with a surface registered with an independent repository to check if the suspect surface is a copy of the copyrighted one. The partial matching technique may provide a method to determine whether or not part of the copyrighted surface has been stolen. Similarity decision depends on user-defined tolerances. Therefore, these tolerances need to be defined by an independent party.

Extension of this work to deal with the problem of matching and similarity evaluation for geometric objects expressed in different representation forms such as polyhedra and range data is a subject recommended for future study.

Acknowledgment

National Science Foundation, grant ♯ DMI-0010127.

10. REFERENCES

[1] O. Benedens. Geometry-based watermarking of 3D models. *IEEE Computer Graphics and Applications*, 19(1):46–55, 1999.

[2] R. Bergevin, D. Laurendeau, and D. Poussart. Estimating the 3D rigid transformation between two range views of a complex object. In *Proc. of 11th IAPR Intern. Conf. on Pattern Recognition. Conference A : Computer Vision and Applications*,

vol. 1, pp. 478–482, The Hague, The Netherlands, 1992. IEEE Computer Society Press.

[3] M. V. Berry and J. H. Hannay. Umbilic points on Gaussian random surfaces. *Journal of Physics A.*, 10(11):1809–1821, 1977.

[4] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[5] W. Blaschke and K. Leichtweiss. *Elementare Differential Geometrie*. Springer-Verlag, Berlin, 1973.

[6] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81(2):166–210, 2001.

[7] C. S. Chua and R. Jarvis. 3D free-form surface registration and object recognition. *International Journal of Computer Vision*, 17(1):77–99, 1996.

[8] T. Dokken. Finding intersections of B-spline represented geometries using recursive subdivision techniques. *Computer Aided Geometric Design*, 2(1-3):189–195, September 1985.

[9] Z. Duric, N. F. Johnson, and S. Jajodia. Recovering watermarks from images. *Technical Report: ISE-TR-99-04*, the Center for Secure Information Syhstems, George Mason University, 1999.

[10] C. Fornaro and A. Sanna. Public key watermarking for authentication of CSG models. *Computer Aided Design*, 32(12):727–735, 2000.

[11] J. M. Galvez and M. Canton. Normalization and shape recognition of three-dimensional objects by 3D moments. *Pattern Recognition*, 26(5):667–681, 1993.

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

[13] N. F. Johnson, Z. Duric, and S. Jajodia. A role for digital watermarking in electronic commerce. *ACM Computing Survey*, To appear. See also *http://www.jjtc.com/pub/*.

[14] S. Kanai, H. Date, and T. Kishinami. Digital watermarking for 3D polygons using multiresolution wavelet decomposition. In *Proc. of the Sixth IFIP WG 5.2/GI Intern. Workshop on Geometric Modelling: Fundamentals and Applications, Tokyo, December, 1998*, pp. 296–307, 1998.

[15] K. H. Ko, T. Maekawa, and N. M. Patrikalakis. An algorithm for optimal free-form object matching. *Computer Aided Design*, 2003, in press.

[16] K. H. Ko, T. Maekawa, N. M. Patrikalakis, H. Masuda, and F.-E. Wolter. Shape intrinsic watermarks for 3D solids. In *Proceedings of the 2003 NSF Design, Service and Manufacturing Grantees and Research Conference*, Birmingham, Alabama, January 2003.

[17] T. Maekawa. Computation of shortest paths on free-form parametric surfaces. *Journal of Mechanical Design, Trans. of the ASME*, 118(4):499–508, 1996.

[18] T. Maekawa and J. S. Chalfant. Computation of inflection lines and geodesics on developable surfaces. *Mathematical Engineering in Industry*, 7(2):251–267, 1998.

[19] T. Maekawa and N. M. Patrikalakis. Interrogation of differential geometry properties for design and manufacture. *The Visual Computer*, 10(4):216–237, 1994.

[20] T. Maekawa, N. M. Patrikalakis, F.-E. Wolter, and H. Masuda. Shape-intrinsic watermarks for 3-D solids. MIT Technology Disclosure Case 9505S, September 2001. Patent Attorney Docket No. 0050.2042-000. Application pending, January 7, 2002.

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

[22] R. Ohbuchi, H. Masuda, and M. Aono. Watermarking three-dimensional polygonal models through geometric and topological modifications. *IEEE Journal on Selected Areas in Communications*, 16(14):551–560, 1998.

[23] R. Ohbuchi, H. Masuda, and M. Aono. A shape-preserving data embedding algorithm for NURBS curves and surfaces. In *Proc. of Computer Graphics International, CGI '99, June 1999*, pp. 180–187. IEEE Computer Society, 1999.

[24] N. M. Patrikalakis and L. Bardis. Localization of rational B-spline surfaces. *Engineering with Computers*, 7(4):237–252, 1991.

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

[26] E. Praun, H. Hoppe, and A. Finkelstein. Robust mesh watermarking. In *Proc. of SIGGRAPH '99, Los Angeles, August, 1999*, pp. 49–56. ACM, 1999.

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

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

[29] H. Samet. Hierarchical data structures and algorithms for computer graphics. I. Fundamentals. *IEEE Computer Graphics and Applications*, 8:48–68, 1988.

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

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

[32] T. M. Tucker and T. R. Kurfess. Newton methods for parametric surface registration. Part I. Theory. *Computer-Aided Design*, 35(1):107–114, 2003.

[33] K. Yin, Z. Pan, J. Shi, and D. Zhang. Robust mesh watermarking based on multiresolution processing. *Computers & Graphics*, 25:409–420, 2001.

[34] J. Zhou, E. C. Sherbrooke, and N. M. Patrikalakis. Computation of stationary points of distance functions. *Engineering with Computers*, 9(4):231–246, 1993.