Journal of Zhejiang University SCIENCE A
ISSN 1673-565X (Print); ISSN 1862-1775 (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Automatic recognition of intersecting features of freeform sheet metal parts

Chun-jie ZHANG†, Xiong-hui ZHOU, Cong-xin LI
(*National Die and Mold CAD Engineering Research Center, Shanghai Jiao Tong University, Shanghai 200030, China*)
†E-mail: zhang_chunjie@263.net

Received Oct. 11, 2008; Revision accepted Apr. 24, 2009; Crosschecked Aug. 14, 2009

Abstract: This paper presents an approach for recognizing both isolated and intersecting geometric features of freeform surface models of parts, for the purpose of automating the process planning of sheet metal forming. The developed methodology has three major steps: subdivision of B-spline surfaces, detection of protrusions and depressions, and recognition of geometric features for sheet metal forming domain. The input geometry data format of the part is based on an IGES CAD surface model represented in the form of trimmed B-spline surfaces. Each surface is classified or subdivided into different curvature regions with the aid of curvature property surfaces obtained by using symbolic computation of B-spline surfaces. Those regions satisfying a particular geometry and topology relation are recognized as protrusion and depression (DP) shapes. The DP shapes are then classified into different geometric features using a rule-based approach. A verified feasibility study of the developed method is also presented.

Key words: Feature recognition (FR), Freeform surface, Feature intersection, Sheet metal forming
doi:10.1631/jzus.A0820705 **Document code:** A **CLC number:** TP391.7; TG3

INTRODUCTION

Feature technology is regarded as an important tool that provides high-level semantic information for interfacing CAD to other engineering applications. Feature recognition (FR) is the processing of a geometric model from a CAD system to find portions of the model matching the characteristics of interest for a given application, such as automated process planning and manufacturability analysis. To date, most studies have involved recognizing regular shaped machining features from CAD models. These feature recognition (FR) approaches include graph-based methods, hint-based methods, rule-based methods, and volume decomposition methods (Han *et al*., 2000; Shah *et al*., 2001; Owodunni and Hinduja, 2002). However, now more and more parts are manufactured using processes such as sheet metal forming, die casting, and injection molding. Components produced by these processes rarely show the regular, angular forms that characterize machined parts. Instead, they are created by freeform surfaces commonly represented by B-spline geometry. B-spline geometry and variable topology create difficulties for many existing recognition procedures developed for FR on 2.5-dimensional (2.5D) components.

In contrast to a large amount of literature about regular shaped FR, few studies related to recognizing features from CAD models with freeform surfaces have been published. Lentz and Sowerby (1993) developed a feature extraction approach for sheet-metal parts. They considered using the properties of general concave and convex regions, obtained using Gaussian and mean curvatures on the surface of a sheet metal component, to extract features. In addition, some types of intersection of concave and convex regions were discussed. Although the approach was limited to dealing only with quadric surfaces and partial torus surfaces, this region-based methodology was a useful contribution to research in this domain.

Sonthi *et al*.(1997) presented a general method for the recognition of freeform features using two principal curvatures. The surfaces are divided into curvature regions (CRs) of points with identical curvature types for both principal curvatures. A representation, the curvature region representation (CR-Rep), is then formed based on the curvature properties of all entities including faces, edges, and vertices. The primitive shapes, protrusions, and depressions (DP) are recognized within the CR-Rep using a graph-based approach. Sets of primitive shapes are identified as a feature if they match the CR-Graphs of the corresponding feature. The CR method is topologically general and can handle any surface geometry, but the mapping of parts into CRs involves the costly process of analyzing curvatures of freeform surfaces at a number of points, which is not always necessary.

Sundararajan and Wright (2004) described an algorithm for extracting features of a part containing a combination of 2.5D features and freeform surfaces. The recursive descent algorithm is employed to extract freeform pockets which are essentially volumetric machining regions used for CAPP and CNC code generation. The algorithm cannot distinguish between genuine freeform features and common non-2.5D features such as rounds and fillets.

Zhang *et al*.(2004) proposed a surface-based algorithm to recognize freeform geometric features. In contrast to the CR method, the product surfaces are subdivided into orientation regions based on the surface normal vector over a certain point density grid, and then all the connected regions with the same orientation type are grouped into surface orientation areas. The geometric feature can then be recognized through the analysis of area connectivity and relationship. This method is also based on discrete point computation and so has deficiencies similar to those of the CR method. Because a normal vector is used to subdivide surface, this method is also orientation-dependent.

Sridharan and Shah (2005) described a combination of graph- and rule-based algorithms for recognizing both simple and complex features having freeform faces. It is assumed that the faces are divided across regions of sharp curvature by edges. However, when a freeform face describes an entire part or region with DPs without the face being divided along regions of major curvature changes, this proposed method will fail to recognize the DP regions on the face. Also, only planar common shared faces are identified by the recognition system.

Lim *et al*.(2005) presented an edge-based method for identifying DP-features on freeform solids, represented as the B-Rep model with trimmed NURBS patches. Edge vexity and G1 continuity were used to identify the feature boundaries. But the algorithm was unable to identify DP features that are not bounded by real edges in the geometrical model.

Recently Sunil and Pande (2008) proposed a method to recognize features from freeform surface models of sheet metal parts represented in STL format. Geometrical properties, such as Gaussian and mean curvatures, are used to segment the STL model into meaningful regions. A feature on a freeform surface is defined simply as a set of connected meaningful regions having a particular geometry and topology. As the STL is already a discrete model, these geometrical properties were directly computed on each vertex of the triangle in the model.

These studies show that edge-based FR methods depend mainly on the original geometric model, and fail to recognize features from the freeform face with DPs without the face being divided along regions of major curvature changes. Thus, region-based methods are more promising for recognizing features of freeform CAD models. However, the key challenge of the region-based FR technology is to subdivide the surfaces efficiently into meaningful regions that facilitate the process of FR. Using previous region-based methods, it was necessary to compute some geometrical properties on a number of discrete points and then to group together all the connected element regions with the same type. This process is usually time-consuming and is unnecessary when the analytic function (such as the B-spline based representation) of the surfaces is given. In this paper, symbolic computation is used to characterize the curvature properties of the freeform surface and to help decompose the surface into different curvature regions. Symbolic computation is a global approach that has been successfully applied in freeform surface region optimization for 3- and 5-axis milling (Elber, 1995). The approach enables the property regions to be computed with exact precision and eliminates problems that rise from discrete sampling. So it is the perfect tool to preprocess the original surface model for subsequent FR. Once subdivision is completed, regions obtained that satisfy a particular geometry and topology relation will be recognized as DP shapes and further identified as geometric features using a rule-based approach. In this study, more complex intersections between DP shapes are discussed that may result in multiple interpretations of the same model.

SYMBOLIC COMPUTATION-BASED REGION SUBDIVISION OF B-SPLINE SURFACE

Freeform surface using B-spline-based representation has gained tremendous popularity in the CAD/CAM community. Several properties such as the surface slope, speed, and curvature, may be of interest to the designer or required for surface analysis. Here, property surfaces whose definitions are derived from different attributes of the original surface are used as auxiliary surfaces to help analyze the original surface. They are usually represented as original B-spline surfaces with some symbolic operators including dot/cross products, sum/difference, and derivative operators acting on them. If the property surface is also representable as a B-spline and shares the same parametric domain with the original surface, some useful shape information can be mapped back easily to the original surface.

Differential geometry

The major property of interest is the curvature of the surface. Before further discussion, several property surfaces which are helpful for analyzing surface curvature will be introduced.

Let $\boldsymbol{r}=\boldsymbol{r}(u, v)$ be a C^2 regular parametric surface. The Gaussian curvature is a scalar value defined as

$$K=\frac{LN-M^2}{EG-F^2}. \tag{1}$$

The mean curvature is defined by

$$H=\frac{LG-2MF+NE}{2(EG-F^2)}, \tag{2}$$

where

$$L=\boldsymbol{r}_{uu}\cdot\frac{\boldsymbol{r}_u\times\boldsymbol{r}_v}{|\boldsymbol{r}_u\times\boldsymbol{r}_v|},\ M=\boldsymbol{r}_{uv}\cdot\frac{\boldsymbol{r}_u\times\boldsymbol{r}_v}{|\boldsymbol{r}_u\times\boldsymbol{r}_v|},\ N=\boldsymbol{r}_{vv}\cdot\frac{\boldsymbol{r}_u\times\boldsymbol{r}_v}{|\boldsymbol{r}_u\times\boldsymbol{r}_v|},$$
$$E=|\boldsymbol{r}_u|^2,\ F=\boldsymbol{r}_u\cdot\boldsymbol{r}_v,\ G=|\boldsymbol{r}_v|^2. \tag{3}$$

According to the sign of Gauss curvature (K) and mean curvature (H), six types of CR are summarized in Table 1.

Table 1 Surface types based on signs of Gauss and mean curvatures

Region type	Gaussian curvature	Mean curvature	Example
Peak	+	−	
Pit	+	+	
Ridge	0	−	
Valley	0	+	
Saddle	−	Any	
Flat	0	0	

Assuming the surface is curvature continuous, a locus of points satisfying K=0 is called a parabolic curve, since it separates ‘elliptic’ regions of positive Gaussian curvature from ‘hyperbolic’ regions of negative Gaussian curvature on the surface. Denoting the numerator of Eq.(1) as Δ, the problem of computing K=0 is equivalent to solving Δ=0 (the denominator is always non-zero for a regular surface). Substituting Eq.(3) into Δ gives

$$\Delta=\frac{(\boldsymbol{r}_{uu}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))(\boldsymbol{r}_{vv}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))-(\boldsymbol{r}_{uv}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))^2}{|\boldsymbol{r}_u\times\boldsymbol{r}_v|^2}. \tag{4}$$

Since the zero set is of interest, it is necessary only to examine the numerator of Eq.(4). Let Ψ be the numerator of Eq.(4). Then the parameter values of the intersection curve between Ψ and plane Z=0 can serve as the parameter values of the trimming curves of the original surface. The sign of Ψ at a single point on each trimmed surface is used to classify the type of region. Because the signs of the peak and pit regions in Ψ are both positive, they are further distinguished from each other by simply evaluating the sign of L at that single point. In addition, in the case $\Psi\equiv0$, the surface is developable and may consist of a ridge, valley, or flat region. The method for identifying them is identical to that for distinguishing peak, pit, and saddle regions as long as we can compute the zero set of the scalar field H. By simplification, the scalar field Φ with the same zero set of H can be obtained as follows:

$$\begin{aligned}\Phi &= (\boldsymbol{r}_{uu}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))\,|\,\boldsymbol{r}_v\,|^2 - 2(\boldsymbol{r}_{uv}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))\boldsymbol{r}_u\cdot\boldsymbol{r}_v\\ &\quad+(\boldsymbol{r}_{vv}\cdot(\boldsymbol{r}_u\times\boldsymbol{r}_v))\,|\,\boldsymbol{r}_u\,|^2,\end{aligned}$$

and the zero sets of Φ are used to trim the developable surface into ridge, valley, or flat regions.

Obviously, both Ψ and Φ are signed Gaussian and mean curvature scalar fields which provide a qualitative description of the freeform surfaces' curvature.

Symbolic computation of B-splines

Symbolic computation is used to obtain the B-spline-based representation of Ψ and Φ. A set of symbolic operators of B-spline entities was presented in some studies (Piegl and Tiller, 1997; Chen *et al*., 2007). Central to most symbolic operators is the computation of the product of B-spline, which is the prerequisite component of dot/cross products. In our research, the method proposed by Piegl and Tiller (1997) is employed. It is an indirect approach that exploits the algorithm for computing the product of Bezier functions. The major steps of the algorithm are: decompose the B-splines into their Bezier components using knot insertion, compute the product of the Bezier functions, and recompose the Bezier product functions into B-spline form using knot removal. The algorithm relies on the product of Bezier patches. They are computed as follows:

$$B_{k,l}=\sum_{i=\max(0,k-p_2)}^{\min(p_1,k)}\ \sum_{j=\max(0,l-q_2)}^{\min(p_1,l)}\frac{\mathcal{P}}{\mathcal{Q}}(B_1)_{i,j}(B_2)_{k-i,l-j},$$

where

$$\mathcal{P}=\binom{p_1}{i}\binom{p_2}{k-i}\binom{q_1}{j}\binom{q_2}{l-j},$$

$$\mathcal{Q}=\binom{p_1+p_2}{k}\binom{q_1+q_2}{l},$$

$$k=0,1,\ldots,p_1+p_2;\ l=0,1,\ldots,q_1+q_2,$$

B_1 and B_2 denote the coefficients of the corresponding Bezier patches in $B_1(u, v)$ and $B_2(u, v)$ respectively, and B denote the coefficients of $B(u, v)$ which is the product of $B_1(u, v)$ and $B_2(u, v)$. p_1, q_1 and p_2, q_2 are the u, v degrees of $B_1(u, v)$ and $B_2(u, v)$, respectively.

Another important symbolic operator is derivative. Given a B-spline surface

$$\boldsymbol{r}(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{m}N_{i,p}(u)N_{j,q}(v)P_{i,j}$$

defined over

$$U=\{\underbrace{a_u,\ldots,a_u}_{p+1},u_{p+1},\ldots,u_n,\underbrace{b_u,\ldots,b_u}_{p+1}\},$$

$$V=\{\underbrace{a_v,\ldots,a_v}_{q+1},v_{q+1},\ldots,v_m,\underbrace{b_v,\ldots,b_v}_{q+1}\},$$

the (1, 0) derivative surface is computed as follows:

$$\boldsymbol{r}_u(u,v)=\sum_{i=0}^{n-1}\sum_{j=0}^{m}N_{i,p-1}(u)N_{j,q}(v)P_{i,j}^{(1,0)},$$

where

$$P_{i,j}^{(1,0)}=p\frac{P_{i+1,j}-P_{i,j}}{u_{i+p+1}-u_{i+1}},\quad V^{(0)}=V,$$

$$U^{(1)}=\{\underbrace{a_u,\ldots,a_u}_{p},u_{p+1},\ldots,u_n,\underbrace{b_u,\ldots,b_u}_{p}\}.$$

Other partial derivatives of the B-spline surface can be computed analogously. As for the operator of sum, difference, and division, we can make two surfaces compatible using degree elevation and knot refinement. Once both surfaces are transformed to have a common order and knot vector, their control polygons can be summed, differenced, or divided.

Since the operators involved in Ψ and Φ are all covered, we now can obtain the B-spline representation of the property surface from the original one.

Curvature region construction

In this section, the intersection curve between the signed curvature scalar fields and the plane Z=0 will be computed. Because the signed curvature scalar fields are globally computed, the accuracy of segmentation is within machine precision. If the coefficients of the scalar field are all positive or negative, then the scalar field will have no intersection with the Z=0 plane according to the convex hull property of the B-spline. The original surface of such a signed Gaussian curvature scalar field is completely peak/pit or saddle. So only those scalar fields whose coefficients have different signs need intersection computation to extract the trimming curve for surface subdivision. Several object classes of Open CASCADE 6.2 Object Libraries are used to compute the intersection curves and subdivide the original surface into CRs. The obtained result is robust. After each surface patch is type-determined or subdivided into CRs, the region boundary and the connecting relation between CRs are obtained. If the connected regions in two stitched surface patches share the same region type then they are incorporated into one region. The edge boundary of the merged region is determined by grouping the boundaries of two regions without their shared common edges. Each region has one outer boundary loop and any number of inner loops. Boundary edges which belong to only one CR are called free edges.

Two examples are given in Figs.1 and 2 (see p.1447). Fig.1 shows a subdivided biquartic B-spline surface with its scalar field Ψ intersecting plane Z=0. Fig.2 shows a subdivided developable surface with its scalar field Φ intersecting plane Z=0. All figures have been colored, with yellow marking saddle, red for peak, green for pit, pink for ridge, and blue for valley.

DP SHAPE DETECTION

The CR is an aggregate geometric abstraction which contains several topological entities sharing identical curvature properties. It represents the model at a higher level. However, features tend to be very domain-specific. There is a need to build a level of abstraction intermediate between the CR and the domain-dependent feature, which has a certain meaning but yet is domain-independent. Depressions and protrusions are such domain-independent entities and they can be obtained conveniently from the CR abstraction.

Definition

The classification of domain-independent primitive shapes is based on the following definitions:

1. Pure peak protrusion: a shape comprising only peak regions.

2. Pure pit depression: a shape comprising only pit regions.

3. Pure ridge protrusion: a shape comprising only ridge regions.

4. Pure valley depression: a shape comprising only valley regions.

5. Compound protrusion: a shape consisting of a combination of peak, ridge, and flat regions with C^0, C^1 continuity and which must contain at least one peak region.

6. Compound depression: a shape consisting of a combination of pit, valley, and flat regions with C^0, C^1 continuity and which must contain at least one pit region.

7. Transition: A transition usually lies between DP shapes or other unallocated flat, ridge, or valley regions and separates them from each other. It must contain at least one saddle region. Regions forming a particular transition zone may be chosen from the following two combinations with two different vexity characteristics: (a) Saddle, peak, and ridge regions (a convex region combination); (b) Saddle, pit, and valley regions (a concave region combination). A transition is either a chain or a simple cycle of edge-connected regions. The regions of the transition are edge- or vertex-adjacent to the outer boundary of the shapes it separates.

Algorithm for detection of all kinds of primitive shapes

According to the definition of all kinds of primitive shapes, the algorithm for detecting them will be described below. Before detection, all the CRs are marked as unallocated.

1. Detecting pure peak (pit) protrusions (depressions)

Detecting these two kinds of shapes is simple. A pure peak (pit) protrusion (depression) is formed by a peak (pit) region whose outer boundary edges are free or connected to a saddle region. Thus, pure peak (pit) protrusions (depressions) are extracted by searching all the peak (pit) regions that satisfy this condition.

2. Detecting pure ridge (valley) protrusions (depressions)

A pure ridge (valley) protrusion (depression) is formed by a ridge (valley) region in which the edges of its outer boundary loop are free or connected to a saddle region. Thus, pure ridge (valley) protrusions (depressions) are extracted by searching all the ridge (valley) regions that satisfy this condition.

3. Detecting compound protrusions (depressions)

Detection of a compound protrusion (depression) is employed by some region growing rules. The growth of a compound protrusion (depression) starts with the searching of an unallocated peak (pit) region. An adjacent region is then merged into the growing primitive shape if it satisfies the definition of a compound protrusion (depression).

However, sometimes the obtained compound protrusion (depression) may be incorrect when a compound protrusion intersects a compound depression. The transition between them may be improperly absorbed by one of the shapes resulting in failure to detect the transition between them. For example, the valley regions marked by blue and the flat regions marked by gray in Fig.3a (see p.1447) will be improperly absorbed by a compound depression marked by green in Fig.3b. Fortunately, this pathological case can be avoided by changing the order of creating two such primitive shapes, since one of them must not absorb the transition falsely. The method for discovering and processing such a pathological case will be discussed in the next section.

4. Detecting transitions

Detection of a transition is also achieved using some growing rules. As a transition must have at least one saddle region, the detection starts with the searching of an unallocated saddle region, which is regarded as a seed region to grow. To cover different types of intersection of primitive shapes, three schemas used to build transition are described below.

(1) Transition to be absorbed. This condition occurs when two protrusions (depressions) intersect each other and share a common region at the same time. In schema (a), an unallocated saddle region which is adjacent to only a primitive shape (assume it is shape A) will be selected as a seed to detect transition. The region adjacent to this seed region will then become part of the transition if it satisfies the following conditions:

- It is the only region not belonging to shape A or the detected transition, that is, adjacent to the seed;
- The vexity of the region is opposite to the vexity of shape A, i.e., convex (concave) for depression (protrusion) shapes;
- If it is a peak/pit it should have at least one boundary edge shared by shape A;
- If it is a ridge/valley it should be edge-adjacent to two regions in shape A;
- If it is a saddle it should be adjacent to the outer boundaries of shape A and have at least one boundary edge shared by shape A.

Once such a region is found, it will be treated as a new seed region to search for further regions that satisfy the above five conditions. If not, the saddle region will be abandoned. The loop of the growing process will be terminated if there is no region available for growing. The obtained transition should also accord with the definition of the transition mentioned previously. In contrast to methods for dealing with other kinds of transition, the detected transition will be absorbed into the shape instead of separating it. To avoid affecting the detection of other transitions, these kinds of seed regions should be dealt with first. For example, the saddle region (marked by yellow) in Fig.4 (see p.1447) is adjacent to a compound protrusion shape (marked by red). There is a valley (marked by blue) edge-adjacent to this saddle and its edges, which are vertex-adjacent to the saddle, are shared by the compound protrusion shape. So the saddle and the valley conform to make a transition to be absorbed.

(2) Clear-cut transition. When two protrusions/depressions intersect each other, or a protrusion/depression is embedded in a group of unallocated flat, ridge, or valley regions, the transitions between them have not been absorbed and so this kind of transition is clear-cut. In such cases an unallocated saddle region which is adjacent to two shapes with the same vexity, or adjacent to a primitive shape and an unallocated flat/ridge/valley region will be taken as a seed region. The requirement for the region adjacent to this seed region to be incorporated should also be met by an unallocated region edge- or vertex-adjacent to the boundaries of both sides, and the absorbed region will be treated as a new seed for further growing. The detected transition should accord with the definition. When a detecting loop is finished, the obtained transition will be regarded as a separation between the shapes or unallocated regions.

(3) Unclear transition. As mentioned previously, when a compound protrusion intersects a compound depression the transition between them may be absorbed by one of the two shapes except the saddle region. Similar to other schema, an unallocated saddle adjacent to a compound protrusion and a compound depression is regarded as seed. Because there is no unallocated region adjacent to the saddle region selected as a seed, the requirements for the region adjacent to this seed region to become part of transition are modified as follows:

It is a peak/pit region in one shape and edge- or vertex-adjacent to the boundaries of the other shape; or a ridge/valley in one shape and edge-adjacent to the boundaries of the other; or a saddle adjacent to the boundaries of both shapes.

The first selected region connected to the seed will decide the combination of the transition and from which shape the region forming the transition zone will be chosen. Once such a region is found it will be removed from the shape to which it originally belonged and become a part of a transition. It will then be treated as a new seed for further growing.

Because the detected compound protrusions (depressions) may be pathological, the process of identifying the transitions between them could encounter some conditions which are contradictory to the definition of primitive shapes and transitions, as follows: i) There is no saddle region at all between two shapes; ii) There are no remaining candidate regions for growing and no other saddle adjacent to two shapes, but there still exist regions of both shapes that are connected to each other.

When the above conditions occur or the obtained transition is invalid according to the definition of transition, we need to change the order of detecting such two primitive shapes and then identify the transition between them again. The transition can then be generated successfully. Fig.5a (see p.1447) shows the primitive shapes constructed in Fig.3. The saddle regions in yellow and the ridge regions in pink will be absorbed into the depression shape in green for they satisfy the requirements for the transition to be absorbed. The shapes after the transition is absorbed (Fig.5b) fail to detect correct transition because there is no saddle between two detected shapes. Fig.5c shows the generated transition after changing the order of detecting the primitive shapes with the protrusion in red, the transition in yellow, and the depression in green.

Other complex intersections leading to multiple interpretations

When two intersected protrusions (depressions) with common regions intersect another protrusion, depression, or flat region, some ambiguity arises. The transition between two shapes may produce pit/peak regions and it can make sense if a compound depression/protrusion grows from them because a saddle also exists in the original protrusion/depression to build the transition. Two alternative interpretations of the model are then available.

For example, Fig.6 (see p.1447) shows two interpretations of the same model: one is a union of two intersected protrusions embedded in a flat region shown in grey; the other is a union of intersected depressions growing from the pit region shown in dark green and enclosing a grey flat region. The regions in pink are the transitions absorbed by the protrusion/depression respectively and the regions in yellow are the transitions separating the shapes or the flat regions.

Fig.7a (see p.1447) shows the status before detecting the transition. Every saddle, as a seed, can choose a region from either shape (in green or red) as its descendant to build the transition. But one path is invalid because it breaks the rule of being a chain or simple cycle. As shown in Figs.7b and 7c, four saddle regions make one transition and the left saddle region makes another. Once the transitions are detected and no regions in protrusion and depression are adjacent, the redundant saddle can be absorbed by the adjacent shape (as the saddles shown in pink).

Fig.8a (see p.1447) shows the result of a multiple intersected protrusion and three chains of transition identified from three saddles meeting in a pit region. Another interpretation is that a depression growing from the pit region shown in dark green is embedded in a protrusion (Fig.8b).

All the above examples show the existence of multiple interpretations of the same model. The interpretation made depends on the order in which the shapes are detected. Some hints are used to find all the alternative interpretations of the model. If there exist pit or peak regions in the transition to be absorbed or in the transition between a pair of intersecting DP shapes, then the alternative interpretation can be activated by changing the building order of the related shape and identifying the transition between them. One special case (Fig.7) occurs when the saddle to be absorbed by the shape is instead selected as seed to detect a new transition. Once such processes are successful, an alternative interpretation is generated.

FEATURE RECOGNITION FOR SHEET METAL FORMING DOMAIN

Unlike methods for recognizing machining features that reveal only the concave entities of the model, the approach for sheet metal forming domain needs to identify both protrusions and depressions. When multiple interpretations arise there should be a choice between them.

Sheet metal for automobile body parts is usually formed by only one operation that incorporates a deep drawing and bulging process. Deep drawing produces the main shape of the part, which is either protrusion-dominant or depression-dominant associating with the direction of the punch. To avoid unnecessary combinations, only two interpretations, protrusion- or depression-dominant, of the whole part are needed. For example, in the protrusion-dominant context, the interpretation of protrusion shape is preferred when both alternatives are available. In addition, some models may have two alternative interpretations but one of them may be meaningless in sheet metal forming domain. Figs.6b and 7c are two alternative interpretations, but the flat region as a flange cannot be completely enclosed by the depression drawing cavity. Once such an instance occurs it will be abandoned and the other interpretation will be accepted whether or not it conforms to the dominant shape of the whole part.

In current research, geometric features in sheet metal forming domain are obtained by interpreting the domain-independent DP shapes and specific types region. This paper focuses mainly on deformation features (Fontana *et al*., 1999). A variety of protrusion and depression features such as boss, beads, dimple, dump, dart, and ridge are recognized. Other elimination features, such as holes, are not discussed. For a comprehensive account, please refer to Fontana *et al*. (1999).

A rule-based approach is used in this work for FR. Geometrical and topological information of the identified shapes and regions are applied to recognize features. Detailed FR rules for some of the features are described below.

Dart

A dart feature is generally used as a stiffening feature which helps to maintain a 90-degree flange or a protrusion/depression having a high/deep side wall. It is a ridge (valley) region sandwiched between two triangular shaped flat regions enclosed with a simple cycle of transition of concave (convex) region combination. Dart features usually exist in the rest of the unallocated regions or the regions absorbed as transitions by some DP shapes. The ridge/valley regions appearing in two parts of the model above can be treated as a hint to search for dart features. Fig.9 is an example of a dart feature which improves the stiffness of a bend. The ridge is originally a part of a transition absorbed by the depression shape growing from the pit region shown in green.

DP shapes attributed to the feature of drawing and bulging

Drawing operations usually involve severe plastic deformation and the material of the part extends around the sides of the punch. Bulging is a kind of operation which makes material stretch and produces local protrusions and depressions on the blank. Most detected DP shapes are attributed to these two operations. The biggest difference between the features of drawing and bulging operations relates to whether the detected DP has an obvious side wall or not. Fig.10 shows different features of these two operations commonly used in sheet metal forming.

Fig.10a shows a feature of bulging called a dent. It is actually a pure pit depression (marked by green) embedded in a flat region (marked by grey) with the transition (marked by yellow) separating them. In Fig.10b there is a bead which is also produced by a bulging operation. A bead feature is a kind of compound depression combining a valley and a pit region (marked by green). The feature in Fig.10c is a boss formed by a deep drawing process. A boss feature is also a kind of compound depression. It differs from the bead in that the connected element region of a boss should satisfy the following geometric and topological conditions:

1. There exists an internal flat region completely enclosed by a cycle of regions consisting of ridge, peak, or saddle regions.

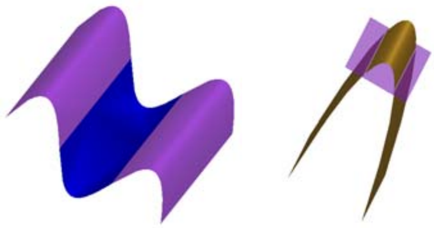
2. Adjacent to the outer boundary loop of the region cycle is a cycle of regions made up of ridge, valley, or flat regions.

In Fig.10c the first cycle of regions is marked by red and the second by purple.

Here we describe only three kinds of protrusion and depression features. Other features can be constructed by specific configurations of the element regions of the primitive shapes.

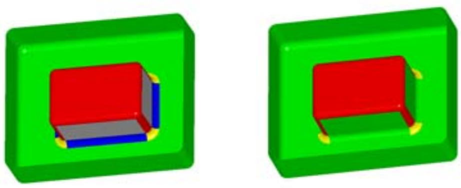
(a) (b)

Fig.1 Region segmentation of a B-spline surface
(a) A biquartic B-spline surface with peak (red), pit (green), and saddle (yellow); (b) The scalar field surface of Ψ with its zero set



(a) (b)

Fig.2 Region segmentation of developable surface
(a) A developable surface with ridge (pink) and valley (blue); (b) The scalar field surface Φ with its zero set



(a) (b)

Fig.3 Incorrect compound protrusion (depression) detection
Valley regions (blue) and flat regions (gray) in (a) are improperly absorbed by a compound depression (green) in (b)

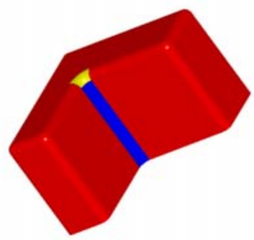
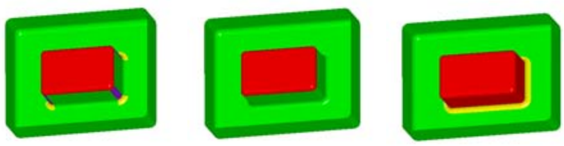
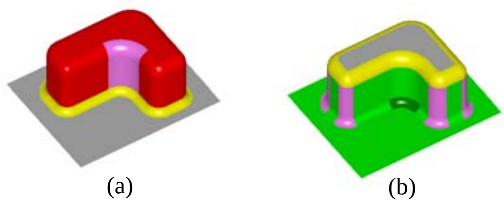


Fig.4 Transition (the region marked by yellow and blue) to be absorbed
The saddle region (yellow) is adjacent to a compound protrusion shape (red)



(a) (b) (c)

Fig.5 Detecting transition between DP shapes
(a) Original result of DP shapes; (b) Result after transition absorbed; (c) Correct result after changing the creation order of DP shapes

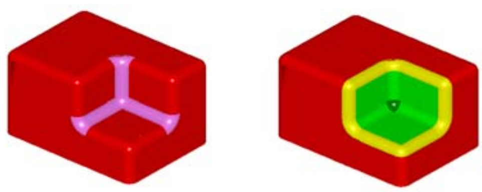


(a) (b)

Fig.6 Multiple interpretations of sample part 1
(a) Interpretation A; (b) Interpretation B

(a) (b) (c)

Fig.7 Multiple interpretations of sample part 2
(a) Status before detecting the transition; (a) Interpretation A; (c) Interpretation B



(a) (b)

Fig.8 Multiple interpretations of sample part 3
(a) Interpretation A; (b) Interpretation B

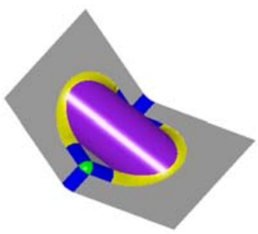
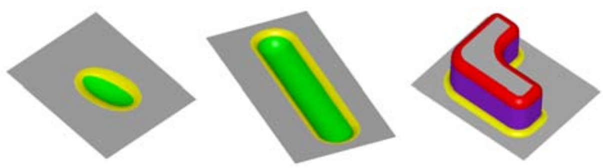


Fig.9 Example of dart feature



(a) (b) (c)

Fig.10 Frequently used features
(a) Dent; (b) Bead; (c) Boss

IMPLEMENTATION AND VERIFICATION

The proposed approach has been implemented in a prototype system running on a Pentium IV PC configuration. The development platform was VC++ 6.0, and the 3D modeling kernel and display interface was Open CASCADE (version 6.2, http://www.open-cascade.org), which is available in open source. The FR system developed can accept any freeform surface CAD model represented in trimmed B-spline surface format from an IGES text file exported by a general CAD system like UG, Pro/E, CATIA, SolidWorks, etc. The developed algorithms have been tested with a number of test parts and industry sheet metal part models. Typical case studies are presented as follows.

Case study 1

Figs.11 and 12 demonstrate the procedures of feature extraction for an industrial automotive cross member. The surface part model had 209 faces and the IGES file was about 3.3 MB. The system was able to classify correctly all curvature regions and recognize protrusions and depressions instantaneously. Fig.11 shows the curvature regions of the part model, identified by different colors. By assuming the model was protrusion-dominant, the DP shapes were recognized and the result is displayed in Fig.12 with a depression/protrusion tree (DP tree) listing the DP shapes and their corresponding element curvature regions. There were three protrusions and five depressions in this part. Among the five depression shapes three were dent features, one was a bead feature, and one was a flat-bottom bulging feature.

Case study 2

The part shown in Figs.13 and 14 is a floor member of an automotive. The IGES file was about 5.8 MB and contained 558 faces. Fig.13 illustrates the

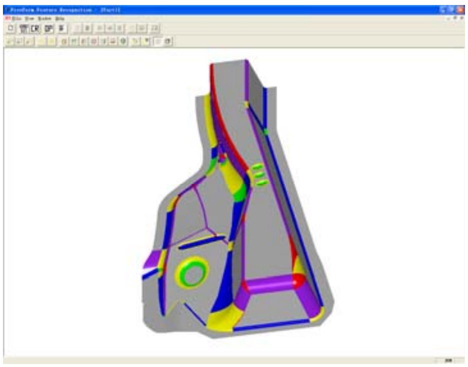


Fig.11 Curvature regions of case study 1
Peak in red, pit in green, ridge in purple, valley in blue, saddle in yellow, and flat in grey

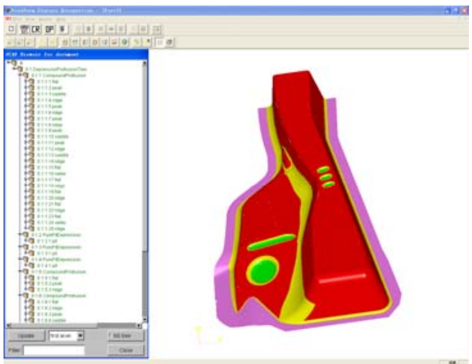


Fig.12 Depression/Protrusion (DP) primitive shapes and DP tree of case study 1
Protrusion in red, depression in green, transition in yellow, and other unallocated regions in pink

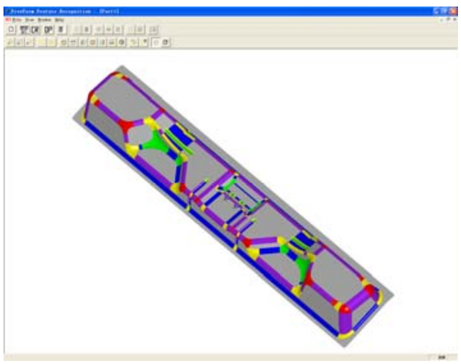


Fig.13 Curvature regions of case study 2
Peak in red, pit in green, ridge in purple, valley in blue, saddle in yellow, and flat in grey

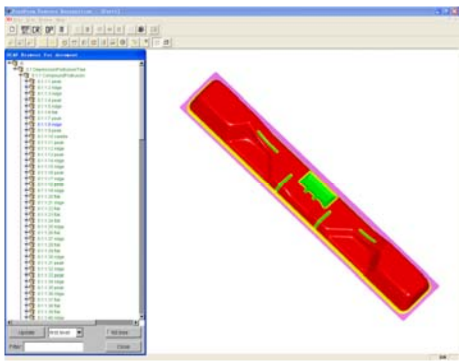


Fig.14 Depression/Protrusion (DP) primitive shapes and DP tree of case study 2
Protrusion in red, depression in green, transition in yellow, and other unallocated regions in pink

curvature regions of the part model, identified in different colors, and Fig.14 shows the recognized DP shapes assuming protrusion dominance. In Fig.14, four beads and one depressive boss are shown embedded in a main compound protrusion.

CONCLUSION

In this paper an improved region-based FR approach is proposed, which is capable of automatically analyzing the geometry of a B-spline surface model and generating the geometric features of freeform sheet metal parts. The symbolic computation of B-splines, a tool which plays an important role in many CAD/CAM areas, is applied to obtain meaningful regions from low level B-splines surfaces. A procedure to detect domain-independent primitive shapes, such as DP shapes, from those meaningful regions has been developed to define a higher level representation of a model which is suitable for FR of specific domains. Different types of intersections between DP shapes are also dealt with during this process. According to the geometric information and topological relationships of shapes, a rule-based FR approach suited to sheet metal forming domain is presented. Some deformation features are discussed in detail. The method was implemented and tested in some case studies to verify its capabilities in identifying freeform features on sheet metal parts.

The method may also serve as an interface that links measured data to the downstream application such as sheet metal tool design and process planning, if some B-spline surfaces are reconstructed to fit the original dataset. However, there exists one deficiency in this algorithm. To obtain the zero sets of Gauss curvature K and mean curvature H, the scalar field surfaces Ψ and Φ are derived from the numerators of K and H. Such simplification may lead to unstable results when the curvatures of surface regions are very close to zero. So the scalar field surfaces could be improved by taking the denominators of K and H into consideration. Future work may also include recognizing flange features whose element regions may be still in unallocated regions or have already been contained in compound DP shapes. Other possible issues may include the parameterization of recognized features, freeform feature deformation, and mouldability analysis.

References

Chen, X.M., Riesenfeld, R.F., Cohen, E., 2007. Sliding Windows Algorithm for B-spline Multiplication. Proc. ACM Symp. on Solid and Physical Modeling, p.265-276. [doi:10.1145/1236246.1236283]

Elber, G., 1995. Freeform surface region optimization for three- and five-axis Milling. *Computer-Aided Design*, **27**(6):465-470. [doi:10.1016/0010-4485(95)00019-N]

Fontana, M., Giannini, M., Meirana, M., 1999. A free form feature taxonomy. *Comput. Graph. Forum*, **18**(3):107-118. [doi:10.1111/1467-8659.00332]

Han, J.H., Pratt, M., Regli, W.C., 2000. Manufacturing feature recognition from solid models: a status report. *IEEE Trans. Rob. Autom.*, **16**(6):782-796. [doi:10.1109/70.897789]

Lentz, D.H., Sowerby, R., 1993. Feature extraction of concave and convex regions and their intersection. *Computer-Aided Design*, **25**(7):421-437. [doi:10.1016/0010-4485(93)90004-8]

Lim, T., Medellin, H., Torres-Sanchez, C., Corney, J.R., Ritchie, J.M., Davies, J.B.C., 2005. Edge-based identification of DP-features on free-form solids. *IEEE Trans. Pattern Anal. Mach. Intell.*, **27**(6):851-860. [doi:10.1109/TPAMI.2005.118]

Owodunni, O., Hinduja, S., 2002. Evaluation of existing and new feature recognition algorithms: Part 1. Theory and implementation. *Proc. Inst. Mech. Eng., Part B: J. Eng. Manuf.*, **216**(6):839-851. [doi:10.1243/095440502320192978]

Piegl, L., Tiller, W., 1997. Symbolic operators for NURBS. *Computer-Aided Design*, **29**(5):361-368. [doi:10.1016/S0010-4485(96)00074-7]

Shah, J.J., Anderson, D., Kim, Y.S., Joshi, S., 2001. A discourse on geometric feature recognition from CAD models. *J. Comput. Inf. Sci. Eng.*, **1**(1):41-51. [doi:10.1115/1.1345522]

Sonthi, R., Kunjur, G., Gadh, R., 1997. Shape Feature Determination Using Curvature Region Representation. Proc. 4th ACM Symp. on Solid Modeling and Applications, p.285-296. [doi:10.1145/267734.267805]

Sridharan, N., Shah, J.J., 2005. Recognition of multi-axis milling features: Part II. Algorithms & implementation. *J. Comput. Inf. Sci. Eng.*, **5**(1):25-34. [doi:10.1115/1.1846054]

Sundararajan, V., Wright, P.K., 2004. Volumetric feature recognition for machining components with freeform surfaces. *Computer-Aided Design*, **36**(1):11-25. [doi:10.1016/S0010-4485(03)00065-4]

Sunil, V., Pande, S., 2008. Automatic recognition of features from freeform surface CAD models. *Computer-Aided Design*, **40**(4):502-517. [doi:10.1016/j.cad.2008.01.006]

Zhang, X., Wang, J., Yamazaki, K., Mori, M., 2004. A surface based approach to recognition of geometric features for quality freeform surface machining. *Computer-Aided Design*, **36**(8):735-744. [doi:10.1016/j.cad.2003.09.002]