

MODELLING AND INDEXING OF FUZZY COMPLEX SHAPES

Jinglan Zhang, Binh Pham and Phoebe Chen

Faculty of Information Technology

Queensland University of Technology

GPO Box 2434 Brisbane QLD 4001 Australia

(jinglan.zhang, b.pham, p.chen)@qut.edu.au

Abstract: At the conceptual stage of design, designers have only vague ideas of initial shapes that they gradually refine. A tool that supports conceptual design should capture such imprecise features but current CAD systems that are based on precise geometry and topology information cannot satisfy this requirement. This paper presents the representation, construction and display approaches for fuzzy complex shapes. We also discuss techniques for indexing and retrieving of fuzzy shapes in an object relational database with a fuzzy processing module.

Key words: Fuzzy shape, Fuzzy solid modelling, Fuzzy CAD system, and Fuzzy database.

1. INTRODUCTION

Design, especially in the conceptual stage, is intrinsically imprecise because of designers' vague thinking and incomplete initial information. The tools that support conceptual design should be human-oriented and able to capture such imprecise features. However, current Computer-Aided Design (CAD) systems that are based on precise geometry cannot satisfy this requirement. Although the parametric and feature-based modelling approaches (Shah and Mäntylä 1995) have made current CAD systems more flexible and meaningful, they still cannot cater for the imprecise characteristics of human because of the underlying precise representation. Since people have the ability to qualitatively identify, describe, and analyse shapes and spatial relationships by a natural language, it is desirable to use words which are related to shape, position or orientation to specify shape and

spatial relationship in top-down conceptual design stage. Fuzzy set theory (Zadeh and Kacprzyk 1999) is able to deal with linguistic terms, therefore is a promising approach for simulating human behaviour in describing, locating and reasoning process in shape design. Hence, we chose this approach to address the imprecise design problem in a solid modelling system. A fuzzy model can describe a system qualitatively using linguistic words whereas these words are associated to fuzzy quantities represented by fuzzy sets. So, fuzzy modelling is both a qualitative and quantitative scheme.

The applications of fuzzy sets in geometric modelling area are relatively new and most of existing work concentrates on fuzzy modelling of two-dimensional geometric objects (Anile et al. 2000; Buckley and Eslami 1997a; Buckley and Eslami 1997b; Martin 1994; Pham 2001; Rosenfeld 1998). We aim to investigate techniques for applying fuzzy theory into solid modelling systems, hence, we introduced the concept of a fuzzy shape as a set of shapes that looks similar yet slightly different. For example, the term *very sharp shape* with *squarish* traverse section potentially refers to a set of shapes from an object that looks like a pyramid to an exact pyramid (Figure 1(a)). A fuzzy component (fuzzy part or simple fuzzy shapes) is defined as a family of shapes which are represented by a set of parameters with fuzzy values. A fuzzy complex shape is the combination of any fuzzy parts under crisp or fuzzy spatial information through Boolean operations.

In previous papers, we have investigated the representation, specification, optimisation and management approaches of fuzzy components (Pham and Zhang 2000; Zhang et al. 2001; Zhang et al. 2002a; Zhang et al. 2002b). Although the superquadric model has relatively wide geometric coverage, its representation ability is still limited since it uses only one equation to represent shapes. Hence, a mechanism for creating complex shapes must be provided. This paper focuses on the construction of fuzzy complex shapes based on descriptive terms that are related to shape, spatial location and orientation, and the indexing and retrieving of fuzzy complex shapes in a fuzzy database. This paper is organised as follows. Section 2 describes the geometric representation of fuzzy complex shapes. The construction of fuzzy complex shapes based on Boolean operations is discussed in section 3. Section 4 gives an overview of the conceptual modelling of the fuzzy database for the management of fuzzy complex shapes. Section 5 discusses the indexing and retrieving of fuzzy complex shapes in a fuzzy database implemented in the framework of object relational database. Finally, the conclusions and future work are presented in the last section.

2. REPRESENTATION OF FUZZY COMPLEX SHAPES

Existing solid modelling methods, such as Constructive Solid Geometry (CSG) and Boundary Representation (B-Rep) (Mäntylä 1988), are based on precise representation of geometric objects, such as vertices, edges, surfaces as well as the exact topology relationship between them. These methods are good at supporting detail design where all design details must be represented exactly yet they are very weak to support conceptual design that is qualitative and uncertain. Qualitative models such as *geons* (Dickinson et al. 1993) cannot provide sufficient quantitative information that is required for shape specification and manipulation in CAD system. Superquadric models proposed by Barr (Barr 1981) use several shape parameters to quantitatively define a shape, yet each shape parameter is related to one quality of the shape such as squareness/roundness and size. Hence superquadrics is both quantitative and qualitative. Such a property makes superquadrics the ideal candidate for bridging qualitative conceptual shape design and quantitative detail design. It also makes it possible to introduce fuzzy set approach into CAD systems where fuzzy sets are used to represent design intents. We therefore choose superquadrics as the basic representation for 3D shape components, where a shape is represented by a set of shape descriptors and a set of shape parameters (Pham and Zhang 2000).

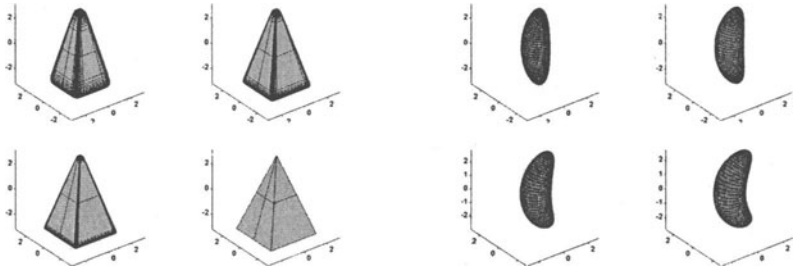
2.1 Representation of fuzzy primitives

Fuzzy primitives (or fuzzy shape components, simple shapes) refer to fuzzy shapes that can be represented by superquadrics directly. The detailed parameters and equations of superquadrics can be found in (Barr 1981; Barr 1984; Pham and Zhang 2000) and here we give only an abstract representation:

$$SQ = F(\varepsilon_1, \varepsilon_2, a_1, a_2, a_3, k_x, k_y, k_v, t), \quad (1)$$

where SQ is a superquadric shape, F is the function relating all these parameters; ε_1 and ε_2 are shape parameters which control the shape roundness and squareness along north-south direction and west-east direction respectively; a_1 is the overall size; a_2 and a_3 are ratio parameters which represent the ratio of the lengths of a shape along y and z axes to the length along x axis, and a_2 and a_3 control the relative dimension of the two cross sections of a superquadric shape; k_x and k_y control the tapering property; k_v controls the bending property; and t controls the twisting property. This representation is superellipsoids (Barr 1981) with deformation parameters and we still call them superquadrics for simplicity. When these parameters take fuzzy values, in particular, fuzzy real numbers, the resultant shape is a fuzzy shape which has many shape elements with variable membership degrees. Figure 1(b) shows a *slightly bent* fuzzy superquadric

shape with round traverse section, where the membership degrees vary from 1.0 to 0.6.



(a) Typical shape elements of a very sharp with squarish traverse section fuzzy shape.

(b) Typical shapes of a *slightly bent* with round *traverse section* fuzzy shape.

Figure 1. Elemental fuzzy shapes

Complex shapes can be generated by combining simple shapes through transformation (including translation, rotation, and scaling) and Boolean operations (intersection, union, difference) (Mäntylä 1988). We now discuss the representation of fuzzy complex shapes and their construction.

2.2 Representation of fuzzy spatial information

To support vague conceptual design, an effective method for specifying and reasoning of spatial relationship between objects is crucial to efficiently construct assemblies. Spatial information such as distance and orientation can be crisp, interval or fuzzy. We focus on the specification of spatial information using fuzzy sets because crisp and interval spatial information are special cases of fuzzy spatial information. We first define a set of spatial description words and then define fuzzy quantities associated with them.

Fuzzy representation of directions

In user interface level, we use words related to the *primary directions* along the three axes X, Y, and Z and *secondary directions* described according to their locations in XY, XZ, and YZ planes to describe directions. Primary directions include:

Primary Directions = {*Left, Right, Front, Back, Top, Bottom*},

and secondary directions include:

Secondary Directions = {*Top Right, Top Back, Top Left, Top Front, Bottom Right, Bottom Back, Bottom Left, Bottom Front, Right Front, Right Back, Left Front, Left Back*}.

The eight directions in the horizontal plane (XZ) can be represented by a circular neighbouring structure (Hernández 1994). We use the trapezoidal and triangular shape of membership functions to represent the fuzzy set values of directions as they are easy to manage. As the directions in each plane vary between 0° and 360° , the value range is fixed for any directions. We use 0° to 360° (beta angle) in XZ plane and 0° to 180° in YZ plane (gamma angle) (Figure 2) to describe directions because the product of these two intervals covers all directions in the three dimensional Euclidean space. Hence, we can equally partition the direction space into overlapped fuzzy regions, where the number of fuzzy sets are determined by the number of directions we aim to describe. We take eight directions in XZ plane and four directions in YZ plane as these are the mostly used directions in daily life.

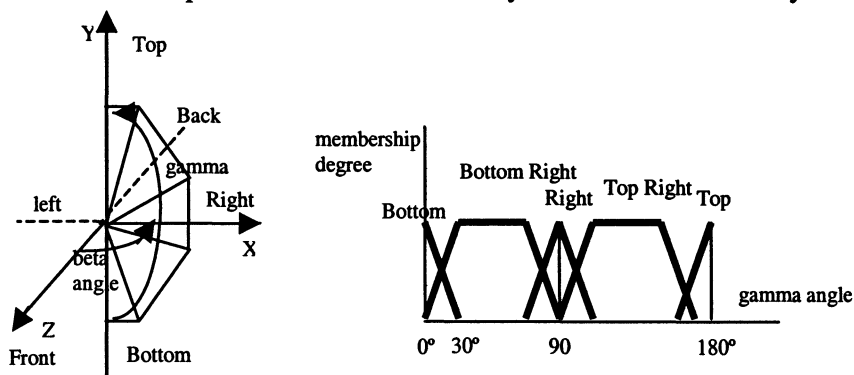


Figure 2. Directions in vertical plane (XY) and corresponding fuzzy quantities.

Fuzzy representation of distances

We assume all shape elements in a fuzzy set are defined in the same local reference system. The location of a fuzzy object is defined according to its reference point in a global coordinate system and a global transformation matrix can be formed for each shape components. Locations defined according to an existing object can be converted to the global coordinate system. Although distance can be specified linguistically such as *very close*, the domain of distance can hardly be fixed, hence it is very difficult to construct fuzzy sets for each word related to distance description. We therefore seek for the representation of fuzzy distance using fuzzy numbers, where each number is a fuzzy triangular norm. Since the orientations and distances are specified using fuzzy sets, the transformation matrix is also a fuzzy set. Take the translation transformation matrix for example. Assume

that we move an object right about 3 units, up about 2 units, and forward about 4 units, then the transformation of the object will be:

$$X = X + \tilde{3}, \quad Y = Y + \tilde{2}, \quad Z = Z + \tilde{4},$$

where $\tilde{2}$, $\tilde{3}$, and $\tilde{4}$ are fuzzy triangular numbers with centre point at the real number of 2, 3, and 4.

2.3 Representation of fuzzy complex shape

A complex solid can be constructed using a graph model such as B-Rep or a Boolean model such as CSG. Since a superquadric model has no representation of vertices and edges, we use Boolean operations to compose complex objects from simple components. After a Boolean operation, the resultant shape becomes another shape component and can be submitted to further Boolean operations. This constructive approach, called Constructive Solid Geometry (CSG), uses a CSG tree to represent the construction process and the structure of a complex shape. In the CSG tree, each leaf node is a superquadric shape component along with its transformation matrix while each branch node is a Boolean set operation, including union (+), intersection (*), and subtraction (-) (Figure 3). Since the CSG model can be described in concise text language, it supports the top-down design process and is suitable for introducing fuzzy theory into complex shape design.

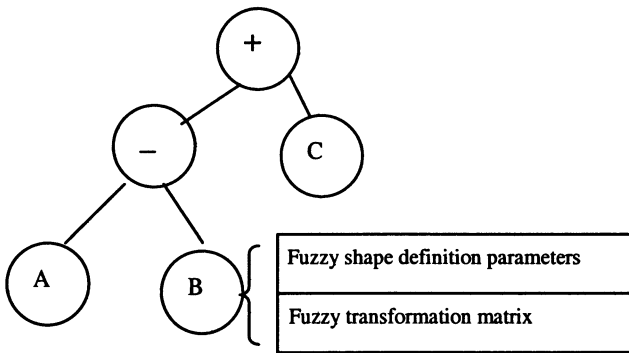


Figure 3. CSG representation of a composite shape

3. CONSTRUCTION OF FUZZY COMPLEX SHAPES

Once we know the size, position and orientation of two objects that have the same dimension (e.g. either 2D or 3D), we can perform Boolean operations on them to compose a complex shape. We discuss the Boolean

set operations of crisp superquadric shapes first, then move to the discussion of Boolean operations on fuzzy shapes.

3.1 Boolean operations of crisp superquadric shapes

In CSG models, Boolean set operations are essentially the point set membership classification problem (or boundary classification) which determines if a given point *in* or *out* of a given solid and separates a whole space into two half spaces (Mäntylä 1988). The point set classification consists of two parts: the point set classification against each primitive in the CSG model and the combination of all classification results. If a primitive is composed of many half spaces, the point set classification against a primitive includes the classification against each half space and the final recombination. Since superquadrics uses only one analytic equation to define whether a point belongs to an object, it is also a kind of half-space modelling approach. Hence, we can apply the point set classification approaches for other CSG models to superquadric models.

As a superquadric primitive has only one half space, the point set classification is conceptually easier than other types of half spaces. However, superquadrics are higher-order analytic representation of geometric shapes, the calculation of the intersection points of two superquadric shapes cannot be based on the analytic calculation of two surfaces. Instead, they can only be obtained by numeric methods which discretise a solid into many units of basic shapes such as cubes. We therefore convert the superquadric shape representation to regular, exhaustive space enumeration where the interested space is partitioned into equal interval distance along x , y , and z directions. Boolean operations are then performed based on the point-set classification of all points in the space against each superquadric shape.

This hybrid model based on CSG and space enumeration uses pure half-space concept for Boolean operations. The exhaustive enumeration approach is simple, general, but needs huge memory consumption and is only approximate. We employ this model for testing the methods for performing Boolean operations on fuzzy objects which will be discussed in the next subsection. The union and difference operations of two crisp shapes, a rectangular prism and a double pyramid, are calculated using this model (Figure 4).

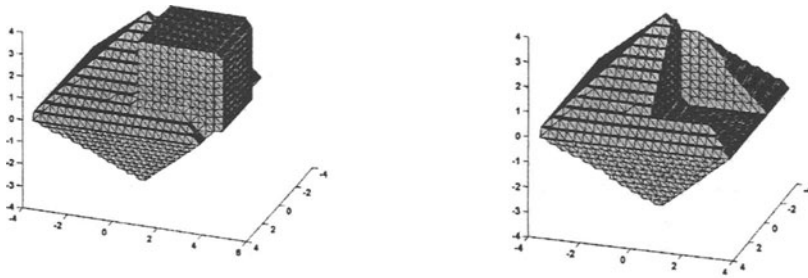


Figure 4. Union and difference operation of two crisp superquadric shapes.

3.2 Boolean operations of fuzzy superquadric shapes

A fuzzy shape is a set of crisp shapes controlled by fuzzy shape parameters (Figure 1). As the boundary of a fuzzy shape is not a crisp boundary but a boundary zone, the point set membership classification for Boolean operation cannot be performed using the conventional approach based on crisp half-space. One approach to solve this problem is to use the family of alpha-cuts representation of fuzzy sets and the interval principles (Klir and Yuan 1995). The family of alpha-cuts of a fuzzy set (Figure 5), which are fuzzy subsets with the membership grades of all elements equal to or greater than the value alpha, can uniquely represent a fuzzy set (Klir and Yuan 1995). Each alpha-cut can be represented by the two endpoints of this alpha level. Similarly, a fuzzy shape, which is a fuzzy set of shape elements, can be represented by its boundary shape elements in all alpha-levels. Each alpha-cut of a fuzzy shape includes an interval of crisp shapes between the two boundary shapes of this level. For example, a fuzzy circle with fixed centre location and fuzzy radius, can be represented by its lower and upper boundary circles at one alpha level (Figure 6(a)). The Boolean operations of two fuzzy shapes can therefore be mapped to the Boolean operations of the boundary shapes at each alpha-level. For each alpha level, only the boundary shapes are calculated and used for set operations. The resultant fuzzy shape is all shape elements between these boundary shapes at each alpha level. Using this approach, the set operations of fuzzy shapes are mapped to traditional Boolean operations of crisp objects.

The result of the set operation of two real number fuzzy sets is also a real number fuzzy set (Figure 5). The results of Boolean operations of two fuzzy shapes A and B are also fuzzy shapes. Assume $p = (p_1, p_2, \dots, p_n)$, $n = 9$, are shape parameters of a superquadric shape $S(p)$, then the alpha-cut of fuzzy shape A is:

$$A(\alpha) = \{S(p) \mid p_i \in P_i(\alpha), i = 1 \dots 9, \alpha \in [0,1]\}, \tag{2}$$

where P_i is the fuzzy value of parameter p_i . The membership degree of a crisp shape to a fuzzy set (shape) is the minimum of its membership degrees of all its parameters to the corresponding fuzzy values.

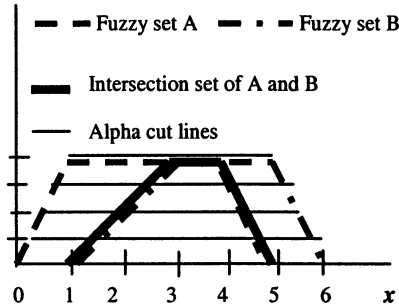


Figure 5. Fuzzy set intersection based on family of alpha-cuts.

According to the interval arithmetic principle and the family of alpha-cuts representation of fuzzy sets, the set operations of the two fuzzy shapes A and B , denoted by $A \circ B$, can be represented as:

$$A \circ B(\alpha) = \{a \circ b \mid a \in A(\alpha), b \in B(\alpha)\}. \tag{3}$$

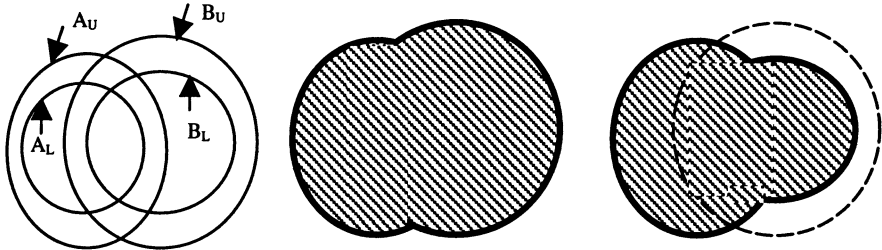
Replacing each interval using their two end points, we can derive the following equation for calculating the interval of $A \circ B$:

$$[A \circ B(\alpha)^-, A \circ B(\alpha)^+] = [\min\{a \circ b : a \in \{A(\alpha)^-, A(\alpha)^+\}, b \in \{B(\alpha)^-, B(\alpha)^+\}, \max\{a \circ b : a \in \{A(\alpha)^-, A(\alpha)^+\}, b \in \{B(\alpha)^-, B(\alpha)^+\}\}], \tag{4}$$

where $A(\alpha)^-$ and $A(\alpha)^+$ represent the lower and upper bound of an interval for an alpha level of a fuzzy set A . The corresponding notions for set B and $A \circ B$ have the same meaning.

Since a shape has many parameters and each parameter has two boundary values at each alpha level, the boundary shapes of each alpha level are all shapes formed by the combination of all these boundary values. The two boundary shapes that participate in the set operation are selected according to their spatial occupancy (volume). The resultant boundary shapes of a set operation at an alpha level are again selected according to their spatial occupancy. The result of set operations of two fuzzy shapes is a new fuzzy complex shape and can attend further set operations. We take the union of two fuzzy circles as an example to show how the method based on the family of alpha-cuts works. Assume two fuzzy circles A and B are bounded, closed, and regular point set. They are located in their precise location and their radii take fuzzy values. At each alpha level, the radius for each circle has two lower and upper boundary values and thus defines two crisp circles denoted by $A_L, A_U, B_L,$ and B_U (Figure 6(a)). The resultant fuzzy shape of the union operation on these two fuzzy shapes is bounded by

four crisp shapes: $A_L B_L$, $A_L B_U$, $A_U B_U$ (Figure 6(b)), and $A_U B_L$ (Figure 6(c)). The resultant shapes, $A_U B_U$ and $A_L B_L$, are selected as the boundary shapes for the union of the two fuzzy shapes at the corresponding alpha level because they have the minimum and maximum areas.



(a) Two fuzzy circles bounded by their boundary circles at an alpha level. (b) Union of the two upper boundary circles ($A_U B_U$). (c) Union of the an upper circle and a lower circle ($A_U B_L$).

Figure 6. Union of two fuzzy circles at one alpha level. (Dashed lines for reference only).

4. CONCEPTUAL MODEL OF THE FUZZY SHAPE DATABASE

A Solid Modeling (SM) system contains the geometry and topology information of modelled objects as well as a large amount of non-geometric information such as material properties. We focus on the management of geometry and topology information.

In our system, a fuzzy shape component is represented by a set of linguistic descriptors and a set of parameters with fuzzy values. A complex shape (or assembly) consists of shape components and subassemblies which are the combination of a group of components and/or other subassemblies. The order and the way in which shape components and/or subassemblies are assembled is modelled in a CSG tree (Figure 3). A CSG tree can be stored in different levels where leaf nodes and sub-trees can be used many times.

We use the CSG model as the main modeller for structural representation and the regular space enumeration model as the secondary modeller for Boolean operations and the calculations of geometric properties. The CSG model is internally converted into space enumeration. These two models coexist to avoid the conversion effort and support quick display. When the CSG model changes, the corresponding space enumeration model should be upgraded associatively. We focus on shapes that were created by a set of Boolean operations on a set of predefined fuzzy shape components according to specified fuzzy spatial information. As the

dataset for the space enumeration model is very large, it is calculated after the construction process and only the dataset for the resultant shape is stored.

Relational database management systems (RDBMS), which use tables as basic units, are very popular because of their simple structure, quick response time, and firm foundation on relational mathematics. Hence, many existing SM systems have interfaces to one or more RDBMS such as ORACLE (LaCourse 1995). As a RDBMS uses flat tables to represent data, it meets difficulties in handling large assemblies because of the complex semantics associated with 3D models and complex nested entities. An object-oriented database management system (OODBMS), which is a collection of data and methods, can capture the complex semantics and structures, but it has no firm theory foundation and no easy-to-use query language. Object relational database systems (ORDBMS), a hybrid of RDBMS and OODBMS, can handle complex data and still can use the knowledge and experience obtained from RDBMS (Connolly et al. 1999). We therefore chose the ORDBMS to construct the shape database and developed a fuzzy processing module for handling fuzzy values. We call this shape database a fuzzy database because the attribute values are fuzzy and the query processing is based on fuzzy theory. The basic structure of an assembly database, including the entities and attributes as well as relationships between tables are shown in Figure 7. The Descriptors, Parameters, and Transformation matrix take complex array structure and have fuzzy values. A fuzzy set is treated as an object with data and corresponding methods for processing the data.

5. INDEXING AND QUERY OF FUZZY SHAPES

A CAD database allows the attachment of attributes to a solid object and uses these attributes to retrieve objects. Indexing is the task aimed at finding a set of attributes which can identify an object in a relation. In the fuzzy shape database, every shape has a unique, system-generated identification code and is employed as the primary key to support quick searching. Database users often want to retrieve data based on one or more values which are not the primary key. Hence, secondary keys need to be specified to index shapes for different searching purposes. The combination of shape name and version is usually employed as an alternative index. Hence, we also provide this index to conform to the general systems. As conventional CAD systems are based on exact geometric information and qualitative information is not readily available, higher-level semantic shape searching is almost impossible. However, in the fuzzy solid modelling system, fuzzy shape components are specified using a set of linguistic words such as *very*

round or *extremely bent*. This set of shape descriptors can be used to index shape and support semantic-based shape query and retrieval. We therefore derive a set of descriptors from the characteristics of superquadrics, including *squariness*, *roundness*, *bevelness*, *pinchness*, *bendness*, *taperness*, and *twistness*, to index shape components (Zhang et al. 2002b).

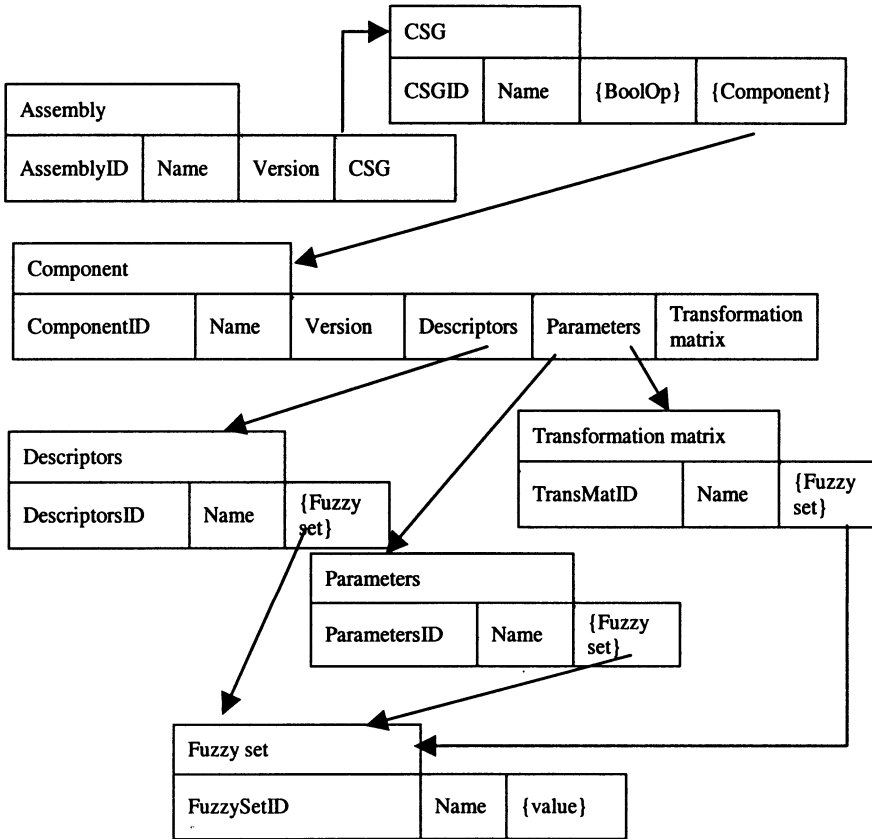


Figure 7. Database structure for fuzzy shapes

In a conventional database, a data item is self-indexed. For example, the query “find people whose salary is over \$2000” can always find the satisfied people if suitable tuples exist. Hence, users just need to specify attributes for indexing and do not need specify a special measure to evaluate the index. However, in a fuzzy database, an attribute may take a fuzzy value and it cannot index itself because a fuzzy set is a range value with different membership degrees. Hence, a measure determining whether a fuzzy datum satisfy a fuzzy condition must be defined. We use the *similarity measure* which is the minimum ratio of the cardinal (area) of the intersection of two fuzzy sets to their own cardinals. Once we define this measure, we can perform a fuzzy query such as “find a slightly bent shape” from a set of

shapes having the attribute bendness and taking fuzzy values. If the attribute value is crisp, the similarity measure degenerates to the membership degree to which the crisp value belongs to the condition fuzzy set.

A complex shape is the result of Boolean combinations over fuzzy shape components but the properties of the original components may be eliminated after Boolean operations. Therefore, we cannot query a complex shape by descriptors directly. However, a complex shape is maintained by a CSG tree, including a set of components and their corresponding transformation matrices. So, we can query a complex shape according to its construction process. Although the same object may have many different construction processes and hence have many different CSG models, one CSG model corresponds to only one solid shape. In addition, experienced designers tend to use a set of rules (experiences) to guide the construction process, hence, for the same complex shape, different designers will have the same construction process in many cases. The support of CSG-based query is therefore desirable. CSG model is concise and can be represented using text such as “ $C = A + B$ ”. Such a property along with the linguistic description of shape components and spatial information make it possible to formulate a query using words. For example, the query “find a shape which is the union of a very bent cylinder and a very sharp pyramid” can be performed.

The CSG-based searching is essentially a graph matching problem. Graph matching techniques, such as the Labelled Graph Matching (LGM) which finds the optimal correspondence between two graphs (Shams and Brady 2001) or the Largest Common Subgraph detection (LCSG) which finds the largest collection of objects with the same relationship in both images (Shearer et al. 2001), have been applied mostly to image processing area. As the CSG tree representation of a complex shape is also a graph, these techniques can also be used to support structure-based object searching. To support the graph matching in an SM database, we first translate the text-based query into a CSG tree, then we can use the approach proposed by Shearer et al (Shearer et al. 2001) to find the largest common subgraph of the condition CSG and a datum CSG. As the shape component and the transformation matrices contain fuzzy data, this matching process will also use the similarity measure of fuzzy sets to calculate the similarity of two CSG trees. The detailed equation for calculating the similarity of two graph without fuzzy data can be found in (Shearer et al. 2001). The similarity calculation of two CSG trees with fuzzy shape components and fuzzy transformation matrices needs to calculate the similarity of two fuzzy components or two fuzzy numbers and corresponding algorithms can be found in (Zhang et al. 2002b). This CSG-based index can find complex shapes according to their construction process, spatial information and shape components. The similarity calculation of two CSG graphs can also be weighted and the weightings of these three factors can be predefined.

A query interface which supports flexible, uncertain understanding can better grasp a designer's thinking and will be easier to be accepted. Hence, the CSG-based shape indexing and the natural language-based query will be very helpful and complementary to the conventional shape indexing and query approaches.

6. CONCLUSIONS AND FUTURE WORK

The fuzzy representation of shapes, distance, and orientation as well as the set operations on fuzzy complex shapes based on the technique of family of alpha-cuts make it possible for a solid modelling system to handle imprecise information. The fuzzy database model, which is implemented within the framework of an object relational database with the incorporation of a fuzzy processing module, makes it possible to manage fuzzy shapes. However, fuzzy computation is time consuming because of the higher order of superquadrics and the large amount of data processing for fuzzy sets.

The construction of fuzzy complex shapes based on the family of alpha-cuts approach has been tested in a prototype system. A prototype system for the management of fuzzy components has been constructed. Hence, future work includes extending the fuzzy database to cater for the management of fuzzy complex shapes, refining and optimising the proposed approach and investigating new approaches for fuzzy shape design and management.

REFERENCES

- Anile, A. M., Falcidieno, B., Gallo, G., Spagnuolo, M., and Spinello, S. (2000). "Modeling Uncertain Data With Fuzzy B-Splines." *Fuzzy Sets and Systems*, 113, 397-410.
- Barr, A. H. (1981). "Superquadrics and Angle-preserving Transformations." *IEEE Computer Graphics & Appl.*, 1, 11-23.
- Barr, A. H. (1984). "Global and Local Deformations of Solid Primitives." *Computer Graphics*, 18(3), 21-30.
- Buckley, J. J., and Eslami, E. (1997a). "Fuzzy Plane Geometry II: Circles and Polygons." *Fuzzy Sets and Systems*, 87(1), 79-85.
- Buckley, J. J., and Eslami, E. (1997b). "Fuzzy Plane Geometry I: Points and Lines." *Fuzzy Sets and Systems*, 86, 179-187.
- Connolly, T. M., Begg, C., and Strachan, A. (1999). *Database Systems*, ADDISON-WESLEY, England.
- Dickinson, S., Biederman, I., Pentland, A., Eklundh, J.-O., Bergevin, R., and Munck-Faiwood, R. (1993). "The Use of Geons for Generic 3-D Object Recognition, ." *Proc. Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, Chambery, France, 1693-1699.
- Hernández, D. (1994). *Qualitative representation of spatial knowledge*, Springer-Verlag, Berlin ; New York.
- Klir, G., and Yuan, B. (1995). *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, New Jersey.

- LaCourse, D. E. (1995). *Handbook of solid modeling*, McGraw-Hill, New York.
- Mäntylä, M. (1988). *An Introduction to Solid Modeling*, Computer Science Press, Inc., U.S.A.
- Martin, R. R. (1994). "Modelling inexact shapes with fuzzy sets." Proc. of CSG'94, J. R. Woodwark, ed., Information Geometers Ltd., Winchester, 73-98.
- Pham, B. (2001). "Representation of Fuzzy Shapes." Proc. of International Workshop on Visual Form 4, C. Arcelli, ed., Springer-Verlag Berlin Heidelberg, LNCS, 2059, 239-248.
- Pham, B., and Zhang, J. (2000). "A Fuzzy Shape Specification System to Support Design for Aesthetics." *Soft Computing in Measurement and Information Acquisition*, L. Reznik, ed., Physica-Verlag, Heidelberg, in print.
- Rosenfeld, A. (1998). "Fuzzy geometry: An updated overview." *Information Sciences*, 110, 127-133.
- Shah, J. J., and Mäntylä, M. (1995). *Parametric and Feature-Based CAD/CAM*, John Wiley & Sons, Inc., NY, U.S.A.
- Shams, L. B., and Brady, M. J. (2001). "Graph matching vs mutual information maximization for object detection." *Neural Networks*, 14, 345-354.
- Shearer, K., Bunke, H., and Venkatesh, S. (2001). "Video indexing and similarity retrieval by largest common subgraph detection using decision trees." *Pattern Recognition*, 34, 1075-1091.
- Zadeh, L. A., and Kacprzyk, J. (1999). "Computing with Words in Information/Intelligent Systems 1: Foundations." *Studies in Fuzziness and Soft Computing*, L. A. Zadeh and J. Kacprzyk, eds., Physica-Verlag Heidelberg.
- Zhang, J., Pham, B., and Chen, P. (2001). "Fuzzy Genetic Algorithms Based on Level Interval Algorithm." Proc. of the 10th IEEE Intl. Conf. on Fuzzy Systems, Melbourne, Australia.
- Zhang, J., Pham, B., and Chen, P. (2002a). "A Fuzzy Genetic Algorithm for Creative Shape Design." Proc. of the Fourth International Symposium on Tools and Methods of Competitive Engineering (TMCE 2002), accepted, Wuhan, China.
- Zhang, J., Pham, B., and Chen, P. (2002b). "Indexing and Retrieval of Fuzzy Shapes." *Fuzzy Logic: A Framework for the New Millennium*, V. Dimitrov and V. Korotkich, eds., Physica-Verlag, Heidelberg, 240-250.

BIOGRAPHIES

Jinglan Zhang is a PhD student in Information Technology at the Queensland University of Technology, Australia. She received her BS and MS degrees in CAD/CAM from Beijing University of Aeronautics and Astronautics, China. Her current research interests include applications of fuzzy logic and evolutionary computation in CAD systems.

Binh Pham is currently the Professor and Director of Research in the Faculty of Information Technology at the Queensland University of Technology, Brisbane, Australia. Her research interests include computer graphics, multimedia, CAD, image analysis, and intelligent systems.

Phoebe Y. Chen is currently the Lecturer in the Faculty of Information Technology at the Queensland University of Technology, Brisbane, Australia. She completed her PhD in the field of Information Systems and Computer Science at the University of Queensland. Her research interests include scientific visualization, multimedia, bioinformatics, database technologies and web-based intelligent systems.