S3: Similarity Search in CAD Database Systems

Stefan Berchtold, Hans-Peter Kriegel

Institute for Computer Science, University of Munich
Oettingenstr. 67, D-80538 München, Germany
e-mail: {berchtol,kriegel}@informatik.uni-muenchen.de

Abstract

S3 is the prototype of a database system supporting the management and similarity retrieval of industrial CAD parts. The major goal of the system is to reduce the cost for developing and producing new parts by maximizing the reuse of existing parts. S3 supports the following three types of similarity queries: query by example (of an existing part in the database), query by sketch and thematic similarity query. S3 is an object-oriented system offering an adequate graphical user interface. On top of providing various state-of-the-art algorithms and index structures for geometry-based similarity retrieval, it is an excellent testbed for developing and testing new similarity algorithms and index structures.

1 Introduction

The S3 system is the prototype of a database system for the management and similarity search of industrial parts. The key idea of S3 is to allow a user to retrieve similar parts specified by geometry and thematic attributes such as functionality or material. In S3, parts are described by a two-dimensional polygonal contour and thematic attributes. In contrast to other systems, S3 retrieves similar parts not using a single similarity measure but offering the user a collection of algorithms. Since S3 does not define similarity a priori, the application of S3 is not restricted to a special domain.

S3 has been devoloped in cooperation with a major European supplier of the car manufactoring industry. Our industrial partner produces parts called 'clips' which are used for holding and joining components in a car. Since, in general, 'clips' are made of plastic and produced by injection molding, the variety of shapes is nearly unlimited. The number of parts manufactured by the company is quite large since many different parts are necessary for each car model. The goal of the cooperation is to reduce the cost of producing new parts by maximizing the reuse of existing parts. Important for parts to be reusable is that the new part coincides in some detail with previously designed parts which are stored in the part database of the company. If similar parts can be found, the cost for designing and manufacturing a new part can be reduced considerably since the time for designing the part is shortened and it is possible to reuse the expensive machinery for manufacturing the part. Finding all similar parts for a given query part is therefore the key to cost reduction. The potential for cost savings of the company is in the range of 1 to 5 million Dollars a year.

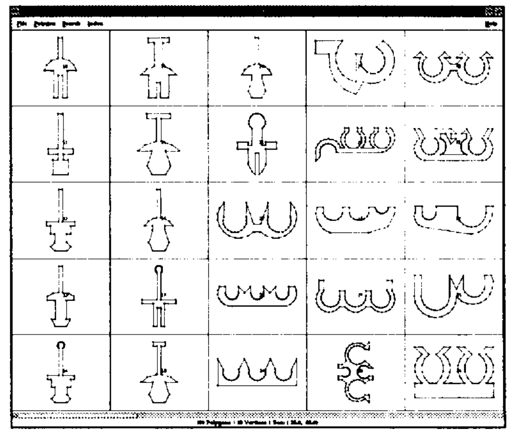

Figure 1: The S3 System (Part Database)

In addition to reducing cost, S3 is a testbed for new similarity algorithms. Therefore, various algorithms and index structures for the geometry-based similarity search are integrated in the system. Thus, S3 gives an overview of the research activities in the field of geometry-based similarity search and enables the user to compare different techniques. From the insights we got from that comparison, we developed some new similarity algorithms within S3 ([BKK 97], [BKK 97a]). Furthermore, S3 is a good testbed for index structures. Since in S3 each similarity algorithm may be combined with an adequate index structure, we were able to compare the efficiency of the implemented index structures on indexing high-dimensional feature data. This comparison motivated the developement of a new index structure for high-dimensional data, the X-tree [BKK 96], which we integrated into S3.

2 Overview

In **S3**, parts are described by two-dimensional simple polygons and related thematic attributes.The system consists of three major components. Each component has a graphical user interface and interacts with the user and the other components. The components are:

- the part database
- similarity indices
- result visualization and explanation component

The advantage of this design is the good extensibility of an object-oriented system: similarity algorithms and index structures may be added to the **S3** system without changing other components.

Several similarity algorithms were implemented in **S3**. All these algorithms use a feature transformation to determine the amount of similarity. Similarity indices in **S3** are instances of a class of a similarity algorithms, managing feature data located on secondary storage. Therefore, similarity indices are described by a tupel (database, algorithm, parameter).

When the system finishes the processing of a query, the result of the query is presented graphically. This is done for two reasons: first, users can scan the whole set of answers very quickly if the answers are presented graphically. Second, users want to know not only *if* the parts are similar but also *why* the parts are similar. Therefore, **S3** depicts not only the set of answers but also the according features and the query specification in one window.

The implementation of the **S3** system is based on an object oriented design and has been implemented in C++ using X11 / OSF Motif. It is running under HP-UX as well as Linux.

3 Similarity Algorithms

As our experience shows, the user's notion of similarity is rather subjective. This means that there does not exist a *single* notion of similarity but *many* notions of similarity. Moreover, similarity depends not only on the application domain but also on the user. This, however, implies that a system such as **S3**, which is supposed to be independent from a special application domain must provide a way to let the users specify their notion of similarity. In **S3**, the users may do this by describing the properties of their notion of similarity. An example of such a property is the invariance against some kind of affine transformations such as translation and rotation. **S3** maps these properties to a collection of similarity algorithms and corresponding similarity indices provided by the system. **S3** currently supports the following similarity algorithms:

- *Mehrotra-Gary* [MG 93]: Mehrotra and Gary proposed in [MG 93] an algorithm for retrieving similar contours. We implemented a slightly modified version of the algorithm in **S3**. The algorithm extracts regions of constant length from the polygon of the part and determines the angles and lengths of the segments of the region. After a normalization step, a feature vector is build from this information. We insert the feature vector into one of the index structures listed below.
- *Angular profile* [BMH 92]: The algorithm samples a set of points with an equal distance from the contour of the CAD-object. Feature vectors are computed using the angles built from neighboring sample points. Since, for polygons, most angles are zero and in order to make the algorithm more robust, we slightly smooth the angles using a filter.
- *Section Coding* [BKK 97a]: The basic idea of section coding is to determine the surrounding circle of a polygon, to divide this circle into k sectors, and determine the proportions of area of the polygon which is inside each sector.
- *Fourier-based* [BKK 97]: This approach focuses on a specific similarity problem: the partial similarity. Partial similarity means that polygons are similar if they contain similar subregions. In [BKK 97], we presented a new approach to solve the partial similarity problem using extended feature objects instead of feature vectors.

The index structures currently supported by the **S3** system are the R*-tree [BKSS 90], the X-tree [BKK 96] as well as linear search of the database.

4 Query processing

S3 supports three basic query-types:

- query by example
- query by sketch
- thematic query

Query specifications of all these query types are entered using a graphical user interface. The user chooses query by example if he wants to find all parts which are similar to a known part in the database. For example, if the user searches for a special cable binder: he chooses a cable binder from the part database, performs a query by example and **S3** selects all similar cable binders. If the user is not too familiar with the content of the part database (e.g. he does not know which part in the database is a cable binder) he draws a sketch of the part in the polygon editor of **S3** and **S3** tries to find all parts similar to the drawn sketch. Depending on the similarity algorithm, this sketch may describe either the whole part (global sketch) or part of the polygon's contour (partial sketch). Additionally, the user may select similar parts by specifying thematic attributes. He specifies some ranges of attributes and **S3** selects the set of all parts having attributes within the specified ranges. If this set is empty, **S3** retrieves the k parts which are closest to the specified ranges.

As a result of each query, **S3** gives an algorithm dependant explanation of the amount of similarity between a part in the answer set and the query specification.

5 Examples

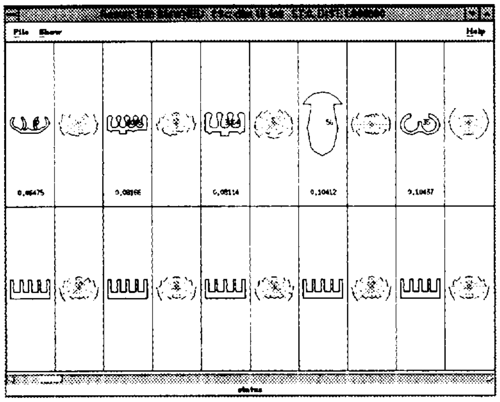

Figure 2:Explanation Component (Section Coding): The figure presents an example of an explanation component in S3. Two columns belong to one part in the answer set. In the lower half of the window we see the query specification, in the upper half the answer set. The original part is presented on the left side, the according feature representation on the right side. The numbers present the amount of similarity calculated by S3.

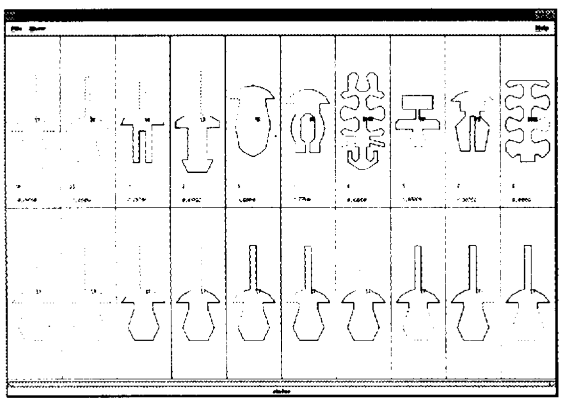

Figure 4: Explanation Component (Mehrotra-Gary): The figure shows another example for an explanation component in S3. Each column belongs to one part in the answer set. In the lower half of the column we see the query specification, in the upper half the answer set. The upper numbers show the length of the similar region, the lower number the amount of similarity inside this region. Answers are sorted by the length of the similar region.

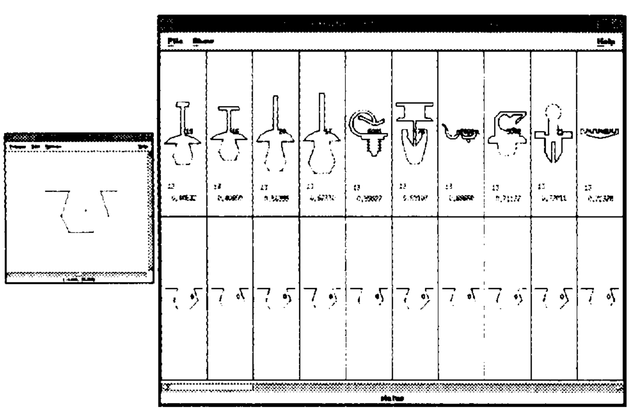

Figure 2:Query by sketch (Angular Profile): In this figure, an example for a query by partial sketch is presented. In the left part of the figure the user draws a sketch of an interesting detail. S3 selects parts which include a detail similar to the drawn detail. The numbers show the amount of similarity calculated by S3.

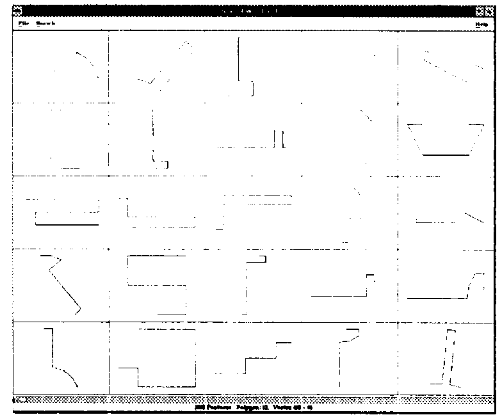

Figure 3: Index Visualization (Mehrotra-Gary): This figure shows a visualization of a set of features in a 9-dimensional feature space. Each feature in the figure describes a region containing 5 segments of a polygon in the database. Odd dimensions in this feature-space describe lengths of the segments whereas even dimensions describe angles between the segments.

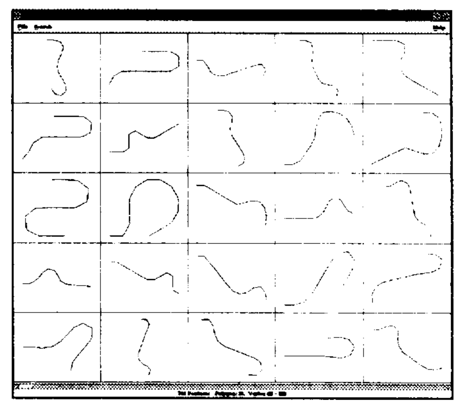

Figure 7: Index Visualization (Angular Profile): This figure presents a visualization of a set of features in a 7-dimensional feature space. Each feature in the figure describes a region containing 8 sample points of a polygon in the database. The dimensions in this feature-space describe angles between the line segments defined by the sample points.

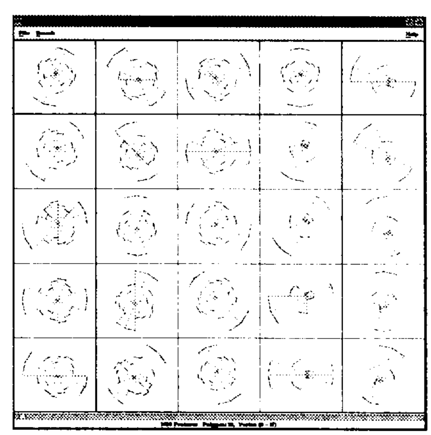

Figure 5: Index Visualization (Section Coding): The figure depicts a visualization of a set of features in a 16-dimensional feature space. The feature transformation is based on a circle surrounding the polygon. The i-th dimension of a feature describes the fraction of the area of a polygon which is inside the i-th circle sector.

References

[BBB+ 96]Berchtold S., Boehm C., Braunmueller B., Keim D. A., Kriegel H.-P.: *'Fast Parallel Similarity Search in Multimedia Databases'*, Proc. ACM SIGMOD Int. Conf. on Management of Data, Tuscon, Az, 1997.

[BKK 96]Berchtold S., Keim D., Kriegel H.-P.: *'The X-tree: An Index Structure for High-Dimensional Data'*, 22nd Conf. on Very Large Databases, 1996, Bombay, India, pp. 28-39.

[BKK 97]Berchtold S., Keim D., Kriegel H.-P.: *'Using Extende Feature Objects for Partial Similarity Retrieval'*, accepted for publication in VLDB Journal, 1997.

[BKK 97a]Berchtold S., Keim D., Kriegel H.-P.: *'Section Coding: Ein Verfahren zur Ähnlichkeitssuche in CAD-Datenbanken'*, GI-Fachtagung Datenbanken in Büro, Technik und Wissenschaft, (BTW), Ulm, Germany, 1997, (in German).

[BKSS 90]Beckmann N., Kriegel H.-P., Schneider R., Seeger B.: *'The R*-tree: An Efficient and Robust Access Method for Points and Rectangles'*, Proc. ACM SIGMOD Int. Conf. on Management of Data, Atlantic City, NJ, USA, 1990, pp. 322-331.

[BMH 92]Badel A., Mornon J. P., Hazout S.: *'Searching for geometric molecular shape complementary using bidimensional surface profiles'*, Journal of Molecular Biology, Vol. 10, December 1992, pp. 205-211.

[MG 93] Mehrotra R., Gary J. E.: *'Feature-Based Retrieval of Similar Shapes'*, Proc. 9th Int. Conf. on Data Engineering, Vienna, Austria, 1993, pp. 108-115.