

RETRIEVING MOULD DRAWINGS BY CONTENT

Manuel J. Fonseca, Alfredo Ferreira, Joaquim A. Jorge
 Department of Information Systems and Computer Engineering
 INESC-ID/IST/Technical University of Lisbon
 R. Alves Redol, 9, 1000-029 Lisboa, Portugal
[mfj@inesc-id.pt](mailto:mjf@inesc-id.pt), alfredo.ferreira jr@inesc-id.pt, jorgej@acm.org

Abstract

While drawing a mould for a new design, draftspeople often consult previous projects, searching for something similar. This way they can use solutions already developed in the past, thus reducing the time required to design a new mould. However, current systems do not make this an easy task.

We developed a new approach for content-based retrieval that supports automatic indexation of technical drawing databases and later recovering information by using structural descriptions of mould designs. In this paper, we describe a prototype that is able to retrieve technical mould drawings by content. The prototype allows searching for drawings using sketches, a digitized drawing or a combination of both. Queries can be refined by sketching on a digitized drawing or on one of the results interactively.

Preliminary usability tests revealed that users were generally pleased with the concept of using sketches as the main query mechanism.

Introduction

The widespread use of CAD tools in the mould industry during the past two decades made the creation and edition of drawings easier, increasing the number of mould drawings available in electronic format. Even though reusing drawings often saves time, manually searching for them is usually slow and problematic, requiring designers to browse through large and deep file directories or navigate a complex maze of menus and dialogs for component libraries. Moreover, current CAD systems provide only conventional database queries or direct-manipulation mechanisms to retrieve past technical drawings.

Most solutions to this problem use textual databases to organize the information [1][2]. These classify drawings by keywords and additional information, such as designer name, style, date of creation/modification and a textual description. However, solutions based on textual queries are not satisfactory, because they force designers to know in detail the meta-information used to characterize drawings. Worse, these

approaches also require humans to produce such information manually when cataloguing data.

In contrast to the textual organization, we propose a system to index and retrieve mould drawings based on content. Our approach uses a visual classification scheme based on shape and spatial relationships, which are better suited to this problem, because they take advantage of users' visual memory and explore their ability to use sketches as a query mechanism.

In the remainder of this paper, we first present a brief overview of the overall system architecture. In the next section, we present our prototype to retrieve technical mould drawings. The next two sections explain how we combine sketches and digitized drawings to create queries. Then we describe the database builder and how it is used. Finally, we conclude the paper by presenting results from evaluation tests, conclusions and discuss directions for future work.

Architecture Overview

An architecture for content-based retrieval usually can be divided in two parts. The **Classification** module analyses drawings and maps their features onto numeric descriptors. The **Retrieval** module compares queries to a database of drawing descriptors to produce a set of candidate results [3]. Fig. 1 presents a block diagram of our system architecture, illustrating these parts.

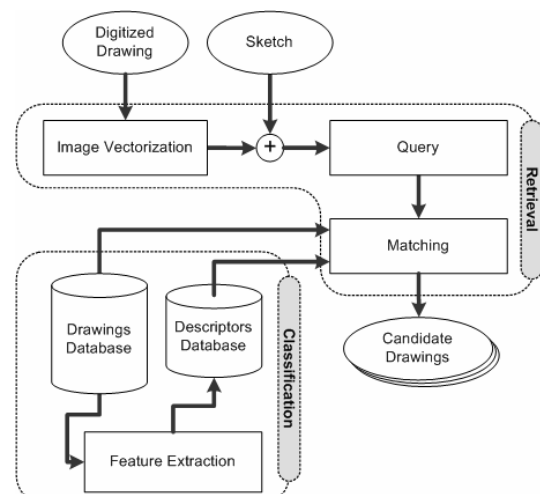


Fig. 1. System Architecture

The classification module extracts geometric and topological information from drawings and creates feature vectors that are then inserted into a multidimensional indexing structure.

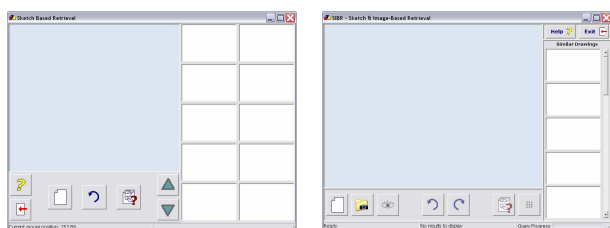
The retrieval module comprises different components. Image vectorization converts digitized drawings to vector format by applying computer vision algorithms. The query component extracts features (the same extracted during classification) from sketches, vectorized drawings or from a combination of both and creates descriptors. Finally, the matching component compares descriptors from the query to those stored in the logical database, yielding a set of similar drawings. Since we use an indexing structure, we only need to compare the query to a set of potential results and not to the entire database.

We developed two prototypes to exercise the two main modules of our architecture. The *Database Builder*, which implements the classification component, classifying existing drawings and producing a logical database with descriptors, and the *Mould Retrieval* prototype, which processes queries (sketches or digitized drawings) and matches candidates against the query, selecting and displaying results. In the next sections we describe in detail these prototypes.

Mould Retrieval Prototype

Using the architecture described before, we developed an application to retrieve technical drawings from the mould industry, through hand-drawn queries¹ [3]. Our system retrieves sets of technical drawings by contents from large databases.

This prototype uses a calligraphic interface to allow specifying queries through sketches. Moreover, it is able to convert raster images into vector drawings and use these as queries. Finally, it incorporates a matching component that compares query descriptors to a logical database describing existing drawings.



First version Second version

Fig. 2. Screen-shots from our prototype.

We have developed two versions of the present prototype. The first was implemented based on the requirements collected during task analysis [5]. The user interface was divided in three areas; a sketching area to draw queries, a result area to display similar drawings and a button area with control buttons, as depicted in Fig. 2 (left).

The second version of the prototype took into account results from the user evaluation performed to the first version and described below. The main changes at the user interface level included an increase of the drawing area size, reducing the number of results shown to five and decreasing the size of the buttons.

We also added some more functionality to this new prototype, namely using digitized drawings as queries, querying by example using one of the results and editing operations, such as deleting or selecting parts of a drawing to use as query.

We will now describe with detail the three possible ways of specifying queries, supported by the second version of our prototype.

Querying Using Sketches

Our application allows users to specify queries using hand-drawn sketches. They can draw an approximate representation of the desired mould part. In response the system returns a set of candidate drawings similar to the query.

Those queries are used in the scenario where users receive a sample part or its drawing and want to search for moulds of similar objects. To find them, the users often sketch a top view of the desired object.

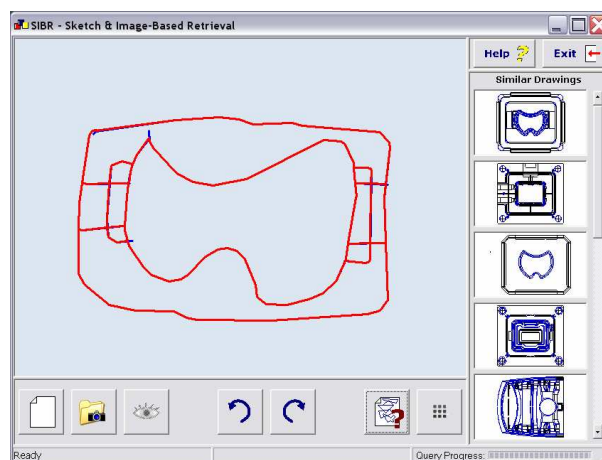


Fig. 3. Sketched query and returned results.

Fig. 3 depicts a screen-shot of our application, where we can see on the left the sketch of the top view of a part and on the right the correspondent query results. These are ordered from top to

¹ More information can be found in <http://immi.inesc-id.pt/projects/sbr/>

bottom, with the most similar on top. From the figure we can see that a rough approximation of the drawing, representing the more relevant features, yields a good set of results with the sought drawing in first place.

However, if users consider the results unsatisfactory, but believe there is a drawing in the returned set which is reasonably similar to the wanted one, it possible to drag-and-drop that drawing to the sketching area. They can then compose a new query by editing the dragged drawing.

Querying Using Drawings on Paper

Another scenario for our application arises when users have a printed hardcopy of the drawing and want to find similar results stored in the database. In this situation they only need to scan the drawing, wait for the vectorization step to finish and then submit the resulting drawing to the system as a query.

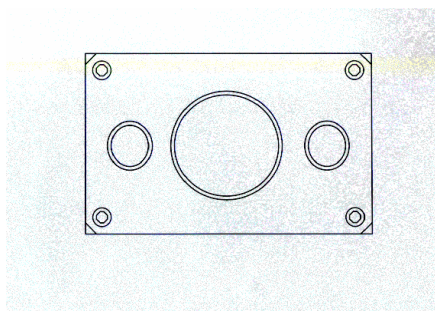


Fig. 4. Original digitized drawing.

Fig. 4 was scanned from a printed hardcopy of a drawing, while Fig. 5 shows an approximate vectorized version of the drawing, obtained by applying a set of computer vision algorithms. We tuned these algorithms to minimize user intervention and to privilege performance in detriment of accuracy. This way, we end up with a fast vectorization method suitable for interactive use. Vectorizing the drawing shown in Fig. 4 took about sixty seconds on an Intel Pentium III @1GHz, with 512MB of RAM, running Windows XP.

Although this method entails some loss in accuracy, the final vector drawing is very acceptable since the main features are kept (see Fig. 5). Furthermore, our matching approach does not have any problem in dealing with drawings of this quality, since it is prepared to accept sketches.

Fig. 5 depicts the vectorized drawing and the correspondent results after submitting it as a query to our retrieval system. We can see that the wanted result was returned in first place.

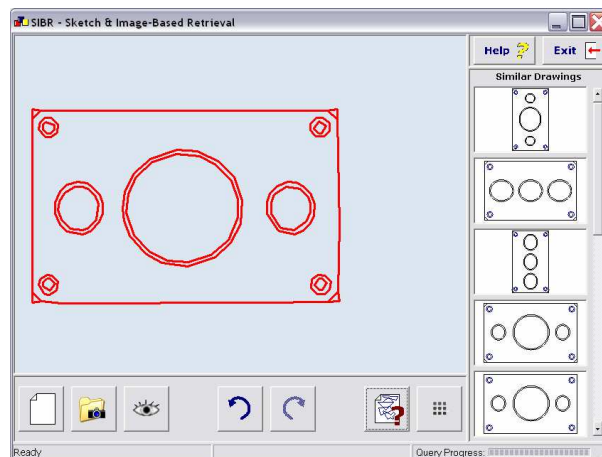


Fig. 5. Vectorized drawing submitted as query and correspondent results.

Mixing Sketches and Images

A third type of queries supported by our application combines digitized drawings with sketches. Our prototype allows mixing sketches and vector drawings to derive more complex queries. Users can add new elements to vectorized drawings by sketching new shapes or they can delete entities by using a simple gesture command. This way, they can start with a digitized drawing and then apply editing commands to refine it.

In some cases, digitized drawings can be very complex. In other situations users may not want to search for the entire drawing. To these end, our system makes it possible to use part of a sketch or vectorized drawing as a query. To accomplish this, our prototype provides tool for selecting a region on the sketch and submitting it as a query. Thus, it is possible to discard visual elements without deleting them, making further searches easier.

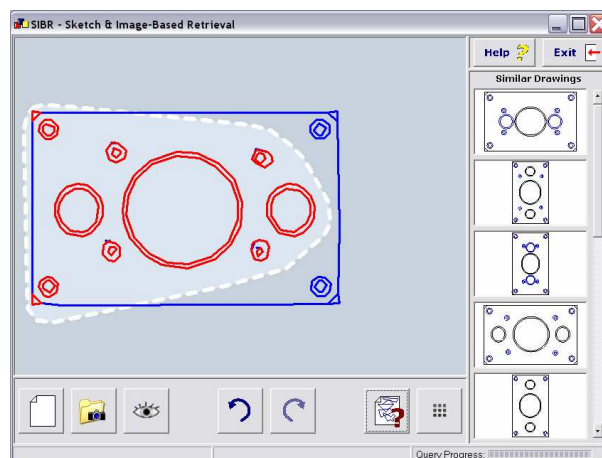


Fig. 6. Selecting parts of a vectorized drawing to define a query.

Fig. 6 presents a query created from the drawing of Fig. 5. The query was refined by adding

shapes to the original figure in order to detail more information. The figure illustrates also how to select a part of the drawing as a query, without deleting the remaining elements. To that end, we select a region that surrounds the relevant shapes.

Database Builder

We will now describe the classification part of the content-based retrieval architecture presented before. This application classifies existing drawings, by creating logical descriptors based on features extracted from drawings. Currently we are extracting topological and geometric information from polygons detected in drawings.

Before creating the database with the logical information (descriptors) extracted from drawings, our Database Builder performs simplification steps to reduce the number of elements in a drawing, thereby speeding up both classification and query. We simplify drawings by reducing adjacent lines to one, by discarding small area polygons and by removing spurious lines. This way we get a simple version of the technical drawing while retaining its main features. This will yield less descriptors and consequently a smaller database. Furthermore, simpler drawings will be easier to match to hand-sketched queries.

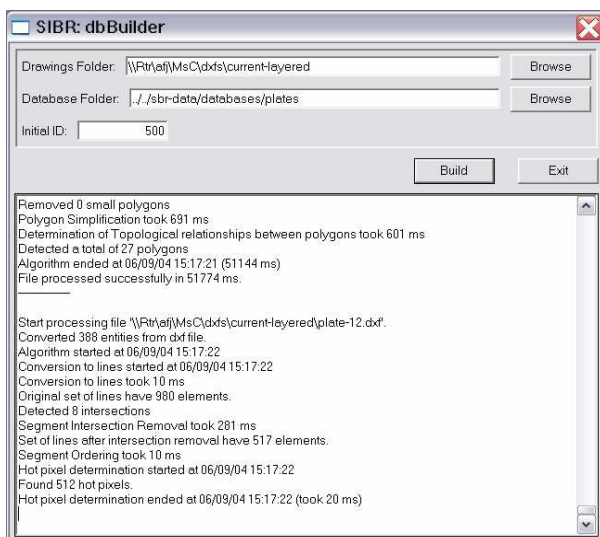


Fig. 7. Screenshot of the Database Builder.

The classification of a technical drawing might take from a few seconds to several minutes, depending of its complexity. Therefore, the classification of a large set of drawings may take several hours. However, since this is an operation that will be performed only once, users can execute it in batch mode. To that end, our Database Builder offers the possibility of automatically classifying a set of drawings without user intervention. Users only need to specify where drawings can be found and the system will

do the rest. Fig. 7 shows an image of our classification tool.

User Testing

To validate our classification and retrieval algorithms, the user interface and the specification of queries, we performed an experimental evaluation of our first retrieval prototype. At the time of writing this paper, we are performing usability tests with a improved version of the system.

The user evaluation tests involved three draftspeople from CENTIMFE and were divided in three parts. First, we explained the experiment and introduced the prototype to users. During the second part users executed two sets of queries using the prototype, first by sketching basic drawings and after by querying for simple technical drawings. Finally, users answered a questionnaire, where we tried to figure out their profile, their opinions about the prototype and their evaluation of the user interface. We also asked them informally about suggestions and ideas to improve the current version of the prototype. This experiment involved a database with 78 drawings and 12 queries.

Notwithstanding the low number of users engaged, preliminary results are very encouraging. In the majority of the queries, sought drawings were among the top five results and almost always within the top ten results. One of the things that we observed during the execution of tasks was that users did not care about where in the order of retrieval the intended drawing appears, the important fact being that it was there. One of the users commented "It [the system] found it [the drawing]! That is what counts!".

In summary, users liked the interaction paradigm very much (sketches as queries), were satisfied with returned results and pleased with the short time they had to spend to get what they wanted in contrast to more traditional approaches.

From the observations and suggestions provided by users, we developed the second version of the prototype, which we have described in this paper.

Conclusions and Future Work

In this paper we presented a novel tool to index, search and retrieve existing mould drawings using either free-hand sketches, printed drawings or a combination of both.

We believe that our system will reduce the development time of new moulds, since it allows re-using previous drawings and solutions. Furthermore, it offers to engineers and designers a

powerful new way to classify and retrieve drawings without using textual data or annotations. Moreover, it will free users from browsing through large and deep file directories or recalling part numbers, dates, locations or client and designer names. Finally, it incorporates a multidimensional indexing structure [4], which provides the mechanisms to store and efficiently retrieve large collections of drawings.

We are currently analyzing the data collected during the evaluation tests performed using the latest version of the retrieval prototype. From the outcome of these tests we plan to improve our prototype to incorporate functionalities required by users.

Acknowledgements

This work was funded in part by the Portuguese Foundation for Science and Technology, project 34672/99 and the European Commission, project *SmartSketches IST-2000-28169*.

Keywords

Sketch-Based Retrieval, Content-Based Retrieval, Query Specification, Drawing Classification.

Bibliography

- [1] D.V. Bakergem, "*Image Collections in The Design Studio*", In *The Electronic Design Studio: Architectural Knowledge and Media in the Computer Age*, MIT Press, pages 261-272, 1990
- [2] M. Clayton and H. Wiesenthal, "*Enhancing the Sketchbook*", In *Proceedings of the Association for Computer Aided Design in Architecture (ACADIA'91)*, pages 113-125, 1991
- [3] Manuel J. Fonseca and Alfredo Ferreira and Joaquim A. Jorge, "*Content-Based Retrieval of Technical Drawings*", Special Issue of *International Journal of Computer Applications in Technology (IJCAT)* "Models and methods for representing and processing shape semantics" (To appear), 2004
- [4] Manuel J. Fonseca and Joaquim A. Jorge, "*Indexing High-Dimensional Data for Content-Based Retrieval in Large Databases*", In *Proceedings of the 8th International Conference on Database Systems for Advanced Applications (DASFAA 2003)*, pages 267-274, Kyoto, Japan, IEEE Computer Society, March, 2003
- [5] Joaquim A. Jorge and Manuel J. Fonseca and Filipe Dias and Alfredo Ferreira, "*Consolidated User Requirements and Task Analysis*", 2003, available at <http://smartsketches.inesc.pt/technicalreports/tr-03May8-af.pdf>

RETRIEVING MOULD DRAWINGS BY CONTENT

Manuel J. Fonseca, Alfredo Ferreira, Joaquim A. Jorge
Department of Information Systems and Computer Engineering
INESC-ID/IST/Technical University of Lisbon
R. Alves Redol, 9, 1000-029 Lisboa, Portugal
mfj@inesc-id.pt, alfredo.ferreira.jr@inesc-id.pt, jorgej@acm.org

Manuel da Fonseca is a Ph.D. student at Instituto Superior Técnico (IST), Technical University of Lisbon (UTL), Portugal, under the supervision of Prof. Joaquim Jorge. His dissertation topic is in Sketch-Based Retrieval in Large Sets of Drawings. He received his MSc (1996) and BS Degrees (1991) in Electrical and Computer Engineering from IST/UTL. He is a Lecturer at the Information Systems and Computer Science Department, IST/UTL and a researcher at INESC-ID's Intelligent Multimodal Interfaces Group. His research interests are in Sketch-Based Retrieval, Content-Based Retrieval, Indexing High Dimensional Data, Calligraphic Interfaces, Gesture and Shape Recognition and Multimodal Interfaces.

Alfredo Ferreira is a MsC student at Instituto Superior Técnico (IST), Technical University of Lisbon (UTL), Portugal, under the supervision of Prof. Joaquim A. Jorge. His dissertation topic is in Classification and Search of Technical Drawings using Sketches. He received his BS Degree (2002) in Computer and Informatics Engineering from IST/UTL. He is a Teaching Assistant at the Information Systems and Computer Science Department, IST/UTL and a researcher at INESC-ID's Intelligent Multimodal Interfaces Group. His research interests are in Sketch-Based Retrieval, Calligraphic Interfaces, Usability Engineering and Human-Computer Interaction.

Joaquim Jorge teaches Computer Graphics and User Interfaces with the Computer Science Department at IST/UTL. He received PhD and MSc degrees in Computer Science from Rensselaer Polytechnic Institute, Troy, NY, in 1995 and a BsEE from IST/UTL. He is head of the Intelligent Multimodal Interfaces Research Group at INESC. He was co-chair of EUROGRAPHICS'98, which took place in Lisboa, Portugal. He was invited as proposal evaluator for the ITR program of the National Science Foundation in 2000 and EU's IST (Fifth framework program) INFO2000, EUREKA and related consultation meetings. He has served on the program committees of many international conferences and is member of the Editorial Advisory Board of Computers & Graphics Journal (Elsevier) and Computer Graphics Forum Journal. He is affiliated with ACM/SIGGRAPH, IEEE Computer Society (Senior Member), IFIP TC13 (Human Computer Interaction). His interests are in Distance Learning, Calligraphic and Multimodal User Interfaces, Visual Languages and Human-Computer Interaction.