ELSEVIER

Knowledge-Based Systems 14 (2001) 259–262

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

Information visualization for intelligent decision support systems

Tong Li[a,*], Shan Feng[a], Ling Xia Li[b,c]

[a]*Department of Automatic Control Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*
[b]*Department of Information Systems and Decision Sciences, Old Dominion University, Norfolk, VA 23529, USA*
[c]*National Key Laboratory for Manufacturing Systems Engineering, Xian 710049, China*

Abstract

To work efficiently with decision support systems (DSS), most users benefit from representation conversion, i.e. translating the specific outcome from the DSS, normally portrayed in a numerical format, into the universal language of the visual. In general, interpretation of data is much more intuitive if the results from the DSS are translated into charts, maps, and other graphical displays because visualization exploits our natural ability to recognize and understand visual patterns. In this paper we discuss the concept of visualization user interface (VUI) for DSS. A proprietary software system known as AniGraftool is introduced as an example of an information visualization application for DSS. In addition, a visualized information retrieval engine based on fuzzy control is proposed.

Keywords: Information visualization; User interfaces; Fuzzy sets; Software tools; Decision support systems

1. Introduction

In modern decision support systems (DSS), an increasingly larger percentage of the total design effort is devoted to the user interface, or that portion of the software system concerned with providing the means for a human user to interact with a system's application software. The structure of the user interface therefore has a major impact on the quality of the whole system [1–5].

Current DSS user interfaces are dominated by the presentation of information via character and numerical formats and is often considered an area of pursuit for those who have a propensity for numbers and enjoy mathematical equations. Unfortunately, many of the individuals who stand to benefit most from DSS, such as CEOs, executives, politicians, administrators, etc. lack this mathematically-oriented mindset.

To work efficiently with DSS, most users benefit from a "representation conversion", i.e. translating the specific DSS alphanumeric results into the universal language of the visual. In general, interpretation of data is much more intuitive if the results from the DSS are translated into charts, maps, and other graphical displays because visualization exploits our natural ability to quickly recognize and understand visual patterns. For instance, family planning DSS users better understand the problem of population age structural impacts through the display of an intuitive visually oriented population age pyramid. Macro-economic DSS users, for example, have a better grasp of the momentum of national industrial structure after seeing a bar chart moving up or down dynamically.

Steen summed up this problem nicely by defining the expression "I see" in relation to mathematics ([6]): "'I see' has always had two distinct meanings: to perceive with the eye and to understand with mind. For centuries the mind has dominated the eye in the hierarchy of mathematical practice; today the balance is being restored as mathematicians find new ways to see patterns, both with the eye and with the mind."

This can be also applied to DSS. Information visualization exploits the natural human ability to recognize and understand visual patterns. For many people, it is the easiest and most intuitive way to interpret data in the DSS application domains.

2. Visualized user interface (VUI) for DSS

In the 1970s, the user interface for DSS consisted primarily of a one line at a time dialogue with the computer, either through a command language, or through a question-answer style format. Beginning in the mid 1980s, DSS user interfaces had evolved to a mouse-driven, multi-windows interfaces such as that found in Apple's Macintosh computers, Microsoft's Windows system and X Window based systems. Visualized User Interface for DSS (VUI-DSS) is the next step in the evolution of DSS user interfaces.

The goal of information visualization was mainly to provide suitable methods and instruments to explore and depict data and information through graphical representation. Information visualization takes advantage of the fact

* Corresponding author.

0950-7051/01/$ - see front matter
PII: S0950-7051(01)00104-6

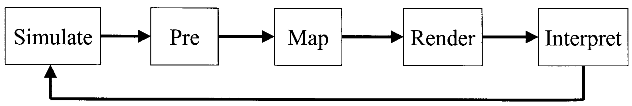

Fig. 1. Visualization pipeline.

that visual representations can serve as powerful "vehicles of thinking" that help us extract useful information from complex and/or voluminous data sets [7]. It also provides processes for manipulating the data set and seeing what may have previously been invisible, thereby enriching existing investigation methods. However, the concept of information visualization is no longer limited to the graphical display of data but now encompasses a much broader spectrum including the design of graphical interfaces used to input and access that data, in addition to the creation of standard and novel data presentation formats. With the overwhelming amount of information that is generated and received through OLTP, well-designed vehicles for facilitating data capture coupled with creative and powerful means for clearly, accurately and concisely conveying meaningful information are essential to effective DSS implementation. The DSS designers should offer users effective solutions for accomplishing these tasks. From our point of view, information visualization is not a goal in itself, but an integral part of the overall process of presenting scientific data. Information visualization functions to support DSS therefore have to be embedded into the system's software that deals with all the aspects of the scientific problem.

As Fig. 1 shows, the information visualization model is referred to as the "visualization pipeline" and consists of five stages: simulating, preparing, mapping, rendering and interpreting [8].

Data flows from left to right through the pipeline. The "Simulating Stage" is the starting step. The "Preparing Stage" involves data preparation through normalization or other mathematical steps. The "Mapping Stage" represents the natural juncture of scientific and graphical data and as the key to the visualization pipeline, is the most difficult step. The "Rendering Stage" and "Interpreting Stage" are the last two steps. VUI for DSS addresses the mapping stage of the visualization pipeline and is essentially a data- and knowledge-based graphical software shell that automatically translates DSS outcome data into charts, maps, and animations.

3. AniGraftool: an example of VUI for DSS

3.1. Tools and applications: anigraftool and DisMEIDSS

AniGraftool is a software development tool designed for VUI for DSS that aids in the development of visual user interfaces for DSS applications software. Display for Macro Economic Intelligent DSS (DisMEIDSS) is a graphical display with animation features for MEIDSS, an intelligent decision support system for macroeconomic decision-making [9,10]. AniGraftool has been employed in DisMEIDSS. With a click of the mouse, the software automatically translates the resulting data from the DSS into charts, thematic maps, scatter plots, and other graphics. As an example, time-series data can be translated into "moving" charts or thematic maps which dynamically change their shape or color-coding. DisMEIDSS's other features include its capacity for knowledge-based query and animation. The information in application domains for DSS can be inquired through indices.

3.2. Basic elements

The AniGraftool development software consists of three elements: (1) an *interpreter/inference engine* (an executable that runs under MS-Windows software); (2) a set of *knowledge base files*, including script file, keyword definition file (standard ASCII text files); and (3) a set of *data base files* (space-separated standard ASCII files) (see Fig. 2).

3.2.1. Interpreter/inference engine

The interpreter/inference engine is an MS-Windows compatible program with Multiple Document Interface (MDI) style written in Borland C++. During run time it reads knowledge base (KB) and data base (DB) files. There are three types of KB and DB files: (1) script files (ending with *.lot extension); (2) data files (ending with *.dat); and (3) keywords definition file (ending with *.def) for information query. These KB and DB files make up a specific application.

The purpose of the interpreter/inference engine is to search the users' needs by his/her query and to translate the operators in the KB and DB files into executable functions and routines for displaying a specific dialogue box, loading certain sections of the data, displaying a specific chart, or running an animation sequence. The interpreter/inference engine gets its commands from the KB and DB files through the user's selection or his/her questions. They precisely tell the interpreter/inference engine how the data are organized and what it should do with them.

3.2.2. Script files

The script files make up the core of the software. They are linked to menu or dialogue box items and precisely define what the interpreter/inference engine should do in response to the user clicking on that specific menu or dialogue box item. For instance, the script files define which columns/rows the interpreter/inference engine should read from a data file. One can specify coherent blocks of data or very complex non-rectangular data structures. One can also use the data description to select columns/rows in a data file that should be used for animation, etc. Besides this simple (but quite powerful) data description language, the script file has sections, which define the type and layout of the screen display. The developer can specify various kinds of graphs

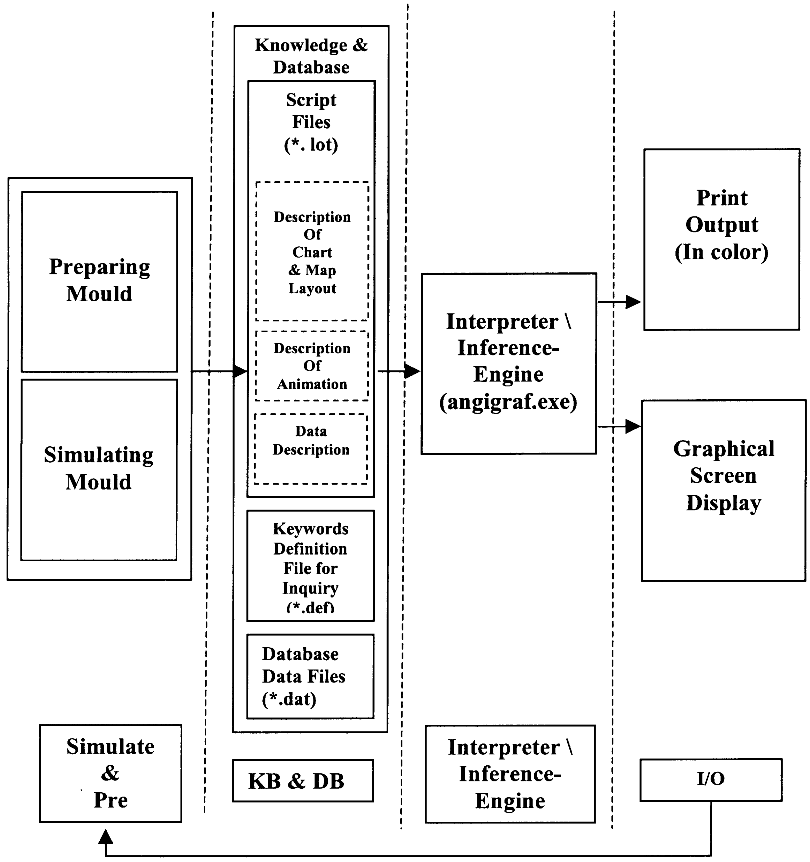

Fig. 2. Main elements of the AniGraftool.

(e.g. bar, line, pie, area, scatter plot, maps, etc.). All charts and maps can also be animated.

3.2.3. Data files

The data files are organized in very simple format: they are space-separated standard ASCII files which makes it easy to prepare the data for the graphical database. In the first section there can be a *data header*, such as the title of the specific data set, etc. that readily identifies what it is about. The next section is the *data section*, which are organized into rows and columns. A more advanced data structure could be envisioned than this simple, spreadsheet-type arrangement, but we think that this scheme has four big advantages: first, it is *easily interactive* with the application portion (the simulating stage) of the software system. The specific outcome from the DSS could be easily processed to get the scheme by a simple data preparation program. Second, it is *intuitive*. Most computer literate individuals can immediately work with data files that stick to the column/row concept. Third, it is *simple*. It does not require understanding of more advanced record-oriented data structures. Fourth, One can observe the data files directly. It is not necessary to run special software just to look into the data file (as is the case with many data bases distributed on tape). Since it is a plain ASCII file, any editor can be used to check the data.

3.2.4. Keyword definition files

The keyword definition file is for knowledge-based information query. Some definite concepts of the special application domain for a DSS are defined as the keywords for the indexes of the information query.

3.3. Knowledge-based information query

The development of a software tool to support the user interface design of a DSS is not sufficient by itself. In addition, we need to enhance the human operator's ability to use these tools. For example, although information can be displayed graphically, users may not be able to understand all of what's being displayed or may be overwhelmed if too many graphical displays are presented on the same screen. To minimize these shortcomings, a fuzzy control engine is embedded in the *AniGraftool*development tool that supports a fuzzy query based on specific keywords in the application domains.

As we know, the input of a computer application system consists of a given set of symbols. The output is a set of symbols that is readily understandable by the user. A good system, therefore, should support users in finding meaningful information in a simple way, such as through inputting certain keywords. A fuzzy retrieval engine based on the

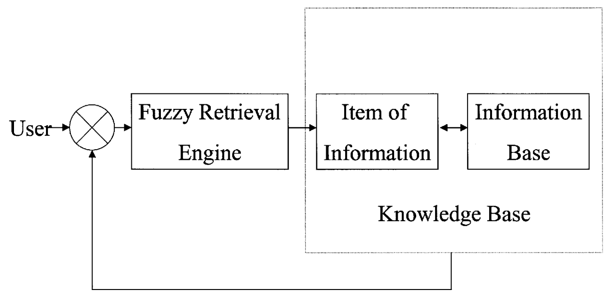

Fig. 3. Fuzzy retrieval engine and query in Anigraftool.

concept of fuzzy control that is supported by the system has been included (see Fig. 3).

Many decision support problems have data that lack obvious structures to provide information visualization with a base. We formulate $Y = \bigcirc F(X)$. Y is the set of symbols representing the user's need (also called the satisfied solution set). X is the input, such as some keywords specified in our system. F is a transformation function that transforms X into $F(X)$. ^ is the "satisfaction" operator which transforms $F(X)$ into Y following the users requirement.

The term "satisfaction" is a fuzzy concept and can be represented by a fuzzy set. Concretely, one to four keywords that are related to the application domains are allowed to be entered into the system, which define the scope of information retrieval. The visualized information that is related to the keywords are provided by the fuzzy retrieval engine (see Fig. 3). The fuzzy retrieval engine, as a function module embedded in the system, retrieves the information through interaction with user. The 'item of information' in Fig. 3 is a classified set of the information about the DSS application domains. It is the title of the content of the information supported by one to four keywords about the DSS application domains.

4. Conclusions

Information visualization is an approach that can assist DSS users in gaining insight into the quantitative data so that eventually better decisions can be reached. In this paper, we have discussed the role that information visualization plays in the DSS and introduced the software development tool *AniGraftool* as an example of the VUI for DSS.

Information visualization is a powerful tool with tremendous potential for supporting complex decision support problems and their problem-solving processes. However, information visualization is still in its infancy and requires advanced study on techniques and applications.

Acknowledgements

The project is partially supported by the grant No.69674041 from the National Natural Science Foundation of China (NSFC). The authors are indebted to Pan Wang, Jingping Huang, Chao Tang and Thomas Howard, who have contributed to this work. The authors are also grateful to the referees for their valuable comments on this paper.

References

[1] G.O. Domik, Guidelines for a curriculum in scientific visualization, Computers and Graphics 17 (2) (1993) 185–191.
[2] C. Douligeris, et al., Development of OSIMS: an oil spill information management system, Spill Science and Technology Bulletin 2 (4) (1995) 255–263.
[3] L.A. Jesse, J.K. Kalita, Situation assessment and prediction in intelligence domains, Knowledge-Based Systems 10 (1997) 87–102.
[4] P. Zhang, An image construction method for visualizing managerial data, Decision Support Systems 23 (1998) 371–387.
[5] T. Li, S. Feng, P. Wang, L. Xu, Visualization and decision support systems, in: Proceedings of the 14th World Congress of International Federation of Automatic Control, IFAC, Beijing, 1999.
[6] L.A. Steen, On the Shoulders of Giants: New Approaches to Numeracy, National Academy Press, Washington, DC, 1990.
[7] R.H. McKim, Experiences in Visual Thinking, PWS Engineering, Boston, MA, 1980.
[8] D.S. Dyer, A dataflow toolkit for visualization, IEEE Computer Graphics and Applications 10 (1990) 60–69.
[9] S. Feng, L. Xu, Decision support for fuzzy comprehensive evaluation of urban development, Fuzzy Sets and Systems 105 (1) (1999) 1–12.
[10] S. Feng, L. Xu, Hybrid artificial intelligence approach to urban planning, Expert Systems 16 (4) (1999) 248–261.