THE ROLE OF CASES IN ARCHITECTURAL PRACTICE AND EDUCATION

MONEO: An Architectural Assistant System

DINA TAHA, SAMIR HOSNI AND HISHAM SUEYLLAM
Faculty of Engineering, Alexandria University, Egypt
ditaha@idsc.net.eg , samir_hosni@hotmail.com ,
sueyllam@yahoo.com

AND

BERND STREICH
Technical University of Kaiserslautern
streich@rhrk.uni-kl.de

Abstract. The work presented here describes a prototype application, called MONEO that makes use of case based reasoning (CBR) in the field of architectural design. MONEO is a tool that aids architectural students as well as practicing architects in the pre-design phase by supplying them with an adequate number of similar past architectural cases to the design problem they have at hand. The different modules of MONEO will be presented and discussed, as well as the tools used to develop them.

1. Introduction

"…Education in architectural design relies heavily upon the use of cases as a vehicle of discourse between teachers and students; the hope being that particulars in the given cases offer a holistic view of design issues that are difficult to articulate or view if they were taken up separately…" (Dave, Schmitt, Faltings, & Smith, 1994) (p. 146). The process of using past knowledge to solve new design problems continues from being used in the education process to being widely used in design offices. In order to substantiate that, we performed a survey among a sample of architects. We prepared a questionnaire that addressed several issues such as how architects deal with past experiences, where they look for them, how much time they spend in searching for similar designs, how much time they spend in analyzing them, …etc. The questionnaire was sent to a sample of architects that represent the international community (Egypt, Germany, and USA). 42 responses were received and it was found that 100% of those who answered the questionnaire believed in the importance of experiential knowledge. While 40.5% of them use previous cases only *sometimes*, 59.5% use them *always* or *most of the times*. CBD is assumed to be a proper tool for architects since it can handle complex cases; extracting the knowledge is easy; it allows increasing the efficiency of the architects' design process; extends the memory of architects and helps in storing, archiving, and retrieving projects; last but not least CBD systems are able to learn from their previous experiences without major reprogramming (Heylighen, 2000).

For the past two decades, many CBD applications were developed. However, all of them remained in their developers' laboratories, and never saw the market. This might be due to several points. First, similar to today's commercial CAD software that needed more than 40 years of research till it took its current position and acceptance; we believe that CBD applications still need to mature. Furthermore, most of the prior CBD applications depend on textual case representation, at least the ones that perform the retrieval phase of the CBR cycle automatically. While in applications that adapt graphical representation, cases are manually retrieved and case-manipulation is the only automated phase. Therefore, we thought that a CBD application which utilizes graphical case representation for case retrieval might be a contribution, even if a minor one, for establishing CBD approaches within the architectural design offices.

2. Creative Thinking

Although Mozart would write down music almost as he saw it in his mind's eye, Beethoven felt the need to work over his ideas over and over again. Thus, great ideas are unlikely to come to us without effort. Hence, the famous phrase of Thomas Edison: "Genius is one per cent inspiration and ninety-nine per cent perspiration". The history of creativity, creative thinking, and creative problem-solving can be seen as a multidisciplinary review. Several studies have tackled creative behavior (Wallas, 1926; Patrick, 1935; Rossmann, 1931; Osborne, 1963). There was no unanimous description of the creative design process; however, the simple outline given by Kneller reflects all that was previously proposed (figure 1). He identifies five phases in the creative design process which he calls *first insight*, *preparation*, *incubation*, *illumination*, and *verification* (Kneller, 1965 in Lawson 1997).

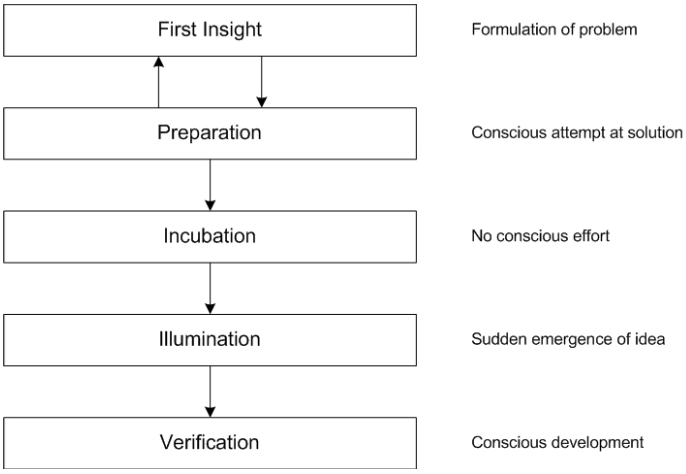

Figure 1: The 5-Stages Model of the Creative Process (Lawson, 1997).

While during the first insight phase, the designer simply recognizes the existence of the problem(s), and makes a commitment to solve them; the preparation phase involves considerable conscious effort in the search for a solution to the problem. As illustrated in figure 1, it is not a one-way flow between the first insight phase and the *preparation* phase; on the contrary, there is a lot of coming and going between the two phases as the problem might be reformulated or even redefined as the range of solutions is explored. This is an intense phase that includes a lot of deliberate hard work. MONEO targets this phase to co-work with the designer. However during the rest of the phases, where the designer's creativity is taking over, MONEO steps aside and leaves the architect to be the sole performer.

3. Case Based Design

Case based reasoning has its roots in four different disciplines, namely cognitive sciences, knowledge representation and processing, machine learning, as well as mathematics (Richter & Aamodt, 2005). Being based on the human way of thinking contributed to the acceptance and success of CBR systems. Aamodt and Plaza (Aamodt & Plaza, 1994) gave a simple description for the typical CBR cycle (figure 2) and the four phases a typical application deals with: (1) identifying the current problem situation and retrieving similar cases from the case-library, (2) using the retrieved case(s) to solve the problem at hand, (3) evaluating the solution, and finally (4) updating the case library with the new case. Yet, the CBR cycle rarely exists as described above. Many CBR tools act primarily as case retrieval systems, leaving the adaptation part to human experts.

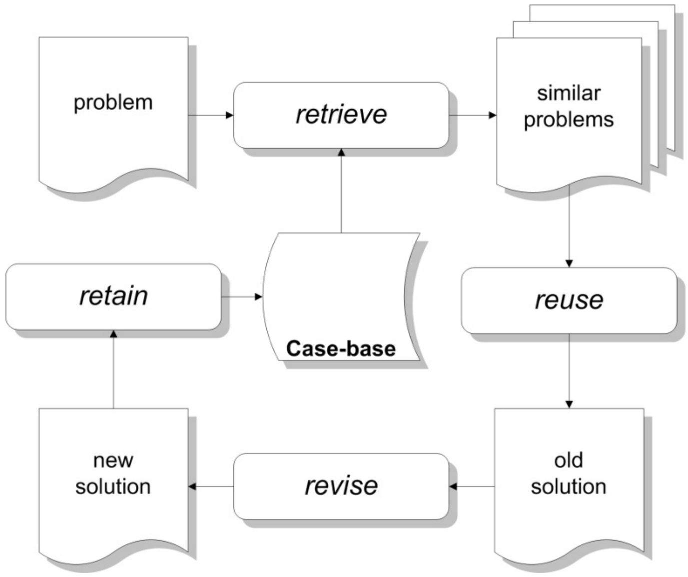

Figure 2: The 4-Phased CBR Cycle (Aamodt & Plaza, 1994)

The knowledge comprised in any CBR application is located in one or more of the so called *four knowledge containers*. These are the vocabulary, the similarity measures, the case base, and the adaptation rules containers (Richter, 2006). Since no single container is able to completely solve a task, all four are highly interactive and very dependent on each other. The distribution of knowledge over these four containers leads to the possibility of creating a functional CBR system, even if in lower quality at the beginning, the system can be improved by reorganizing the containers over time.

The *case* plays the central role in any CBR system. And in the case of architectural CBD systems, the components of the case, its presentation and representation highly contribute to the success or fail of the system.

During the Second International Conference on Artificial Intelligence in Design, the issue of how CBR techniques can facilitate the design process was raised. Researchers were divided into two main categories, the first attempting to develop systems that *do* designs, while the other group was aiming at developing *tools to assist* the human designers (Domeshek & Kolodner, 1993). In the first category it's the reasoner that adapts the retrieved cases to come out with a solution for the problem at hand; therefore, case presentation is not such a big issue. On the other hand, where the CBR system is just an aiding tool to assist the human designer, cases should be presented in a way to facilitate the communication and interaction between the reasoner and the human expert.

4. MONEO

MONEO is the Latin word for *warn*, *admonish*, *remind*, *advice*, or *instruct* (Cawley, 2005). We believed that this word perfectly suits the purpose of our developed application, which aids architects by proposing solutions, warning from potential pitfalls; or suggesting evaluation criteria by reminding them of previous architectural cases.

4.1. MONEO: CONCEPT

MONEO is an assistant system rather than an expert one. It is designed to reduce the time architects spend on searching for relevant cases to the design problems they are dealing with. As many of other CBD applications, MONEO doesn't implement the complete CBR cycle. It leaves case adaptation and manipulation processes to be manually carried out by the human experts, since these processes need the architect's creativity rather than the system's efficiency.

While most of previously designed CBD tools depend on textual representation for their cases, we believe that this paradigm doesn't fit the designer's way of thinking. Therefore MONEO applies graphical case representation rather than the textual one. However, it was not feasible to completely enjoy the powerful automated graphic reasoning without going through the intermediate stage of text and numbers. The system's user only views the graphical representation of the cases. Yet, in the background, cases are converted from the graphical representation to an attribute-value representation. This enables the reasoner to easily compare the different cases and select the relevant ones. The search results are represented as images in addition to some textual information. Afterwards, the selected case is again converted from the attribute-value representation into a graphical representation, so that the user can freely alter it before going through another iteration of retrieval.

4.2. MONEO: IMPLEMENTATION

MONEO was developed by mimicking design processes human architects go through. It consists of three subsystems which were separately developed with different software and which perform different tasks:

4.2.1. The Architectural Program Tool

Any design project starts with writing down the program. In some cases, this program is made available to the architect at the start of the project, while in other cases the architect develops it with the client. This program includes what rooms should be included, their sizes, the (connectivity) relationship they have to each other, and any special requests the client might express.

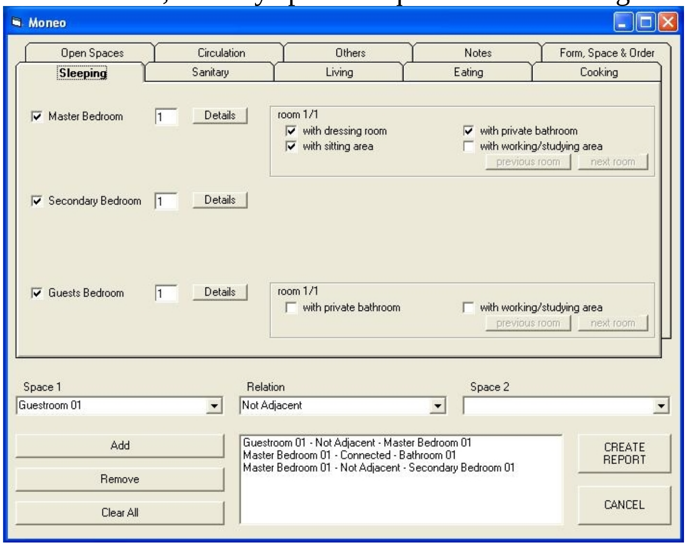

Figure 3: First Subsystem - Architectural Program Tool.

Moneo also starts with writing down the design program. A user-friendly form was designed; giving the architect a simple method to fill out the clients' needs (figure 3). The design of the form was influenced by the analysis of a number of designs for low-income and middle-income residential plans; a number of interviews with potential clients; as well as the review of web pages of design offices that provide online services.

4.2.2. The Bubble Diagram Generator/Editor

Based on the selections the architect made in the first step, a bubble diagram is automatically generated (figure 4). This diagram reflects only the specific rooms with the calculated areas, as well as the relationships between the different rooms. The current position of the rooms doesn't play any role in the retrieval process, which is conducted in the third module. That's why we don't generate all possible solutions for the bubble diagram, but rather the first possible solution that satisfies all relationship conditions. According to this generated diagram, a search query is generated. However, the system user has the ability to alter this diagram -by adding, altering, or removing spaces/relationships- before generating the query so that it reflects his concepts and visions.

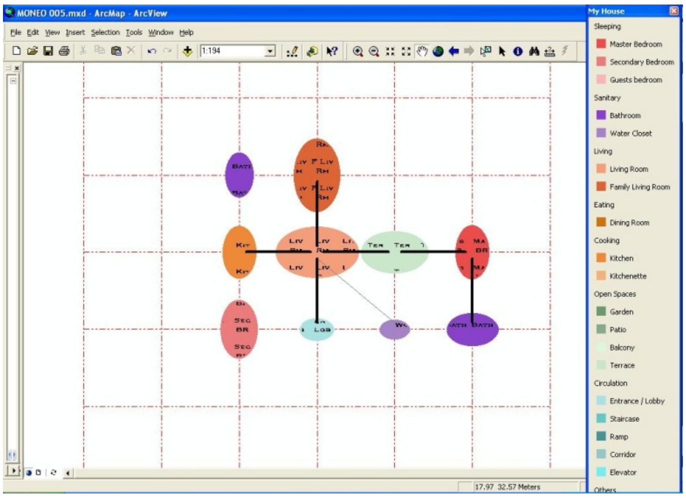

Figure 4: Second Subsystem - Bubble Diagram Generator/Editor

4.2.3. *The Case Based Reasoner*

Based upon the query generated in the prior step, cases within the case-library are compared to this query, and those that carry a resemblance to the query are retrieved. The result of this retrieval is then sent back to the second subsystem, which carries the roles of parsing the text sent back to it as well as extracting the data to be displayed to the user via a web-based interface.

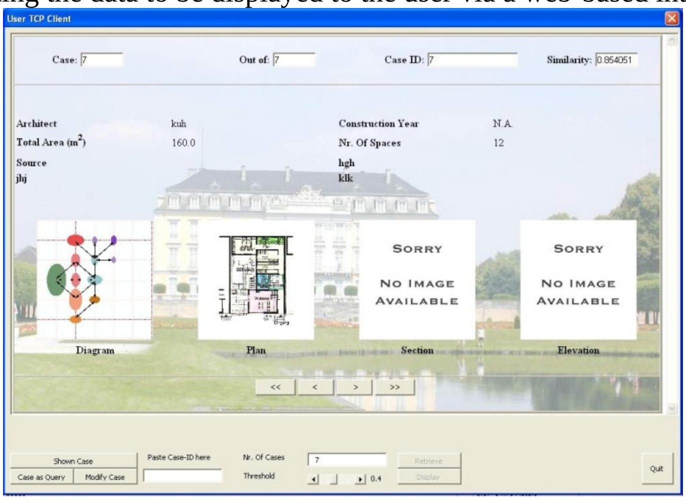

Figure 5: Third Subsystem - Case Based Reasoner

Since the design act is not a sequential process with a single starting point and a single end, but rather is a recursive one with essential needs of backtracking, MONEO enables its users to recursively query the case-library either by the generated queries based on the architectural program, or by any of the cases that were previously retrieved.

MONEO's main challenge was to link all three subsystems into a complete whole. The outputs of the first subsystem that collects the client’s needs serve as the inputs of the second subsystem that generates the bubble diagram according to this data. This link was accomplished by the first subsystem writing text files to the physical memory of the user’s machine, and the second subsystem reading and loading them before generating the bubble diagram.

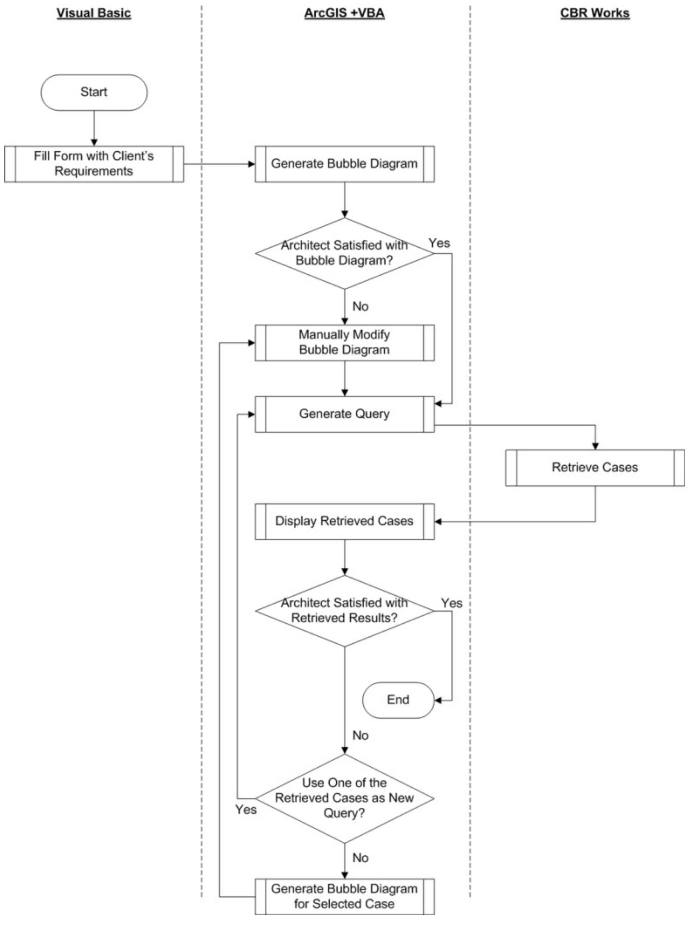

Figure 6: System Architecture.

On the other hand, a live link was needed between the second subsystem that generates both the bubble diagram and the query and the Case Based Reasoner that comprises the case library and which is responsible for retrieving cases. This was realized by introducing a TCP/IP client that links the user's local machine with the server that holds the case-library. Figure 6 represents the system architecture applied in MONEO, and how the different subsystems were interlinked.

A detailed description of MONEO's functionality and design can be found in (Taha, Hosni, Sueyllam, Streich & Richter 2004; Taha, 2006).

4.3. MONEO: EVALUATION

After the design and implementation of MONEO, the system was tested for its effectivity and efficiency. Tests were conducted to test the following: (1) if MONEO is using the same selection criteria human architects use; (2) using this criteria, will MONEO select the same cases human architects select; (3) the time MONEO takes in comparison with human users; and finally (4) what effect will MONEO have on the design outcomes of its users.

It was found that MONEO utilizes the same selection criteria utilized by human architects; yet, it needs significantly less time to retrieve the same number of relevant cases. However, same as in any other applications, MONEO is not fool proof. For sure, there still exist some bugs that need fixing, and which are discovered while running the system over and over again. For one, MONEO's case library is still relatively small. While MONEO's performance is excellent with the currently limited case-library, we don't know yet, how it will perform once its number of cases increases. Second, the system needs some optimization, which could enhance its performance. Optimization is needed in tasks such as generating bubble diagrams; speeding up time for case retrieval; as well as selecting the optimum weights for the different concepts. One final drawback exists in the difficulty of checking the case-library for redundancies. There exists no tool to perform this task, which for the time being should be carried out by the system administrator.

5. Discussion

Based upon the concepts presented and discussed throughout this research, the questions posed regarding the usefulness of CBD systems within the architectural design domain could now be answered.

On the conceptual basis, CBD is believed to hold great promises for the field of architectural design, both on the practical as well as on the educational levels (Domeshek et al., 1993; Oxman & Oxman, 1993; Heylighen et al., 2000). Therefore many CBD tools were developed within the past 15 years. While some of them targeted the educational architectural design process such as PRECEDENTS (Oxman & Oxman, 1993), EDAT (Akin, Cumming, Shealey, & Tuncer, 1997) and DYNAMO (Heylighen et al., 2000); others targeted the practicing architects, such as ARCHIE-II (Domeshek et al., 1993), CADRE (Hua, Faltings, & Smith, 1996), SEED (Flemming, Coyne, & Snyder, 1994), FABEL (Börner, 1998) and IDIOM (Lottaz, 1996).

Among these promises is the easiness of using CBD to acquire the design knowledge. While it is hard and time-consuming to induce the rules by which a design was developed, or build a model of the different design tasks and processes architects go through; telling a story about that design is not as difficult. Thus, CBD bypasses the bottleneck of design knowledge acquisition that is needed for model-based and rule-based approaches.

CBD has the potentials of improving efficiency in architectural design. Instead of always designing "from scratch", cases provide architects with past experiences they can learn from. And by that warn them from potential pitfalls and provide them with various other solutions. CBD tools can also act as external extended memory both for practicing architects and more importantly for students. Such tools are able of storing a vast number of diverse cases that no human being can. Whereas novices approach a design by trial and error, experienced architects rapidly reduce the abundance of possible design solutions to a manageable handful. The difference lies in a repertory of heuristics that is accumulated through many years of design experience (Heylighen & Neuckermans, 2003). Prior to this experience, CBD tools can provide students with substitute for the experience they lack.

The tendency of using past knowledge to solve new design problems, especially within the educational process is not a new concept in design. The earliest forms of case based instruction can be found in the first attempts at formalizing architectural education. These include the academies established during the Renaissance and the Ecole des Beaux Arts in France. Later in the 20th Century, Bauhaus which gave rise to Modernism was also built upon case based models of knowledge (Broadbent, 1995). The use of cases continues to be used with today's design studios. Akin states that "…design educators of our time use exemplars and cases when writing about curricular intent, pedagogic purpose or educational programs…" (Akin, 1997). This was agreed upon by Dave, Schmitt et al (Dave, Schmitt, Faltings, & Smith, 1994).

An experiment, documented in (Heylighen, 2000; Heylighen & Verstijnen, 2000), was carried out to verify whether exposing students to cases during the design process would lead to higher quality design products, or would have a limiting effect on creative design solutions and would increase the danger of design fixation. It was found that being exposed to past cases had a positive effect on the quality of the students' designs. It was also proved that such CBD applications don't hinder students' creativity as well as not leading to any design fixation. However, the altitude of this positive effect depends on each student's skill and motivation.

However, CBD tools are not meant to be used by students only, but another main contribution is within design offices. The process of using past knowledge to solve new design problems continues from being used in the education process to being widely used in design offices. This was verified through the questionnaire referred to in the first section of this paper. It was clear that a significant percentage of the pre-design phase total number of hours was spent in searching for past cases. In particular, 216 hours were spent in searching for cases, while 345 hours were spent is studying and analyzing them (38.5 % and 61.5 %, respectively). As for the sources they refer to when searching for those past experiences, there was an almost unanimous agreement on using architectural magazines and periodicals (95.2%). Reviewing architectural books was the second preferred source (81.0 %). It is remarkable though, that only 40.5% of the sample referred to using the internet as a source for past cases. Other sources included personal experiences of the architects, their own archives, or them paying visits to existing buildings (14.8%, 16.7 %, and 16.7% respectively).

Up to this moment, current CBD tools do not fulfil the early expectations. However, that does not mean that we should give up on further developing them. For CBD to fulfil its promises, more research and development is needed regarding many topics such as when and how they are integrated within the design process; what cases are used to build them (i.e. is it better to use shallow cases or deep ones); and how cases should be compared. But more important is to clear the current confusion of CBD researchers over the meaning of design experience in architecture, and the confusion on the side of architects, students and teachers over what CBD is used for; hence, the need of more interdisciplinary research, where both architects and CBD researchers are involved.

6. Outlook

Future work has to be conducted mainly in two directions: (1) further extensions of the system's abilities, and (2) system fine-tuning based upon further evaluations. Future versions of the system should be able to perform the following:

For the time being, the system compares the different design-cases according to the type, area, and relationship between the different components (rooms) of the design. However, a further modification of the system considers matching different designs according to their shapes (U-shaped, L-shaped, etc.) as well.

While still tackling the case comparison issue, it is planned to introduce more special cases within the comparison phase. For example, adjacent rooms could be integrated into one bigger room, and big rooms could be divided into several smaller ones (i.e. a small dining room, which is adjacent to a small living room, could be compared to a big living room with a dining corner).

It is also planned to apply the system over the Internet, so that the case-library could be updated by several architects with diverse styles, instead of being a stand-alone system with a very limited case-library. The current system design enables this feature. However, we still lack a powerful server that should be made available to the system users.

Although our study is concerned with the graphical representation and retrieval of architectural cases, this should not prohibit a compound system that uses both graphical and textual representation; since some times design data takes the form of facts and figures such as the building budget or the level of finishes.

For the first prototype, we used already existing algorithms in ArcMap© for the sake of saving up on developing time. However, we don't need such a powerful and expensive tool, whose usage introduces a legal and financial hinder for MONEO's distribution. Therefore, the used algorithms should be developed apart from any commercial tools to make MONEO more independent.

MONEO should be given the chance to be tested by architects and students to be able to fine tune the system. Although this was not feasible for the time being due to the inadequate number of cases MONEO's case-library comprises; however, these tests can be carried out in future research. The evaluation of these tests will help adjusting concepts' weights and similarity measures, as well as introducing additional suggestions system-users might find of importance.

It is hoped that such extensions would reduce the gap between research and practice and move applications like MONEO from the research laboratories into architectural offices, and design studios.

Acknowledgement

We would like to thank Prof. Dr. Michael Richter and Dr. Armin Stahl, for their enriching contribution to the research. We would also like to thank Empolis GmbH and QSIT, for providing us with CBR-Works and ArcMap, respectively.

References

Aamodt,A., & Plaza,E. (1994). Case-Based Reasoning. *Artificial Intelligence Communications*, *7* (1), 39-59.

Akin,O. (1997). Case Based Instruction Strategies in the Design Studio. In Anonymous. Spring Research Conference: College of Architecture and Environmental Design, Arizona State University: Herberger Center for Design Excellence.

Akin,O., Cumming,M., Shealey,M., & Tuncer,B. (1997). An Electronic Design Assistance Tool for Case Based Representation of Designs. *Automation in Construction*, *6*, 265-274.

Börner,K. (1998). CBR for Design. In Mario Lenz, Brigitte Bartsch-Spröl, Hans Dieter Burkhard, & Stefan Wess (Eds.), *Case Based Reasoning Technology From Foundation to Application*. (pp. 201-233). Berlin Heidelberg New-York: Springer-Verlag.

Broadbent,G. (1995). Architectural Education. In M. Pearce & M. Toy (Eds.), *Educating Architects*. (pp. 10-23). New York: AD Academy Editions.

Cawley, Kevin. Latin Dictionary and Grammar Aid. University of Notre Dame . 2005. University of Notre Dame. 10. (GENERIC)
Ref Type: Electronic Citation

Dave,B., Schmitt,G., Faltings,B., & Smith,I. (1994). Case-Based Design in Architecture. In J. S. Gero & F. Sudweeks (Eds.), *Artificial Intelligence in Design '94*. (pp. 145-162). Doordrecht, The Netherlands: Kluwer Academic Publishers.

Domeshek,E.,& Kolodner,J. (1992, June). A Case-Based Design Aid for Architecture. In J. S. Gero (Ed.), Netherlands: Kluwer Academic Publishers.

Domeshek,E., & Kolodner,J. (1993). Using the Points of Large Cases. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 87-96.

Flemming,U., Coyne,R., & Snyder,J. (1994, June 22). Case-Based Design in the SEED System. In K. Khozeimeh (Ed.), American Society of Civil Engineers.

Heylighen,A. (2000). In Case of Architectural Design: Critique and Praise of Case-Based Design in Architecture. Katholieke Universiteit Leuven.

Heylighen,A., & Neuckermans,H. (2000). DYNAMO: A Dynamic Architectural Memory On-line. *Educational technology & Society*, *3* (2)

Heylighen,A., & Neuckermans,H. (2003). Learning from Experiences: Promises, Problems and Side-Effects of Case-Based Reasoning in Architectural Design. *International Journal of Architectural Computing*, *1* (1), 60-71.

Heylighen,A., & Verstijnen,I.M. (2000). Exposure to Examples: Exploring Case-Based Design in Architectural Education. In J. S. Gero (Ed.), *Artificial Intelligence in Design '00*. (pp. 413-432). Dordrecht: Kluwer Academic Publishers.

Hua,K., Faltings,B., & Smith,I. (1996). CADRE: Case-Based Geometric Design. *Artificial Intelligence in Engineering*, (10), 171-183.

Kneller,G.F. (1965). *The Art and Science of Creativity*. New York: Rinehart and Winston.

Kolodner,J. (1993). *Case-Based Reasoning*. San Mateo: Morgan Kaufmann Publishers, Inc.

Lawson,B. (1997). *How Designers Think: The Design Process Demystified*. (Third ed.). London: Architectural Press.

Lottaz,C. (1996). *Constraint Solving, Preference Activation and Solution Adaptation in IDIOM*. Swiss Federal Institute of Technology.

Osborne,A.F. (1963). *Applied Imagination - Principles and Procedures of Creative Problem-solving*. (Third revised ed.). New York: Charles Scribner's sons.

Oxman,R.,& Oxman,R. PRECEDENTS: Memory Structure in Design Case Libraries. In U. Flemming & S. Van Wyk (Eds.), Elsevier Science Publishers.

Patrick,C. (1935). *Creative Thought in Poets.* New York: Archives of psychology.

Richter,M.M. (2006). Knowledge Containers. *9999*,

Richter,M.M., & Aamodt,A. (2005). Case-Based Reasoning Foundations. *The Knowledge Engineering Review*, *20* (0), 1-4.

Riesbeck,C.K., & Schank,R. (1989). *Inside Case-Based Reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Rossmann,J. (1931). *The Psychology of the Inventor*. Washington: Inventors Publishing Company.

Taha, D., Hosni, S., Sueyllam, H., Streich, B., & Richter, M. M. (2004). "A Case Based Architectural Design Application for Residential Units", in e-Design in Architecture, Dhahran, Saudi Arabia, pp. 109-124 .

Taha, D. (2006). *A Case Based Approach to Computer Aided Architectural Design. MONEO: An Architectural Assistant System*. University of Alexandria. (Un-Published Dissertation)

Wallas,G. (1926). *The Art of Thought*. New York: Harcourt Brace