INTERNATIONAL DESIGN CONFERENCE - DESIGN 2012
Dubrovnik - Croatia, May 21 - 24, 2012.

INTERACTIVE CASE BASED REASONING THROUGH VISUAL REPRESENTATION - SUPPORTING THE REUSE OF COMPONENTS IN VARIANT-RICH PRODUCTS

J. Johansson and M. Cederfeldt

Keywords: case based reasoning, design automation

1. Introduction

For a manufacturing company on a competitive market it is necessary to develop and produce products that meet requirements from customers and investors. One key factor in meeting these requirements is the efficiency of the product development process. Design automation is a powerful tool to increase efficiency in that process resulting in shortened lead-time, improved product performance, and ultimately decreased cost. Further, automation is beneficial as it increases the ability to adapt products to new product specifications.
This paper shows how to make use of the Case Based Reasoning (CBR) method in order to search for existing components when introducing new variants of variant-rich products. The CBR method is mature and the use of it is straight forward, but in this paper it is investigated how to enhance it by human interaction (through the integration with CAD). The method is described along with an in-production-system where the process of selecting components for roof racks for cars is supported through the use of CBR.

1.1 CBR

Wong et al. mention that when a problem is complicated and difficult, using a successful solution to a previously addressed similar problem could allow one to by-pass the resolution process [Wong et al. 2006]. This is true even if the problem is not complicated but when a new solution would mean high investments, i.e. when introducing a new product variant the reuse of components would save time and cost.
CBR is a method to digitally store experiences (referred to as cases) and reuse them in new situations (new cases). The method is based on four main operations: retrieve, evaluate (also referred to as reuse), revise, and retain [Aamodt et al. 1994], [Sriram 1997]. One advantage of CBR is that knowledge acquisition in CBR consist of a simple process of collecting examples [Bergmann et al. 2003], and when comparing to Knowledge Based Engineering a way to short-cut a work-labor intensive process of developing formal knowledge.
The diagram in Figure 1 show how these four operations are connected when applying CBR to the reuse of components in new product variants. When a new product variant is needed (new situation) the list of existing components are searched (Retrieve). If similar components are found they need to be evaluated. If a component is found to meet the requirements in the new situation it solves that situation. The component is then completely reused without introducing new article numbers or production preparations. If no components are good enough to solve the new situation some of the found similar components might be reused by slightly adjusting them (Revise). In such case the

component is reused but a new article number is introduced with minimal production preparation. If there are no existing components close to a solution new components need to be developed. Adjusted components and new components are retained in the system in a searchable way.

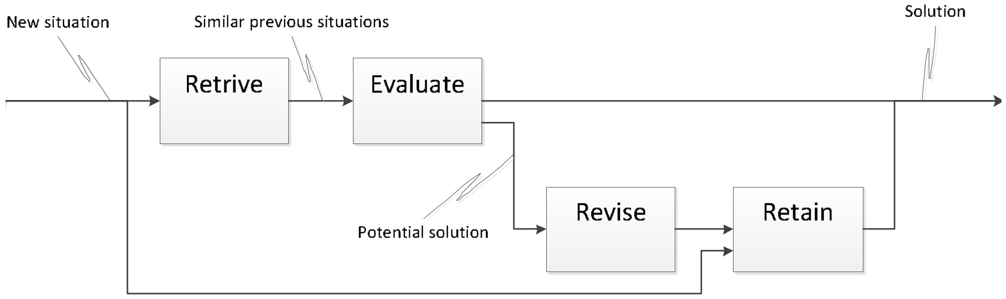

Figure 1. The CBR process applied to the reuse of components

There are several ways to build CBR systems of which the "structured" approach [Bergmann et al. 2003] is suitable for engineering applications with medium to large number of cases. Here cases are described as attributes in forms of tables. The cases are represented in the form of a common domain model [Waheed et al. 2005].

In literature there exist many examples of the use of CBR for a variety of engineering applications. Recently, a system for machining fixture design was developed that is based on the retrieval of previous designs [Peng et al. 2011]. In that system virtual reality was used to visualise final design. CBR has also been applied to the development of power transformers [Guo et al. 2011], where data mining was used to organize the case base. However, in these CBR resarch papers human interaction within the CBR-process is not mentioned. In this work it is suggested that human interaction (through spatial intellect) can support and make the use of CBR more efficient.

1.2 Eyesight and human ability to interpret abstract shapes

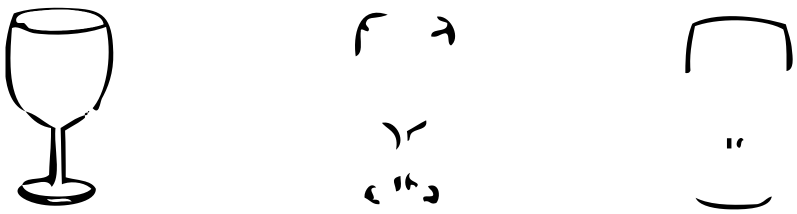

Figure 2. The illustration in the middle is easily, by human vision, interpreted to be a representation of the wine glass to the left. The illustration on the right however lacks the distinctive information needed to be "rebuilt" and interpreted [Biederman 1987]

The brain contains 10^{12} (one million million) cells connected through somewhere around 10^{14} to 10^{15} connections. It is stated that roughly a quarter of these are devoted to the function of vision. Even though we are far from understanding the perception of objects, even such comparatively simple ones as a circle, a triangle, or the letter A, we are capable of making use of it [Hubel 1995].

Geometry is essential in engineering design, yet it is very hard and time consuming to develop computer programs able to interpret geometrical relations to identify its functions (referred to as feature recognition).

A reason for human superiority over computers when it comes to recognising shapes and forms is the use of spatial intelligence [Gardner 1983]. Gardner states that spatial intelligence, common among creative people such as engineers, is totally different than mathematical logical intelligence (which could also be used to describe geometrical problems such as shape comparison). The computer deal only with mathematical and logical instructions and for it to behave like a skilled engineer when for example comparing shapes, it has first to be handed a mathematical logical explanation of what it is "seeing" and comparing. In CBR, indexing is a way of providing a mathematical logic explanation for case comparison. However, the computer is then limited to that set of mathematical logic instructions and can only compare shapes or concepts in a very limited way. Therefore computer instructions for handling detailed geometrical data are large and costly in terms of processing. Also, context and experience comes into play when evaluating shapes (or parts of a shape). See for example Figure 2 where a partial representation of wine glass (middle) is easily recognised by a human but where the indexing for a CBR system would be quite extensive and cumbersome to set up. Moreover, different levels of abstraction might need to be addressed or handled as different individuals (from different disciplines e.g. marketing, stylists, and engineers) conform to different levels of abstraction when describing or comparing attributes [Giannini et al. 2006]. For example a product's characteristics can be described as aggressive, compared to having an accelerated curve or a 30 degree angle. An inherited human ability is to swiftly shift between these levels of abstraction when evaluating concepts, a feat not so easily obtained through CBR indexing.

1.3 Paper overview

In section 2 the product targeted for design automation is described together with its design process. Section 3 details the automated process, in section 4 the resulting system is described, and is followed by conclusions.

2. Roof racks for cars

The product selected for test application is car roof racks which are mounted directly on a car roof, i.e. there are no rails on the car. Consequently, the roof rack product has to be adapted to every car-model it is supporting. The adaption is done by changing two components, the bracket and the foot pad. The foot pad is a rubber pad on which the rack is standing on the roof, and the bracket is used to fix the rack by keeping around the roof end where the doors are (see Figure 3 D).

Both safety and geometrical requirements are put on these two components, especially the bracket, since it has to keep the rack on the roof in case of a crash but still not buckle the car body when fixing the rack.

The company acts on the open market competing with car manufacturers and therefore get no nominal data of car roofs. Instead they have to collect geometrical information about car roofs by measuring.

When the roof geometry is collected for a particular car model (A in Figure 3) a foot-pad (B) is developed and the rack is placed on the foot-pad in a virtual model (C). Traditionally a drawing of that assembly was created and used to search existing brackets among printed drawings (stored in a number of binders that had to be brought when on-site testing was performed). The subsequent section describes how the CBR method has been introduced and used to support the engineers in searching and selecting brackets.

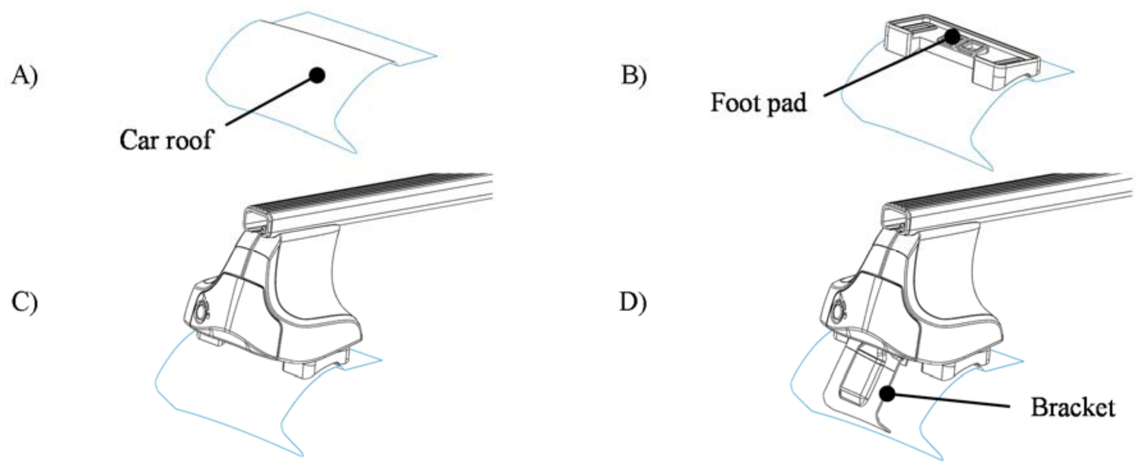

Figure 3. The roof rack product is adapted to new car-models by changing the foot pad and the bracket components

3. Applying case based reasoning to bracket selection

The time to market is critical to the company. A time-consuming step during the development process of a bracket is the search for existing brackets, taking up to several hours. Since manual search is a painstaking task with an ever increasing list of brackets the engineers tend to skip that step and instead just draw a new bracket. However, reusing brackets cut the overall lead-time up to 40%. Therefore, CBR was applied as a method of searching for brackets in 2006 [Cederfeldt 2006]. During that work, focus was also on attaining a CBR system implementation that could be understood, used, and relied upon by the engineers. An additional challenge was that of indexing cases for CBR.

3.1 Indexing and the retaining of existing cases

A difficult task in CBR is to capture the knowledge needed for retrieval of prior cases. An even more difficult task is to decide on a common domain model for storing the needed knowledge in a searchable format (indexing). In many CBR implementation cases a knowledge engineer (or someone with similar function) will try to capture the right knowledge and store this in an indexing template suited for CBR. This may result in a case description that is not in accordance with the designers' way of describing the product and/or case. Therefore it is important that the designers participate in the case indexing process in order to gain an understanding and familiarity with the process of CBR [Cederfeldt 2006]. In the 2006 CBR implementation the bracket designs were parameterised and indexing and case retrievals could be made based on the same parameterisation (consisting of a number of two-dimensional parameters). The process of adding and testing new designs was manually handled as no direct CAD integration was adopted. In this work this has been taken a step further as means of retaining new bracket designs are achieved by the interacting with CAD-models (Figure 4).

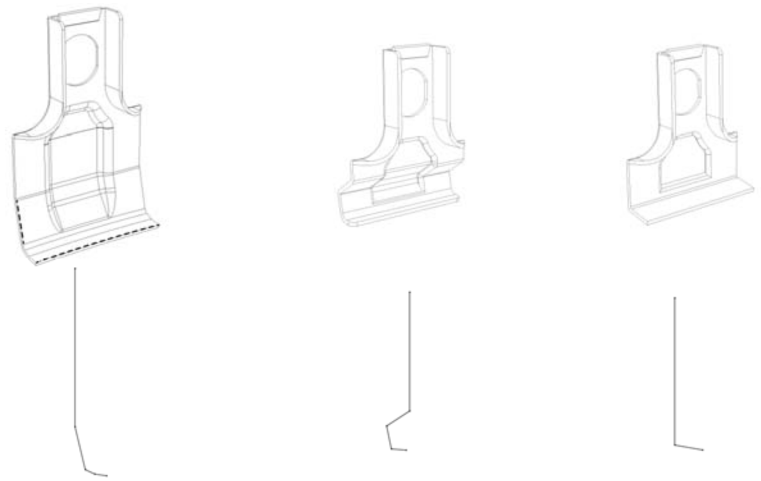

Figure 4. Three variants of the brackets (top) and their representations (bottom). In the leftmost bracket the dashed edges were selected when retaining the bracket representation

The advantage of using a variant design approach to generating searchable documentation comes from focusing on parameterisation of the design. This is because search oriented parameters are an inherent part of variant design when driven by the actual design parameters. By focusing on the engineers' way of describing the product and carefully selecting design parameters for parameterisation, the usability of the CBR approach is enhanced as a common domain model can be automatically obtained [Cederfeldt 2006].

Another advantage of using the variant design parameterisation to develop the common domain model is that the cases can be automatically extracted from CAD-models. To prevent the human factor, computer routines were developed making it possible to acquire existing brackets into the CBR system by selecting at maximum three edges of the CAD-geometry (left in Figure 4). The position and orientation of the selected edges were used to interpret their meaning so that the order of which the edges were selected did not affect the result. Parameters that are automatically recorded when retaining a bracket are the flange curvature, wall thickness, and bending points.

3.2 Retrieve

The retrieve operation of CBR is critical since failing to find relevant cases wastes all initial work of developing a common domain model and subsequently collecting and retaining existing solutions (brackets) using that model. The retrieval can be done by comparing similarity of parameter values of the cases (referred to as closest neighbour). This approach is beneficial as the order of parameters in a case base does not influence the results [Bergmann 1999].

One way to improve the retrieve operation when using the closest neighbour search is to apply weights to case attributes representing their relative importance. The weights can be set using a variety of methods [Koo et al. 2011], [Liu et al. 2012]. In this work however, the attribute weights were determined by interviewing the engineers skilled in the process of selecting brackets.

Interviews with the engineers showed that two parameters are the most important: the angle of the last part of the bracket and the position of the last bending point of the bracket.

When using the closest neighbour approach to retrieve cases, it can be beneficial to list all cases sorted based on the distance from the search criteria. When doing so the engineers get a clue of how far from a previous solution the search criteria are.

When using part geometry as a base for indexing it is important not to have an ambiguous domain model, just as drawings should be unambiguous. For instance the angle parameter is not recorded when retaining the brackets but is calculated from the two last bending-points (recorded when retaining the brackets). The angle is in other words derived from the domain model.

The retrieve function in the example system is initialised by drawing a CAD-sketch representing the bracket needed, see Figure 5. The retrieve process execution time when searching 1100 brackets is less than 1 millisecond on a laptop computer with an Intel® Core™ i7-2630QM (2,00 GHz, 6 MB L3-cache) not using parallel processing. Therefore, the retrieve process can be executed whenever the search-sketch is changed making the search for brackets interactive so that the search results are continually updated while the user moves (drags) points in the sketch.

3.3 Evaluate

As mentioned in the introduction humans are good at interpreting geometrical relations through eyesight and experience. To avoid spending a lot of time on developing an increasing number of geometry-evaluating computer routines it is suggested to let engineers visually inspect the retrieved brackets. This is done by drawing the curve representing the brackets (see Figure 4) into the CAD-model of the roof-rack mounted on the roof as exemplified in Figure 6. The CAD-model is modelled with appropriate connections between parts so that it is possible to interactively move the bracket-curve within its limitations, as illustrated in Figure 6 where the dashed arrows represent the possible translations and rotations of the bracket. When moving the bracket the representation curves are moved along and are visually evaluated against a curve representing the roof, that is the intersection of the roof-surface and the centre-plane of the bracket model (in which the bracket representation curves are drawn).

If no suitable bracket is found the project is either cancelled (due to low volume car model) or a new bracket has to be developed. Developing a new bracket is a straight forward process and it was automated using macro-programming. The search-sketch is used as input to that macro which generates a new bracket in less than 30 seconds on the computer mentioned above. When the new bracket is released for production it is retained in the case base.

3.4 Revise

In most literature, revise (adaption of selected case solution) (see Figure 1) is often pointed out as an important step in the CBR process. When applying CBR to the reuse of components in variant rich products, revising should of course be avoided as far as possible. However the identification of parameters adjustable with non or minor production preparation helps the engineers to fast adapt components close to search criteria. For instance if an existing bracket is found to have a suitable position of the last bending point and also a good angle on the last flange its flange length can be adjusted without changing tooling parts. Similarly a pin (a rivet used to position the bracket) can be added without changing the tooling parts.

3.5 Retain

The new or modified brackets are stored either as new rows in a database or as new objects (see Figure 7) in a xml-serialisation file. File size for 1100 brackets is approximately 1 megabyte when using the xml-serialisation routines in the Microsoft.Net 4.0 framework and approximately 2.5 megabyte when using Microsoft Access database.

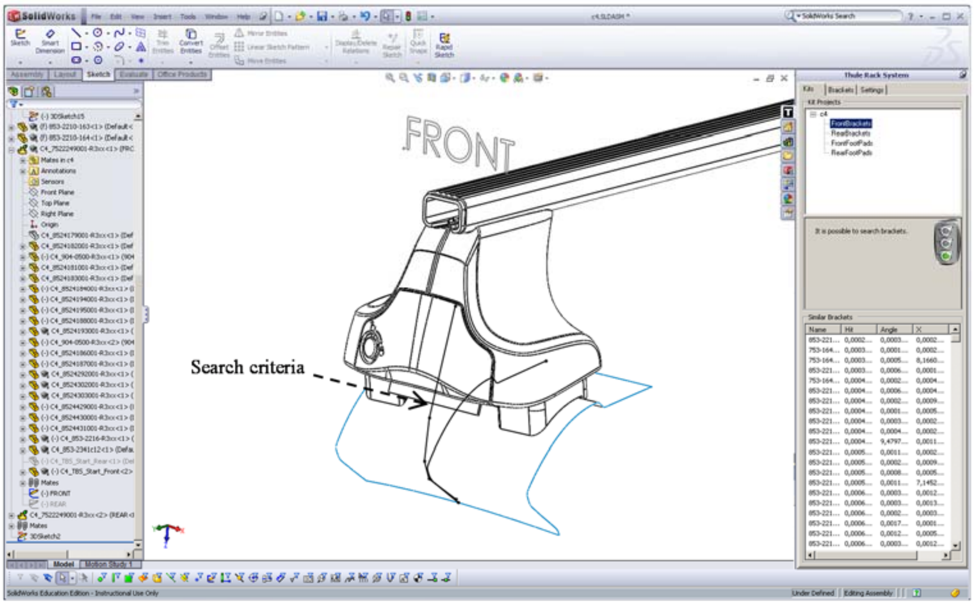

Figure 5. The search is initiated by drawing a perfect "bracket-curve" for the new car model

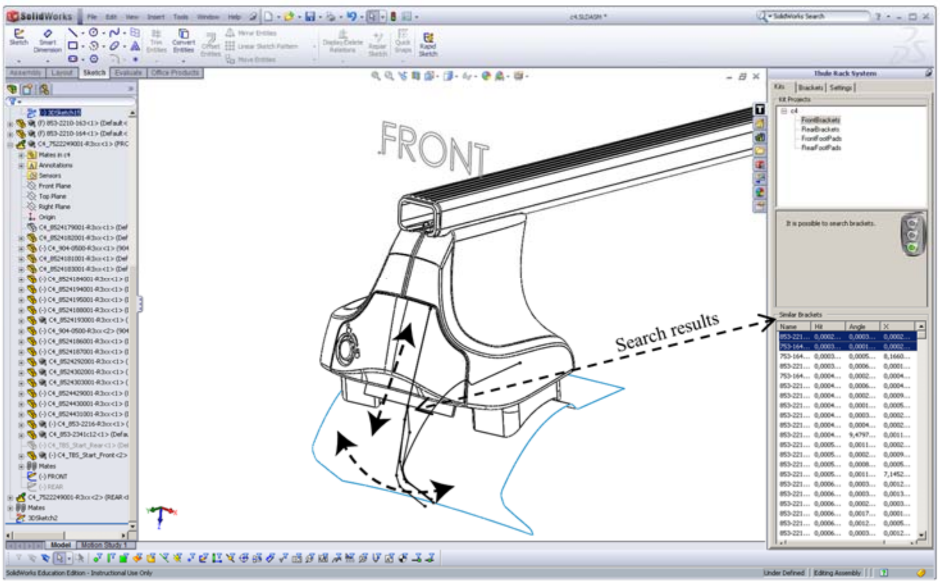

Figure 6. All similar brackets found in the CBR search are then automatically drawn superimposed with the search criteria and then visually evaluated by the engineer(s)

4. System architecture

Object oriented programming was applied in the example application since it makes the resultant software more maintainable, adaptable and reusable [Hopgood 2001]. The base of the example system consists of the bracket, the bracket manager, and the search classes, see Figure 7. A bracket object carries the information on a specific bracket, i.e. the values of the attributes, but also has functionality to calculate angle, and number of bending points. The bracket manager contains all brackets and marshals bracket events. The CBR algorithm was also implemented as a class where the DoSearch function takes one array of parameter values representing the search criteria and a set of parameter arrays representing the case base, and returns a set of parameter arrays representing the found similar brackets.

The user interface contains lists and editors for brackets and is contained in the SolidWorks application window as an add-in, see right panel in Figure 5. Many of the functions were added to the user interface as contextual mouse menus.

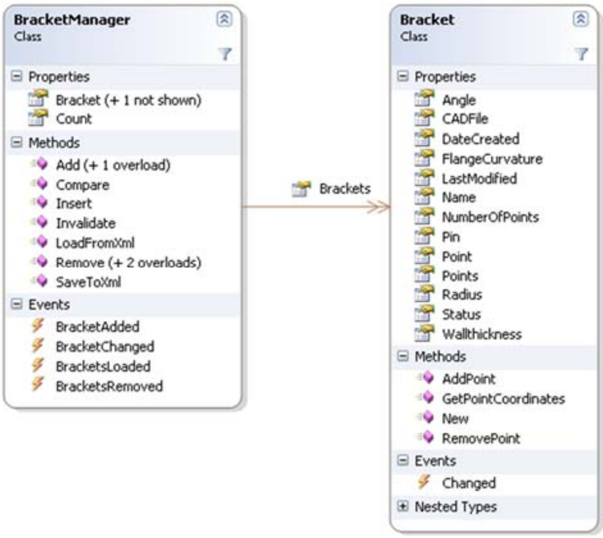

Figure 7. Class diagram showing the bracket class, the bracket manager class, and the search class

5. Discussion

As mentioned, the search for existing brackets was addressed in 2006. That system was used with some success, eliminating several duplicate designs. However it did not incorporate CAD-functionality directly and parameterisation had to be performed manually. The engineers felt it was time consuming and error prone to type in parameter values manually, and therefore some of that work was eventually outsourced. When adding the visual representation the whole case base could be drawn in one part and examined altogether. It was then realized that a vast number of the cases were erroneous due to typos when collecting and retaining the brackets. It is believed that if the input work had not been outsourced many of the mistakes would have been prevented due to the engineers know-why (compare Figure 2 where the wine glass representation in the middle would be hard to interpret to someone who has never seen a glass). When adding visual aids and the possibility to retain cases using CAD-geometry, the input work is faster, fault safe, and again done by the engineers. The whole case base, containing 1100 brackets, was re-input using CAD models in two days.

One of the engineers that found a set of similar brackets when trying the system on a real case exclaimed “We just saved 2500 euro in one click”. After two weeks several doublets of brackets were identified in the case base. Even though the effort to eliminate brackets is not gainful, the potential of avoiding doublets in the future is clear. In other products the elimination of duplicates could be highly profitable.

The engineers state they have started using the CBR-system totally integrated in the development process, and that it has affected the way of developing the footpads, so that if a quite good bracket is found adjustments in the foot pad can make the found bracket useable without adjustments.

Benefits of running the system is not only time to market but also the possibility to develop racks for low volume cars, if there are existing (good enough) brackets.

One drawback is that the company now have yet another system that needs maintenances. Someone needs to maintain the case base so that it only contains brackets that are ready to be produced, such information exist in the PDM system but no integration has been done.

6. Conclusion

It is possible to support the reuse of components in variant rich products through Case Based Reasoning. When doing so it is beneficial to make the cases visually interpretable by engineers as that speeds up the evaluation as well as increases the quality of the information in the case-base. Humans are superior at interpreting geometry – computers are good at number crunching. Combining the two, makes for a powerful system.

Acknowledgement

The authors want to thank the Thule company for financial and technical support and the knowledge foundation (KK-stiftelsen) for financial support.

References

Aamodt, A.,Plaza, E., "Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches." Ai communications : the European journal on artificial intelligence., Vol. 7, No. 1, 1994. 39.

Bergmann, R., "Developing industrial case-based reasoning applications : the INRECA methodology", Springer, Berlin; New York, 1999

Bergmann, R., Althoff, K.-D., Breen, S., Göker, M., Manago, M., Traphöner, R.,Wess, S.,"Case-Based Reasoning Approaches Developing Industrial Case-Based Reasoning Applications" in "Springer Berlin / Heidelberg, 2003

Biederman, I., "Recognition-by-Components: A Theory of Human Image Understanding." Psychol Review, Vol. 94, No. 2, 1987. 115-147.

Cederfeldt, M., "Towards a Strategy for Mapping of Design Problems to Suitable Solutions : A Case of Design Automation Using CBR". Proceedings of the DESIGN 2006 : 9th International Design Conference, D. Marjanovic (Ed.), Dubrovnik, Croatia, 2006

Gardner, H., "Frames of mind : the theory of multiple intelligences", Basic Books, New York, 1983

Giannini, F., Monti, M.,Podehl, G., "Aesthetic-driven tools for industrial design." Journal of Engineering Design, Vol. 17, No. 3, 2006. 193-215.

Guo, Y., Hu, J.,Peng, Y., "Research on CBR system based on data mining." Applied Soft Computing, Vol. 11, No. 8, 2011. 5006-5014.

Hopgood, A. A., "Intelligent systems for engineers and scientists", CRC Press, Boca Raton, 2001

Hubel, D., "Eye, brain, and vision", Scientific American Library, New York, 1995

Koo, C., Hong, T.,Hyun, C., "The development of a construction cost prediction model with improved prediction capacity using the advanced CBR approach." Expert Systems with Applications, Vol. 38, No. 7, 2011. 8597-8606.

Liu, C.-H.,Chen, H.-C., "A novel CBR system for numeric prediction." Information Sciences, Vol. 185, No. 1, 2012. 178-190.

Peng, G., Chen, G., Wu, C., Xin, H.,Jiang, Y., "Applying RBR and CBR to develop a VR based integrated system for machining fixture design." Expert Systems with Applications, Vol. 38, No. 1, 2011. 26-38.

Sriram, R. D., "Intelligent systems for engineering : a knowledge-based approach", Springer, London, 1997

Waheed, A.,Adeli, H., "Case-based reasoning in steel bridge engineering." Knowledge-Based Systems, Vol. 18, No. 1, 2005. 37-46.

Wong, F. S. Y., Chuah, K. B.,Venuvinod, P. K., "Automated inspection process planning: Algorithmic inspection feature recognition, and inspection case representation for CBR." Robotics and Computer-Integrated Manufacturing, Vol. 22, No. 1, 2006. 56-68.

Joel Johansson, PhD, Assistant professor
School of Engineering/Jönköping University, Dep. of Mechanical Engineering
Box 1026, 551 11 Jönköping, Sweden
Telephone: +46 36 10 16 75
Telefax: +46 36 12 53 31 (fax)
Email: Joel.johansson@jth.hj.se
URL: http://hj.se/jth/en/research/staff/product-development/joel-johansson.html