Constructive Representation in Situated Analogy in Design

An essay on a bottle of Eau d'Issey pour Homme

Jaroslaw M. Kulinski and John S. Gero
University of Sydney

Key words: Design, Analogy, Situatedness, Knowledge representation

Abstract: A model of situated analogy considered within the context of design is presented. It shows the impact of constructive knowledge representation on analogy making. The importance of a non-fixed but constructive representation is highlighted. It is suggested that a situated model of analogy fits the observed characteristics of design better than a non-situated one.

1. ARTEFACT ANALYSIS

This part presents the situated nature of design through an inductive analysis of a hypothetical design scenario used in the creation of an artefact. This design scenario is constructed using analogical reasoning. It does not aim to replicate the actual design process that was employed, rather it gives a plausible explanation of how the final result might have been reached and it serves as a scaffolding for a theoretical model that is the subject of the latter part of this paper.

1.1 A bottle

A recent example of industrial design: a bottle of perfume of Eau d'Issey is presented in Figure 1. The particular shape of this container reflects the specific aims it was developed for, i.e., a container for two complementary cosmetic products in the form of a travel kit. Let us try to analyse the design process that hypothetically could lead to this very specific form. Since the subject of this research is focused on analogical reasoning, the scenario invented here is based on the application of analogy as a design strategy.

Figure 1. Karim Rashid: L'Eau d'Issey pour Homme; 2-in-1 travel kit; blow moulded polypropylene bottles with adflex coating

The design process as defined by Gero (1990) is a process of the production of a description of an artefact that meets certain functional requirements (the brief). Functional demands that are related only to the physical performance of an artefact form a core of the artefact specification. In this example, they can have a following form: provide a portable container(s) for two liquid substances. The abstracted form of a brief is useful as a search strategy for a prototypical solution to the design problem. Thus, a prototypical answer to the design problem at hand is a bottle.

However, placement of the design within the actual context reveals a much richer set of functional demands. In the example the product that is to be carried by the container being the subject of the design process is a perfume, moreover it is targeted at male users. Thus, once placed within any cultural context it reveals a much richer cultural functional set. In this design the bottle is not only a container but also it is a signifier of the other contextual meanings. The bottle of perfume is then not a simple container it is also a cultural marker.

The reason for this digression is to make the reader realise how easily a human subject is able to re-classify an artefact, or using more computational terminology how easily one can re-represent an object when considered within a specific context.

The following analysis makes an extensive use of re-representation as a method that contributes to the development of the design.

1.2 The “zipper” bottle

The little bottle that is considered here was conceived as a travel kit. In the context of travel the functional requirements are related to efficient space utilisation, the reduction of the number of items to take care of, and shock resistance, travel also introduces the notion of movement and speed. These aspects can be recognised in the design form.

The design of the bottle is the physical expression of compacting, i.e., a single artefact yet serving a double purpose. The 2-in-1 bottle design reinvents the bottle not as a container but as a package that conveniently reduces the number of individual items, i.e., less things to forget. Let’s trace hypothetically how these elements could have been incorporated within the design of this double bottle. For an artefact, the feature of being compact hinges on the ratio between the actual form and its bounding box, i.e., the more the actual shape resembles its bounding box the ratio of its compactability is higher. The subject of this example apart from being individually compact must also be made out of two parts (2-in-1). The issue was to create a form that is singular yet dividable into parts. And further, the least tangible aspect of the representation, it has to embody the notion of travel.

To recapitulate, the few behaviours that we are seeking in this design are: 1. a single artefact though dividable, 2. a shape that easily fits other objects, 3. bringing in-line the broad cultural notion of travelling.

There are number of methods that can be applied to achieve the requirements, but expressively we are using only one, that is, analogy. Let us imagine than we scan through a set of other artefacts for an object having a matching set of behaviours. One of our candidates could be a zip fastener. A zip fastener, functionally, is a device that connects two in one, i.e., without modifying its own structure it snuggles its part together in one. It is made as two collections of teeth that fit into each other. Also for a human designer, a zip fastener fits within the broad category of an object relating to travelling (ex. travel baggage).

The objective of analogy in design is to extend the design state space by transferring new behaviours (and associated structure features) from the source design to the target. In this example the target design is the double bottle, the zipper is its matched source. One of the behaviours that is unique to the zipper is the snapping together of its parts. This behaviour can be transferred to the double bottle design. This requires structure modification of the singular parts into tooth shaped elements that can snap together.

This example aims to present the issue of representation as a factor that allows a novel analogical match between two objects. In the subsequent paragraphs this intuitive example will be investigated more rigorously giving a general model of situated analogy in designing. This model is defined as situated since at the core of its operation lays the feature of constructive representation.

2. DESIGN, ANALOGY AND REPRESENTATION – THEORETICAL FRAMEWORKS

Analogy is used as one of the strategies employed in the design process that has been recognised as one that can change the design state space. (Gero 1990, Gero & Kazakov 1999, Navinchandra 1991).

2.1 FBS model of design

According to the model of design proposed by Gero (1990), the design process is a task of producing a description, D, of the structure, S, which responds to functional requirements, F, through expected behaviours, B. In order to accomplish these design tasks one has to define the mapping between these three states. In such a framework it can be said that a design is represented as a triplet composed of those three states: function, behaviour and structure (FBS). The FBS triplet can also be defined as causal and abductive knowledge, since it represents a mapping between the design structure onto a set of behaviours that such a structure presents and onto a set of functions that can be attributed from the behaviours, and vice-versa. The use of this model provides an opportunity to represent any design as a graph of relations between states. Throughout the rest of this paper making a "design representation" means creating a graph of transitions between these three states.

2.2 SME framework for analogy

The model of analogy adopted in this research was proposed by Gentner (1983) (also Falkenheimer, Forbus & Gentner 1989) and termed Structure Mapping Theory. The important feature that makes this approach usable within a computational environment is the focus on syntactic properties of knowledge representation, and not on a specific content of the domains. The central proposition of the structure mapping engine (SME) is that analogy is a mapping of knowledge from one domain (the base or the source) into another (the target) which conveys that a system of relations known to hold in the base also holds in the target. The target objects do not have to resemble their corresponding base objects. Objects are placed in correspondence by virtue of corresponding roles in the common relational structure. The model that we propose is an extension of SME that includes the issue of how the actual representation for each of the analogical domains is constructed.

2.3 Situated perspective on analogy

Analogy-making relies on matching two domains that share common representational structures. This requires us to consider knowledge representation. If the knowledge that is stored in long-term memory follows the encyclopaedic paradigm, that is, the knowledge is represented in as fixed, self-contained chunks that can be uniquely addressed and retrieved, analogy making is reduced to a search and retrieve process. Such a conclusion, however, stands in clear opposition to everyday experience in which analogy is acknowledged as the ubiquitous mechanism by which humans understand abstract concepts and carry out abstract reasoning (Lakoff & Johnson, 1980 in Nehaniv, 1999).

In order to escape this search-retrieve model, it has been postulated that analogy-making relies on seeing two domains as “the same” (Chalmers, French & Hofstadter, 1995), that is, finding means of representation that allow matching two domains with respect to specific criteria. In this view making analogy hinges on the ability to re-represent the knowledge in a flexible, i.e., adaptable to needs, manner.

The situated cognition perspective postulates an alternative to the encyclopaedic paradigm of knowledge representation. From the situated standpoint, the setting (i.e. the actual context and its perception) within which human actions develop, shapes the representation of knowledge to its own needs. It is claimed that what people perceive, how they conceive of their activity and what they physically do develop together (Clancey, 1997). As a result of this standpoint a different paradigm of human memory is proposed: it is no longer a repository of ready made structures but rather the memory provides the capability to produce structures, called representations. (Clancey, 1991; Rosenfield, 1988). By applying the notion of constructive representation within analogy-making we are able to meet the postulate of tailor-made domain representations.

2.4 Design as a situated activity

A design process that is not limited to optimisation only, is seeking novel means of meeting the functional requirements of the brief. Schön & Wiggins (1992) observing his own and other designers’ actions concludes: that during the design process under the pressure of perceptual processes an alternative view is developed on the problem at hand. He calls this phenomenon as “reflection in action”. The acknowledgement of this novel view allows inclusion of the emergent features that were not accounted prior to the design process starting.

Partial support for this view can be found in the protocol studies of designers. Suwa, Gero & Purcell (1999) have found that the design requirements that drive the process are built inside the design process itself. They examined the notion of unexpected discoveries (UXDs) that occur to the designer during the design process. They looked at their correlation with the formulation of more general design strategies called situated-inventions (S-inventions). S-inventions are defined as the generation of inaugural issues or requirements in the current design task. Since such formulations occur during the design process they are situated in this process. The S-inventions form the basis for the formulation of new design goals. The study pointed out an important bi-directional causality between unexpected discoveries and the formulation of design goals. A conclusion of this study suggests that the pursuit of various design goals is mutually interwoven. A solution found for a particular set of design goals may potentially trigger reformulation of the design problem through the generation of alternative goals.

Figures 2 and 3 show the reflective process and its consequences on the development of the design process. In Figure 2 the design process is considered as a linear solution process that progresses from the initial problem to the conclusion in one pass.

Figure 2. Initial stage of the situated design process: problem framing

Figure 3 presents the subsequent stages of the process, i.e., the reflective component. The reflective process starts where the preceding phase ends, i.e., with the representation of the artefact (rounded corner box in Figure 3 indicates the new starting position). The comparison of the perception of the design representation in the brief and in the artefact is responsible for finding the emergent features (UXD) in the latter one. These differences lead to modified design goals and subsequent re-representations of the design artefact. The dotted arrow from the last representation of the artefact indicates that it is a cyclic process that can be repeated an arbitrary number of iterations.

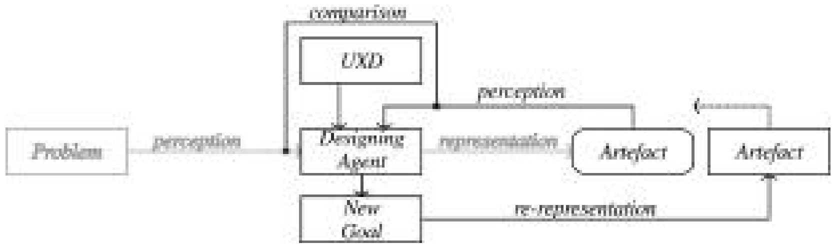

Figure 3. Reflective stage of design process
Greyed elements indicate those in the preceding phase.

3. A MODEL OF SITUATED ANALOGY

There are two aspects of situatedness in this model; the first is related to the mode of the construction of the representation of analogical domains and the second is related to the whole design process as described earlier. Figures 4 to 7 present in a time sequence the analogical process within the framework of situated cognition. Elements greyed out indicate that these particular ones do not take part in the episode. Various types of lines indicate logically continuous flows within a single episode.

Figure 4 presents the initial problem framing, in the case of analogy, it is finding and representing the target design.

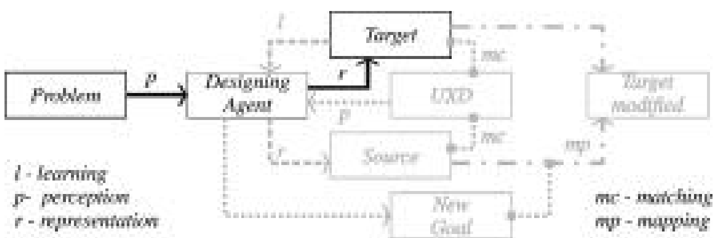

Figure 4. Target representation

Figure 5 presents the continuation of the analogical reasoning, i.e., a search for a source candidate that can match the target representation. For the sake of clarity this diagram is substantially simplified and presents only one potential source candidate. The search for an appropriate source candidate starts from the target representation. The learning link from the target indicates that the process of constructing a target representation has been learnt by the agent.

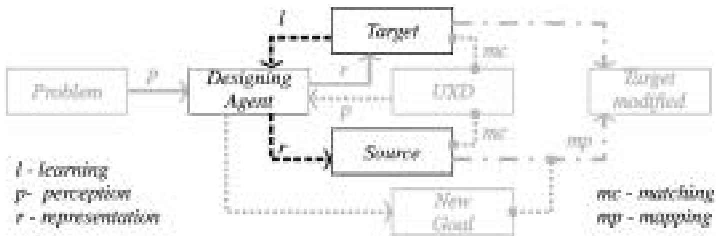

Figure 5. Source representation

This knowledge in turn is applied when the designing agent attempts to construct a matching source representation. In this learning process the form of the source representation is dependent on the target representation: without the target representation there would be no means of constructing the source representation.

Figure 6 shows the consequences of finding an analogical match between the target design and source design. Referring back to the earlier example, the finding that two bottles in a travel kit can be matched against a zip fastener suggests a discovery.

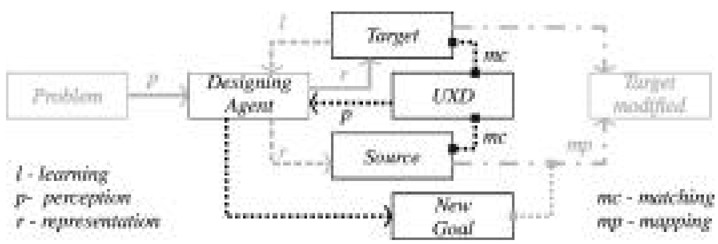

Figure 6. Analogical match

Figure 7 presents the final stage of a single reflective episode in a design process, i.e., the mapping of the analogical match. Mapping in terms of analogy means the process of transferring certain structural features found in the source but not present in the target, that is, modifying the target structure. As in the example a modification of the bottle shape that would provide that “snapping” behaviour.

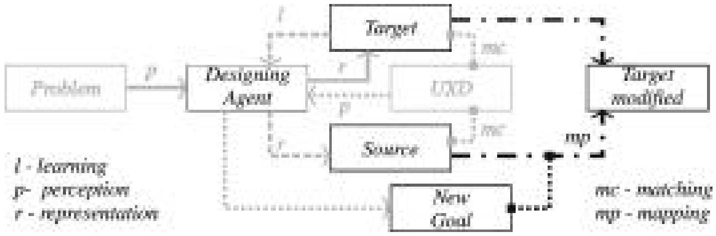

Figure 7. Analogy mapping

4. SYSTEM ARCHITECTURE

The computational system is not developed as an autonomous design system but rather a mind aid for a human designer. Thus, at different stages of operation human input is expected in order to drive the process. The operation of the system can be described as an interaction of 7 different agents. Figure 8 shows the flow of data between the agents. The process is initiated by the Target Chooser that on the basis of a brief analysis selects the target design. It means that from the knowledge base of known designs a target structure is selected. The next step in the process is the construction of the FBS graph of the target design. Through analysis of the brief a context of application is selected.

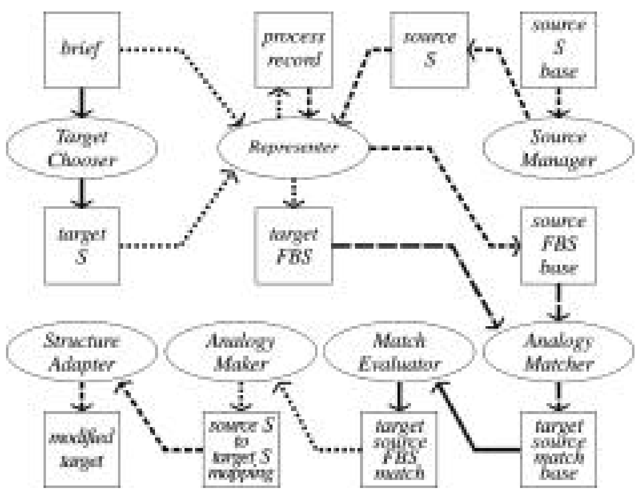

Figure 8. System architecture

This context determines the scope of functions that the target structure is to meet. The function set allows the derivation of the set of expected behaviours that implements the functions. The Representer uses the context derived expected behaviour set and inspects the target structure for features that support the expected behaviour. The set of successful behaviours found in the target structure is recorded. Subsequently it is used as set of expected behaviours when the source candidates are represented. The role of the Source Manager is to run the representational process for each of the source structures in the knowledge base of familiar designs. The product from this process is a base of potential source candidates. The role of the Analogy Matcher is to find the most successful matches between target and source candidates. For a single pass of the system the Match Evaluator selects only one matching pair for further processing. On the basis of this selection the Analogy Mapper proposes an analogical conjecture between the target and the source structures. This conjecture forms the basis for the Structure Adapter to modify the target structure in order to transfer additional features found in the source structure to target structure.

5. IMPLEMENTATION

Here we present only the implementation of the Representer. The construction of the representation process is built around a queue that uses and manages small software agents that can detect design structure features that can support a particular behaviour. Each system behaviour has an associated perceptual agent.

5.1 Elements of the Representer

5.1.1 Knowledge base of familiar design

In the knowledge base the designs are represented only partially, that is, only their structure is encoded. The structure representation remains static throughout the whole process. The structures are composed as a hierarchy of compounds, parts, relations and attributes. On a compound level each design is encoded with all its constitutive parts and topological relations between them. In turn, on a part level each item is considered as an independent entity and the description stored encodes only its physical properties. For example a partial encoding of a bottle design is represented as follows:
on a compound level:

parts(bottle):{bottom, side, neck, cap}
orientation(side):{vertical}
relation(side, bottom):{perpendicular, fixed}
relation(side, neck):{fixed}

relation(side, neck):{bigger}
on a part level:
shape(side):{void, rectangular, continuous, axial}
material(side):{plastic}
…

The knowledge base of design structures is the area in which behaviour detecting agents operate.

5.1.2 Base of known behaviours, functions & contexts

This base is arranged as a graph of relations where there are three categories of nodes, i.e., behaviours, functions and contexts (BFC). This network can be read as mapping between these elements. An arc that links a function *n* and a context *m* can be read as function F_n *is relevant in* context C_m. The arcs between behaviours and functions can be read as: function F_n *is implemented by* behaviour B_m.

5.1.3 Behaviour detecting agents

All behaviours present in the BFC network are associated with respective software agents. These agents are equipped with the rules that can examine design structures encoded in the knowledge base of familiar designs. These rules attempt to determine whether a particular structure presents a particular behaviour or not.

Apart from the detection rules, the agents are also equipped with rules that mediate the relationship between the agents themselves. Thus, there are effects of agent actions: 1. a success or failure in the behaviour detection process and 2. an input into the selection process of agents activation. The latter action is dependent on the result of the former.

5.1.4 The queue processor

The queue processor is the manager that orders and triggers the action of behaviour detecting agents. An overview on the main processing queue is shown in Figure 9.

When a set of behaviour detecting agents is posted to the queue it is ordered and executed in a sequential manner. Due to the nature of ordering in the queue some of the agents may not be able to pass the activation threshold that is required to become executable. These agents are removed from the queue without being processed and they fall into the “discarded bin”. The ones that pass the activation threshold after being executed fall into either successful or unsuccessful bins. They are able to post to other related agents to the queue, shown in Figure 9 as the *in-process batch*. The process terminates when all posted agents in the queue are exhausted.

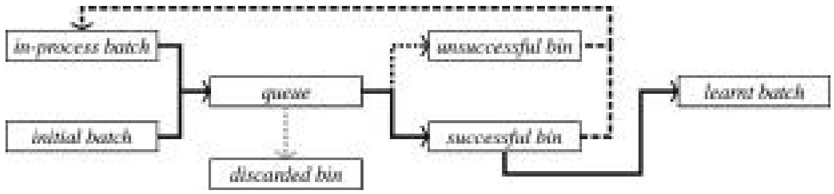

Figure 9. Process queue in representer

The order of execution of the agents in the queue is dependent on their level of activation. The activation of an agent is calculated on the basis of two parameters: the initial confidence factor and the age of posting. These two factors work in an antagonist manner, that is, the age lowers the confidence factor. The age factor constantly increases during the queue processing. It is measured in process cycles of equally valued items in the queue. As a result the agents that were posted with low confidence level have very little chance to become executable, since the in-process posted agents are always younger. The rationale for this process is grounded in an assumption that in a situationally driven process the objective is to search only for the most situation-relevant results and not to proceed with a wide search for all possibilities. The in-process posted agents are the followers of the initially selected ones. This phenomenon of following can be interpreted as a self directing search since the development of the process serves as a reinforcement factor for the direction of the search.

5.2 Operation

The representation process is initiated by a selection of agents posted in the queue. The selection process differs in the case of target and source representation. In the case of the former the initial batch is context derived, while in the latter case it is the learnt batch that has been acquired in the course of constructing the target representation.

In the example presented earlier one of the expected behaviours of a bottle in the context of travel was efficient space occupation. The detecting rule for an agent seeking the efficient space usage has the following form: "If compound *x* (or sub compound *x*) has shape(*x*):{axial, rectangular} and orientation(*x*):{vertical or horizontal} and relation(*x*,*others*):{bigger} then *x* can have an efficient bounding box ratio". Success in meeting the above rule moves this agent to the "successful bin" and it will be re-applied when the source representation is constructed.

The positive finding by this agent also triggers the posting of a related agent such as the one that detects whether two compounds (sub compounds) can fit side by side. A rule for such an agent can be written as follows: "If x_1 and x_2 have shape(x_n):{axial, rectangular} and relation(x_1,x_2):{parallel} then x_1 and x_2 can fit each other." In turn this agent is related to the one that detects the fitting of top-bottom and so on.

The queue manager prevents the circularity of postings, that is, an agent can only be accepted once.

6. CONCLUSIONS

The focus of this research is the conceptual stage of the design process. Design studies show that on this level the design process is not completely defined in terms of constraints that shape the final product, rather they are built along with the ongoing progress of designing. These claims inspired the hypothesis of the importance of the constructive approach to knowledge representation. A system that employs this paradigm in knowledge representation is presented and examined within a context of analogical reasoning.

6.1 Validation

In order to validate the hypothesis the same analogy engine is intended to run in two distinct modes: one when it is connected to the fixed represented database of familiar designs, two when it is inserted into the system as described earlier. It is expected that the runs of the engine in each mode should produce detectably different results. Especially, it is expected that the run that employs constructive representation should generate richer outcomes both quantitatively (more conjectures) and qualitatively (more novel or more surprising conjectures).

6.2 Application

The system presented here is considered to be a mind aiding tool. Its sole purpose is to provide the designer with a potentially novel analogical conjecture. Thus, this system can be seen as a sort of "brain storming" tool that the human designer can employ as an assistant. This system, although without the ability to understand the complexity and semantics of the design process, attempts to provide a framework for a tool that can behave as "a stupid savant" which requires human intervention and understanding to properly interpret and use the proposed conclusions. The system cannot solve design problems on its own, yet one can expect that by engaging the designer in a discourse throughout the whole design process, the resulting dialog should facilitate finding novel design solutions.

ACKNOWLEDGEMENTS

This work has been supported by an Overseas Postgraduate Research Award and by an Australian Research Council grant.

REFERENCES

Chalmers, D., R. French and D. Hofstadter, 1995, "High-level perception, representation and analogy: A critique of artificial-intelligence methodology", in D. Hofstadter , *Fluid Concepts and Creative Analogies*, Basic Books, New York, chapter 4.

Clancey, W. J.,1991, "Book review: Israel Rosenfield: The Invention of memory: A New View of the Brain", *Artificial Intelligence* 50, p. 241–284.

Clancey, W. J., 1997, *Situated Cognition: On Human Knowledge and Computer Representations*, Cambridge University Press, Cambridge.

Falkenhainer, B., K. D. Forbus and D. Gentner, 1989, "The structure-mapping engine: Algorithm and examples", *Artificial Intelligence* 41, p. 1–63.

Gentner, D., 1983, "Structure-mapping: A theoretical framework for analogy", *Cognitive Science* 7, p. 155–170.

Gero, J. S., 1990, "Design prototypes: A knowledge representation schema for design", *AI Magazine*, 11(4), p. 26–36.

Gero, J. S., and V. Kazakov, 1999, "Using analogy to extend the behaviour state space in creative design", in J. S. Gero & M. L. Maher (eds), *Computational Models of Creative Design IV*, Key Centre of Design Computing and Cognition, University of Sydney, Sydney, Australia, p. 113 – 143.

Lakoff, G., and M. Johnson, 1980, *Metaphors We Live By*, University of Chicago Press, Chicago.

Navinchandra, D., 1991, *Exploration and Innovation in Design*, Springer-Verlag, New York.

Nehaniv, C. L. 1999. "Computation for methaphors, analogy, and agents", in C. L. Nehaniv (ed.), *Computation for Methaphors, Analogy, and Agents*, number 1562 in Lecture Notes in Artificial Intelligence, Springer, Berlin; New York, p. 1–10.

Schön, D. A. & G. Wiggins, 1992. "Kinds of seeing and their functions in designing", *Design Studies* 13(2): p.135 – 156.

Suwa, M., J. Gero and T. Purcell, 1999, "Unexpected discoveries and s-invention of design requirements: a key to creativity in designs", in J. S. Gero & M. L. Maher (eds), *Computational Models of Creative Design IV*, Key Centre of Design Computing and Cognition, University of Sydney, Sydney, Australia, p. 297–320

Rosenfield, I., 1988, *The Invention of Memory: A New View of the Brain*, Basic Books, Inc. Publishers, New York.

This is a copy of the paper: Kulinski, J and Gero, JS (2001) Constructive representation in situated analogy in design, *CAADFutures2001* (to appear).