Playing Games: the Role of Computers in Sketching

Kari Rauhala

VTT Building and Transport, Finland
http://www.vtt.fi/rte/

Abstract. Why computers are not yet used in the early phases of architectural design? This question requires a closer examination of the sketching process itself. Looking it from the hermeneutical point of view, it becomes quite obvious that sketching really is and probably should remain the last fortress resisting the computerization of design. Sketching is an intimate dialogue between drawings and language. It is a dynamic and circular process of understanding. Also its intrinsic methods to search solutions cause wicked problems for computer programming. Nevertheless, computers could be used more for assisting sketching. Their proper role would be in validating tentative solutions.

Keywords. Sketching, hermeneutics, language, metaphor, computers.

Introduction

CAD programs have recently been developed more and more towards versatile 3D-presentations and towards the way how architects approach their design task. Nevertheless, at least in the early sketching phase they still prefer the traditional methods. In general, architects seem quite reluctant to use computers at all. Perhaps these never will totally replace the traditional methods. Could computers, nevertheless, have a proper role in enhancing the sketching phase? Before answering to this we need to consider more closely the phenomenon of sketching itself.

Hermeneutical philosophy, by opposing the common "Cartesian" view of thinking, can offer new interesting viewpoints to such phenomena. This approach is not new, however. Especially Adrian Snodgrass and Richard Coyne have made a substantial contribution to this (e.g. Coyne and Snodgrass, 1991; Coyne, 1992). Also Winograd and Flores (1986) have considered the limitations of the use of computers from the same point of view. However, the design process itself has still remained rather ambiguous in these writings. I myself have looked at it closer earlier, also from the hermeneutical point of view (Rauhala, 1992). At that time, however, the interplay of the drawings and the language as well as the crucial assembly of various domains of thinking in the design solutions were not yet fully clarified.

Concepts, drawings and the hermeneutic circle

Concepts, normally referred by words, are linguistic. They are generalizations of our contingent experiences. Concepts actually constitute our understanding and enable our thinking. They bring thoughts to mind. In designing these are usually visual images. However, these images have blurred edges. They depend highly on personal experiences, situation and context. When the context becomes better defined, also the images become clearer. On the other hand, when we understand what a drawing represents, this becomes conceptualized. A very schematic picture can bring to mind quite different thoughts

depending on the viewer, the situation and the general context. Actually, we always also complement the pictures in our minds depending on how we have understood them.

Hermeneutic circle is a circular understanding process between parts and the whole. This is not a vicious circle, but a continuously developing one. According to Martin Heidegger (1986) the process of understanding itself is a hermeneutic circle. In understanding we always have already certain preconceptions conducting it. On the other hand, our understanding will also change these preconceptions. Hans-Georg Gadamer (1979) emphasizes the role of history and tradition in our preconceptions. According to him understanding means a fusion of ones own horizon of understanding with a new unfamiliar horizon of a text or a picture, for example.

We can see the hermeneutic circle also working between drawings and concepts in the design process. In drawings we apply understanding of our task, but on the other hand, the drawings become also reinterpreted, which changes the understanding of the task itself. This principle is actually very analogous to Donald Shön's (1983) idea of "reflection in action". By drawing we are acting according to our understanding, which, in turn, leads to reflection i.e. to a better interpretation of the new situation.

Thought constellations, metaphors and intentionality

Various concepts emerging during the design process create specific associative networks, which, however, will continuously change, expand and even contract. All these more or less coherent clusters of concepts I will call as thought constellations. In the design process such are, for example, the functional principles and objectives, space requirements, constructional requirements and various artistic principles.

Metaphors have a very essential role in structuring our thinking. If we don't have readily available concepts for expressing our intentions, we borrow concepts from other totally different contexts. Metaphor means actually a transfer of descriptive concepts from a familiar context to describe phenomena in an unfamiliar one. These descriptions can actually become as real and true as the original ones. According to Lakoff and Johnson (1980) we also use structural metaphors, i.e. we borrow even larger concept structures for describing unfamiliar phenomena. For Gadamer (1979) the whole language is fundamentally metaphorical. The fusion of horizons is actually a metaphoric event: a fusion of a familiar already known horizon with a new and unfamiliar one. Metaphors can actually be seen as a special case of thought constellations. Ricoeur (1981) suggests that the emergent character of metaphors is the basis of our imagination. In architecture metaphors have been used in many different ways.

According to Heidegger (1986) all things in the world are primordially ready-to-hand for us. We understand them as for something. They are intentional. Accordingly, also our thoughts are intentional. This means that certain things are in accord with our intentions and some are not. Intentions can also be seen as rules, which, however, primarily are unconscious. They form our tacit knowledge. Many of them have, nevertheless, become explicitly formulated as laws, orders, norms, guidebooks etc.

Design process

John Archea (1987) has asserted that architects actually are not at all problem-solvers, but rather puzzle-makers. He justifies this by the fact that architects generally cannot explicate any desired effects prior to realizing them through the design process itself. The basic notion that archi-

tects don't start the design process with an analysis of problems and objectives to be fulfilled by the design holds true. In fact, architects sketch by doing repeated experiments, which they only afterwards validate. Thus, the process indeed seems to be opposite to the problem solving.

However, contrary to Archea, my thesis is that architects still ultimately are problem-solvers. I want to justify this by the fact that their experimentation is not a random process. It is conducted by a certain intentionality and is mainly progressive. The reasons for the experimentation lie elsewhere.

Firstly, architects use so many different and dynamic thought constellations, of which most even are unconscious, that it is quite impossible to go them all through systematically. The situation is similar to a chess player, who cannot beforehand go through all the possible move alternatives, but experiments with moves that seem to be promising according to earlier experiences and some pre-checking. Secondly, architects try to find solutions, where the various objectives inherent in the different thought constellations will be solved simultaneously. The reason for this is simply the fact that by solving one problem in time, one can aggravate other problems and even generate new ones. This situation resembles in a way the solving of a crossword. A skilful player or designer considers several constellations simultaneously. In certain familiar situations he/she will also apply earlier well-tried solutions or solution aggregates i.e. precedents. Nevertheless, design process differs from crossword solving in that that it has innumerable "right" solutions. During this search also the thought constellations themselves can be changed. Both these features of the architectural design situation lead to the fact that the only way to find comprehensive and satisfactory solutions is to make heuristic experiments with promising schemes for the solution.

How the architect knows that the puzzle has been solved, if most of the thought constellations are unconscious? The probable answer is that if the constellations are not satisfactory embedded in a specific solution, certain "variables" tend to generate mental tensions. Now in a good solution all these tensions become more or less "relaxed" and the solution feels good. Such an unanalyzed feeling is very familiar to all practicing architects.

Which kind of role computers could have in sketching?

I have earlier considered the inherent difficulties of using computers creatively in sketching (Rauhala, 1992). It seems impossible to use computers as a creative adviser or as a generator of totally new design solutions. Likewise using computers for generating new and creative associations seems to be in principle infeasible. Because our creative imagination has something to do with discovering completely new metaphorical connections, the basic problem in using computers for this quest is that computer programs always are already based on certain predefined familiar language horizons. There is basically nothing alien in them.

Could we perhaps randomly combine different program parts from a large number of independent program modules for trying in this way to create new ideas? It is quite obvious that this would still be only a faint and very simplified picture of the richness of our everyday world with a myriad of different and varying thought constellations. It would also be very frustrating to use such systems in practice. Further, in normal life the hidden intentionality also plays a crucial role in detecting appropriate metaphors. Thus, trying to use computers creatively seems to be generally impossible.

Nevertheless, this does not mean that computers would be useless in sketching. They must

only have another role in it. The strength of computers is their computing power. Although they are not helpful in creativity itself, they can be indispensable in technically validating new ideas. This validation can be easiest and most naturally done during the design itself i.e. within CAD working.

For releasing the designer from technical verifications interrupting the design process, CAD programs could do such verification automatically. Computer programs are apt to such technical constellations. However, it may be dangerous to include also non-technical thought constellations in them. These may rather hamper than boost the creative process. Anyhow, if, for example, certain norms concerning space requirements are used for controlling the design, these should also be possible to switch off.

Also the computer interfaces could be significantly developed for sketching purposes. 3D-objects should be easily and naturally manipulated in all their projections. It is also important that during sketching one can define objects generically so that only those properties, which are relevant in certain situations, shall be defined. On the other hand, all such properties should also be easy to remove, if desired.

When to shift to CAD working?

Drawings are projections. Nevertheless, in these projections architect "sees" actual three-dimensional objects represented. This makes the use of computers at the preliminary sketching quite tedious, because normally in CAD programs projections are derived from 3D-models. Thus, one has to first build the model, which then will be projected from various viewpoints. The imagined three dimensional objects "behind" manual drawings are much more flexible than 3D-models in CAD. These can even be derived from drawings, which would cause difficulties to computers. The

shift of the mental images is also very quick compared to CAD. One can also transform them deliberately and rapidly. Nevertheless, this could be done rather easily also by the computer, if there are ready made libraries available for it. However, if the transformation transcends the libraries, the task becomes much more tedious. This may also threaten imaginative sketching, because by suggesting ready made alternatives as the easiest solution computers actually hamper the search for totally new solutions.

The advantages of traditional methods are the possibilities to a spontaneous sketching and a rapid visualization. They also allow, at least at the beginning, a quick interplay of drawings and imagination. Their drawback is the slowness and toilsomness during the later phases of the process, as compared to CAD programs. Thus, the earliest phase for starting with CAD working is just after the first scaleless sketches. The latest reasonable phase for this is, when the design needs only a fine-tuning.

Conclusions

Sketching is not a systematic problem solving process with predefined objectives to be fulfilled, but a very complicated circular event, where both the objectives and the design itself are concurrently developing. It is a deep understanding process of linguistic ideas and their visualizations. This means that it is not reasonable to try to model this phenomenon to computers. That would always go one step behind the development of new ideas. Nevertheless, because sketching basically means doing experiments with tentative solutions, computer programs could be developed to facilitate it by validating these experiments. Also the CAD interfaces could be much better tailored for sketching.

References

Archea, J.: 1987, Puzzle making: What architects do when no one is looking, in Kalay, Y. E. (ed.) Computability of Design, John Wiley & Sons, New York, Pp. 37–52.

Coyne, R. & Snodgrass, A.: 1991, Is Designing Mysterious? Challenging the Dual Knowledge Thesis, Design Studies, 12 (3), pp. 124–131.

Coyne, R.: 1992, The Role of Metaphor in Understanding Computers in Design, ACADIA-Conference 92, Mission, Method, Madness, Clemson University, Charleston, South Carolina.

Gadamer, H.-G.: 1979, Truth and Method, Sheed and Ward Ltd., London.

Heidegger, M.: 1986, Sein und Zeit, Max Niemeyer Verlag, Tübingen.

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

Rauhala, K.: 1992, Computer Aided Architectural Sketching, in T. Oksala (ed.) EAI-symp'92. Special Focus Symposium of the 6th International Conference on Systems Research, Informatics and Cybernetics on the Environmental Design, Aesthetic Quality and Information Technology, Sym-posium Proceedings, August 19-20, 1992, Baden-Baden, pp. 51 – 56.

Ricoeur, P.: 1981, Metaphor and the Central Problem of Hermeneutics, in J. B. Thompson (ed. and trans.) Paul Ricoeur, Hermeneutics and the Human Sciences: Essays on Language, Action and Interpretation, Cambridge University Press, Cambridge.

Schön, D. A.: 1983, The Reflective Practitioner: How Professionals Think in Action, Basic Books, Inc., USA.

Winograd, T. and Flores, F.: 1986, Understanding Computers and Cognition: A new Foundation for Design, Ablex Publishing Corporation, New Jersey.