CAD and Creativity: Does the Computer Really Help?

Bryan Lawson

ABSTRACT

We are frequently told by its exponents that computer-aided design (CAD) liberates designers and gives them new ways of envisioning their work, but is this really true? CAD in architecture is examined to see to what extent it has enhanced creativity in design. This is partly done by applying a test of creativity advanced by contemporary architect Herman Hertzberger. In this analysis, CAD is found somewhat wanting, and some suggestions are made as to why this might be so.

Computer-Aided Design in Architecture

Computer-aided design (CAD) has now been with us in a practically useful form for perhaps a quarter of a century. This has generally been seen as progress towards a better way of designing architecture, and the exponents of CAD, including this author, have often argued that it should improve process and product in architecture. Sadly, remarkably little empirical evaluation of such claims has actually been carried out. There are many aspects of the process of design that might be affected by the introduction of computers. The ability to visualize in three dimensions and to simulate aspects of performance and the coordination and control of production information are all obvious examples. This article, however, restricts itself to the question of whether CAD has been demonstrated to enhance creativity in architectural design. It relies largely on anecdotal and experiential data and on analysis of the relevant characteristics of CAD. It finds CAD to be some ways short of supporting true creativity.

To develop this argument, I shall look at both architectural practice and education. I will from time to time rely on insights provided by some contemporary architects who are widely considered to be highly creative. In particular, I will rely on conversations I have had with Spanish architect and engineer Santiago Calatrava, who practices in France and Switzerland, and Dutch architect Herman Hertzberger. Both these designers have made deliberate decisions not to use CAD as a creative design tool. Hertzberger's thoughts seem of particular relevance and importance, since not only has he designed many famous and even seminal buildings, he has published widely throughout his life and now teaches at the Berlage Institute in Amsterdam. He has written one of the most important books of recent times about learning to design, *Lessons for Students of Architecture* [1].

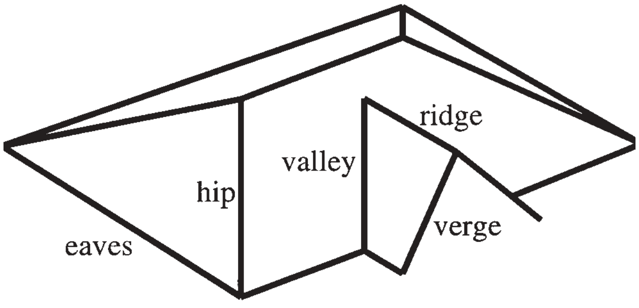
Central to the design process in architecture is the process of drawing. Drawings are used not only to communicate the results of architectural design to clients, users, legislators and constructors, but also, and more importantly here, as a central tool in the design process itself. Consider what Hertzberger has said about drawing during the design process:

> A very crucial question is whether the pencil works after the brain or before. In fact what should be is that you have an idea, you think and then you score by means of words or drawing what you think. But it could also be the other way round: that while drawing, your pencil, your hand is finding something, but I think that's a dangerous way. It's good for an artist but it's nonsense for an architect [2].

At first sight this position might appear to be different from Donald Schön's description of the way an architect "holds a conversation with the drawing" [3]. However, we can see that in reality Hertzberger seems to support Schön's view when he explains: "You are influenced by what you are doing . . . and sometimes inspired by a drawing . . . but don't let the pencil determine your thoughts, it must be the other way round" [4].

I think that Hertzberger is making one of many important distinctions to be found between art and design, namely, the extent to which image making is direct or indirect. In the visual arts, computers may now be used directly in the generation of images. Effectively, the artist who wishes to may now use the computer as an addition to the palette of techniques at his or her disposal. It is fascinating to see the kind of work that can be and is generated by artists and composers in this way. Often this work is both impressive and original. The Creativity and Cognition conference series has revealed that many artists believe that they now have creative opportunities that they would not have had without the computer. Of course, at least two conditions are necessary for this kind of creativity. First, the computer program must offer new possibilities, rather than simply aping existing ones. Second, and we must never forget this, the program must be in the hands of an artist who can be creative in the medium.

Fig. 1. Roof representation. The drawing shows the normal method of describing polyhedral roof form used by architects. This is quite unlike the language of now-conventional computer-based solid modeling software, which uses metaphors quite alien to architects.



Bryan Lawson, University of Sheffield, School of Architectural Studies, The Arts Tower, Western Bank, Sheffield S10 2TN, U.K. E-Mail: <B.Lawson@sheffield.ac.uk>.

An earlier version of this paper was presented at the third Creativity and Cognition Conference, Loughborough University, U.K., 10–13 October 1999, and published in the *Proceedings* (New York: ACM Press, 1999; ISBN: 1-58113-078-3).

In 3D design, this can never be the case. Here, the computer is used to help produce some intermediate statement that in some ways describes the final object or gives instructions as to how to produce it. In fact, there has always been a problem with design images in this process. They are not an end in themselves but often can become so. Indeed, design drawings and models are often exhibited in galleries as if they were art objects. The Royal Academy Summer Exhibition in London, for instance, has a gallery for architecture. This serves to increase confusion about the relationship between art and design.

Integration and Design

In parallel with the development of CAD tools over the last quarter of a century, my research group has also continued to study the design process itself. In fact, we have made considerable progress in understanding design as a process, though much still remains to be done [5]. Several key attributes of the design process seem important in the development of CAD: First, the overall complexity involved, in terms of the sheer number of different and ultimately irreconcilable dimensions of architectural problems. We know that architecture is expected to look beautiful, and hopefully to add meaning to and express the rituals of our lives. We expect it to facilitate our relationships with others. We expect it to be technically sound and safe, to provide shelter and appropriately shelter us from the climate. The list could easily be further extended. However, what we also know about design is that this vast array of problems does not neatly suggest design solutions. The point of good design is its integrative function. Good design is holistic. This principle is of course neither new nor exclusive to architecture. George Sturt's *The Wheelwright's Shop* showed it to apply even in vernacular design [6]. Sturt showed how the simple device of making cartwheels dish-shaped solved many problems, some relating to maneuverability, some to structural stability, some to legislative rules and so on. In architecture, even the humble window simultaneously has to afford a view and yet maintain privacy, provide daylight and yet keep out unwanted solar gain, maintain the weather enclosure of the external skin and yet often provide a means of natural ventilation, and of course play a major role in the visual composition of the façade.

Resolving all these issues requires tremendous mental alacrity. To do this creatively and in an original way requires the most sophisticated cognitive skills. Again, I find the words of Herman Hertzberger help to clarify this point. Here, he describes the problem of merely designing the entrance to a school:

> The problem is that you have certain moments when many children have to [pass through], the problem also is that sometimes you have a small number of people waiting, the problem is that sometimes it rains and then it is not very nice to sit there. The problem is . . . and so on. So you get this whole list of things that altogether represent the problem. And then you say well given all these things, the stair should not be too small, should not be too large, it should be covered over, it should not be . . . and so on. There are always these contradictions. This for me is creativity you know, finding solutions for all these things that are contrary, and the wrong type of creativity is that you just forget about the fact that sometimes it rains, you forget that sometimes there are many people, and you just make beautiful stairs from the one idea you have in your head. This is not real creativity it is fake creativity! [7]

It has seemed to many of us likely that computers could and should be able to help us with this mental conjuring trick. In this sort of design, it is simply so easy to lose sight of one or more of the many issues under consideration and produce an imbalanced design. Designers worry about this greatly. Designers liken this effort to juggling—the need to keep many balls in the air at once. It is also notable that many good designers like to draw on small sheets of paper when doing creative design work. This would seem to reflect their need to keep everything within their field of vision. In fact, they employ any devices they can to assist in the very intense periods of concentration required. Of course, we also know that these periods are interspersed with long periods of more reflective thought and even background thought. Margaret Boden's *The Creative Mind* shows this to be a common feature of creative processes [8].

CAD Can Help Creativity

So why use a computer to aid in all this? Well, the argument goes, computers are simply much better than humans at some tasks. They are both quicker and more reliable at calculation than we are; they are also quicker at looking up information, for example. Potentially, they do not forget things, as we do. Certainly, they are also much worse than we are at other tasks. They are poor at recognition, interpretation and the reconciliation of conflicting demands. One of the most famous cinematic computers, HAL in *2001,* demonstrates how their inflexibility can lead to errors that humans would never normally make.

It is commonly thought that architectural design is done largely by drawing, and that we experience architecture far more through our visual senses than through any other. This is actually rather misleading, however, since in reality much architectural design is actually done through conversation both between members of the design team and with their clients and others [9]. But because that conversation is not recorded and the drawings are, we tend not to notice its importance. In fact, this importance has largely been neglected by those who design CAD systems, too. The fact is that most computer applications intended to assist with architectural design involve an intensely graphical process. So where is the proof now after a considerable number of years of using CAD in architecture that it actually enhances creativity? Three simple examples, one each from education, practice and research, should encourage us.

First, a piece of experiential evidence: One student studying during the early days of CAD completed his first degree in architecture but was not given a sufficiently high mark to begin work on a second degree. His design work was functionally and technically competent but dull, unimaginative, formulaic and unattractive. He went to work for a year at a firm of architects using the CAD system developed by my own research group. He re-applied to the degree program, submitting excellent design work, was admitted and went on to win a final-year prize. In talking to him and looking at his work, I came to understand what had happened. He had a good 3D imagination but was poor at drawing and had therefore restricted himself to designing only forms and spaces that he could feel confident in drawing. The computer, however, had enabled him to represent much more complex forms, and he had flourished once he learned to master it.

Architect Ian Ritchie used the same CAD system to help design his innovative gallery in the Natural History Museum in London. He claimed that he was able to generate a more sophisticated 3D form for this than he would have been able to in the same time using conventional manual drawing and that he would not have tried to use such complex forms without the CAD system [10].

Finally, in an interesting experiment, Robert Aish showed how a very simple CAD system could be used by non-architects to design buildings [11]. He also showed that the designs of a children's nursery produced by nursery school teachers using CAD were rated more highly by a panel of other nursery school teachers than designs produced by architects. This certainly suggests that the creative imagination of non-designers could be unleashed by CAD, which effectively de-skilled the drawing process and certain simple evaluative procedures, so that designers could express and explore ideas that their own drawing skills could not support.

CAD Can Inhibit Creativity

These three examples come from many years ago, and there has still been no major systematic investigation of the impact of CAD on contemporary design creativity. It is largely accepted on faith to be helpful, which is simply not good enough. There is plenty of evidence that CAD is now widely used in the profession, but that is another matter altogether. Certainly, there is some evidence to support the idea that CAD now holds many advantages over manual drafting techniques involving project management and data coordination. However, there seems to be a growing body of experiential and anecdotal evidence that CAD might conspire against creative thought.

Many highly rated architects do not themselves use computers. Thus, while Ian Ritchie creates innovative structural form with CAD, Santiago Calatrava also creates highly original structural form, but does not use CAD. He uses CAD for finite element analysis but uses physical models for form generation. This is remarkable for two reasons. First, Calatrava is both an architect and an engineer and is clearly highly numerate and used to working with computers. Second, he creates adventurous forms that are too difficult to draw by hand, but he prefers making physical models to using computers. Many architects who would be viewed by their peers as creative contributors to the field have expressed concern about using CAD for design. Of course, this could be seen as a purely Luddite tendency. Let us again consult Hertzberger on this:

> It took me forty years to find a way to have this communication [between] my brains and my paper . . . and I feel I'm now quite eloquent in my way of doing it. I need all my energy for my design, and I decided [not to learn to use CAD] like I decided not to learn the violin at my age [12].

Hertzberger may have his tongue slightly in his cheek here, but he is still making an important point. The medium through which a designer represents thought is central to his or her work process. Hertzberger has found a satisfactory way of working and cannot see benefits sufficient to outweigh the obvious burden of learning the new technique.

But the most worrying recent evidence regarding CAD is that of student work. Remember that what students are doing in their work is developing their design process techniques. Increasingly, we have students learning from scratch to design with CAD rather than the manual drawing technique on which Hertzberger was brought up. Amongst those of us who examine such students, there is a growing feeling that a worrying trend is developing. Over the last few years, I have examined design in half a dozen universities in three countries. In each case, I found examples of students combining impressive and convincing computer presentations with poor design. Such pieces of work seem to me to be exactly what Hertzberger meant when he talked of "fake" rather than "real" creativity. They may look extremely convincing, they might be original, but they are most certainly not good design.

Theoretically, it has always been possible to find excellent presentation combined with poor design. However, before the advent of CAD, it seldom happened in practice. This is probably because the visual sensitivity needed to design and to draw well are so similar that it would be unlikely for a student to be skilled in one area but not in the other. Not so, it seems, with CAD and architectural design. This is exacerbated by other factors. First, we live in such a televisual age that any information that looks televisual is automatically considered authoritative. Second, the complete mastery of such computer systems is still sufficiently rare and novel that we tend to admire it in the way we admire an animal taught to do tricks. So it is possible to put forward computer presentations that look attractive and even dazzling, that seem authoritative, while the architecture so represented is really quite awful. A number of critics agree that we have recently seen national prizes given for work that prior to CAD would have stood no chance of attracting a design prize.

What Is Going Wrong?

Why is the work itself so bad? Is it possible that we have failed to notice that such CAD systems are not neutral in this process, but that they actually encourage poor design? Of course, the software cannot intend anything, and certainly the developers did not intend such results, but the effect remains. This phenomenon can also be illustrated with reference to the graphic design that proliferated with the advent of the Apple Macintosh computer. We started to see documents obviously produced by people who owned a Mac, with its (at the time) revolutionary what-you-see-is-what-you-get interface and multiple fonts. These documents characteristically contained as many fonts as possible, apparently merely in order to exploit this new and wonderful facility. Of course, individuals trained in graphics would not make such a mistake, knowing that the quality of a design is normally unlikely to be improved in this way. Today we have this problem in three dimensions. A small but significant and growing proportion of student work all seems to exhibit similar common characteristics. Because it is possible to produce a certain kind of 3D form in a CAD package, the student does so, bypassing that critical visual editing faculty that we try to inculcate in design schools. Even worse, some of these forms are relatively easy to generate in CAD but are hard to represent in manual perspective—for example, shell forms based on ellipsoidal sections, rotations of curved parabolic forms and so on. Perhaps this encourages students to believe that because they have drawn something infrequently seen, they are being creative.

Another problem is that the software is usually a generic 3D package that can only handle form in the abstract—it does not address or comprehend the construction or materiality of the objects represented. Contrast this again with the work of Santiago Calatrava. Many of his original design drawings are freehand watercolors, but are approximately to scale. His staff, after they have applied all their sophisticated engineering software to these forms, often calculate them to be very near the size his original sketch showed. Calatrava knows about materials and their strengths, weaknesses and structural characteristics. Now, it is quite possible today to take a position about architecture that relegates structure to a purely supporting role. Some may even argue this is how it should be. Others would be at odds with this idea. But what view one takes of architecture is unimportant here. What is of concern is that the computer system, rather than a person with a philosophical position about architecture, is setting the design agenda.

First, that must be a bad thing in itself. There are many issues that can rightly claim a place at the front of the queue when it comes to defining architectural form. Over the years, these have included geometrical proportional rules, the values the design symbolically expresses, the primary functional requirements of the building, the technology used to construct it, the context supplied by its surroundings and so on. Completely absent, and appropriately so, from such a list are any influences from the design process itself. To allow a computer, only latterly a participant in that process, to come to the fore seems inappropriate. It has no relevance to the lives of the people who will relate to the building throughout its life. While I myself find the design process fascinating, the architectural experience must nevertheless be ultimately all about product.

Second, it is worrying that this effect occurs without our intending it or even noticing it. Traditionally architects advance theoretical positions about the influences on their work and explain the reasoning behind their position. While it is certainly true that there is a great deal of post-hoc rationalization in such material, it also true that at its best it forms the very heart of the discourse through which architecture advances. If the computer is allowed to creep in unnoticed, then the whole debate is undermined.

How Can the Computer Get in the Way?

The problem is that if the computer uses the wrong metaphor for describing design features, it can inhibit the creative integration that design requires in order to be what Hertzberger calls "real" as opposed to "fake" creativity. Creative designers often are able to work with multiple or parallel lines of thought, each of which involves its own design features. Such a phenomenon poses a problem for the development of CAD interfaces [13]. Here I shall study this problem in more detail by examining just one significant mode of thought that is commonly found in use by architects but missing from many CAD systems that they might try to use.

In our efforts to develop software that would enable architects to describe and manipulate roof forms, we discussed existing software with architect colleagues and found general unhappiness with the commonly available solid modeling metaphor for the interface. Excellent work had been done to identify and understand these forms and relate them to normal design procedures, such as the building of physical models [14]. However, the building of a physical model is often itself a diversion from the creative thinking necessary to the design processes. Using software based on this type of metaphor, the user builds a complex 3D form by starting with a limited number of primitive geometric shapes, such as pyramids, wedges, blocks and so on. These are then operated upon by a series of collisions, modifications and the like, which relate to imagined real-world operations in a physical model shop.

The reason for architects' unhappiness with this process gradually became clear. Architects simply do not normally imagine their roof forms in this way. Using such a system requires them to translate their thoughts into the language of the computer before they can enter data and modify it. Architects characteristically model roof forms using well-established and -understood spatial concepts. In particular, architects have a comprehensive language for the folds and edges that can be found in pitched roofs. The concepts of "eaves," "verge," "ridge," "hip" and "valley" enable all pitched roof forms to be described. The "eaves" is a bottom edge and is normally but not always horizontal. A "verge" is a sloping edge such as is normally found in a gable. A "ridge" is a horizontal convex fold, and a "hip" any non-horizontal convex fold. A "valley" is any concave fold (Fig. 1). Architects think and speak in these terms; one would be unlikely to hear them talking of collided wedges, pyramids and blocks.

Why do architects use this particular way of imagining their roof forms? There seem to be two possible explanations, both of which are of interest. First, architectural roofs are not solids at all, but contain voids. In many, though not all, buildings, these voids are an important part of the spaces that the building exists to provide. In particular, the spaces below the roof can not only offer usable floor space, but also can be used to express form in a most interesting way. One has only to stand in a cathedral or a mosque to see the significance of this. Second, architecture is not simply abstract geometry but must be constructed out of real materials. This poses many problems, but there are two major ones worth considering. These are how the structure holds everything up, transfers loads to the ground and remains stable, and how the roof keeps out the external environment and in particular disperses rainwater.

Why does the language of folds help here? In terms of internal space, it is the underside of the roof planes and their bounding edges and folds that visually define the spaces. These folds and edges are where the main structural elements are likely to be and which will need support from below. The pattern of these folds is used to collect and transfer rainwater. In more detail, each type of fold or edge poses a particular set of construction problems, which are well understood in principle but which must be interpreted in each case. Each type of roofing material usually has a minimum and sometimes a maximum pitch at which it must be used in defining roof angles.

What this illustrates is that architects must ultimately think about their building not as an abstract solid form but as a collection of voids enclosed by surfaces that must have physical constructions. Design is very much a matter of integration, and so these issues need to be in the mind of the architect early in the process [15]. Experienced designers know what sort of problems are likely to be involved and therefore intuitively and habitually use appropriate forms of representation.

Our team eventually developed software that used these concepts of roof planes having a pitch angle and direction, and surrounded by edges or folds that must be eaves, verges, ridges, hips and valleys. Now our first requirement was met, which was that the software use a representational system commonly used by actual designers themselves. But the motivation to use a CAD system must surely be that it adds some value to the process. In this case, our software also would go on to solve the equations of the 3D geometry of the resultant intersecting planes and complete the form absolutely accurately. It could also alert the architects when the information was lacking in some way. This might be because it was incomplete and the geometry remained undefined or because there was conflicting information, allowing for more than one single interpretation [16].

To do this work, however, the software needed to have prior knowledge about the concepts being used. For example, knowledge about the meaning of the various types of edges and folds in roof planes, and about the rules that govern their behavior, was essential. Having this information, the computer could begin to interact like a genuine member of the design team, to understand the language and make inferences about information in order to produce new or revised information. We have relatively simple technology today, in the form of object-oriented programming techniques that allow us to embed such knowledge in the data about objects held in a computer.

We might expect to see this sort of software become more commonplace. Unfortunately, there are other market forces at work that make it hard to create, develop and commercially maintain software for such a restricted clientele as the architectural profession. Increasingly, architects are using generic software that can address form only as abstract geometry. It is beginning to look as though CAD is by no means a neutral tool. Like all tools, it suggests being used in a certain way. This threatens to set an agenda for architecture that is unhealthy and irrelevant.

Where Do We Go from Here?

So what do we do about this? First, we need more research into the effects of CAD on design. Since Aish's early work, few studies of significance have been published that ask critical questions about these tools and their effects. If one attends the conferences of those working the field or reads their proceedings, one finds almost no critical evaluation of the tools being described. The atmosphere of such conferences as the American Collegiate Schools of Architecture (ACSA), European Computer Aided Architectural Design Education (ECAADE) and Computer Aided Architectural Design Research In Asia (CAADRIA), for example, is like that of a religious gathering. The faithful come together to reinforce their common belief in the wonderful benefits of CAD. There are without doubt many benefits to be gained from using computers in architectural design. However, we have not fully demonstrated that it is universally promoting what Herman Hertzberger would call "real" as opposed to "fake" creativity.

References and Notes

1. H. Hertzberger, *Lessons for Students of Architecture* (Rotterdam, The Netherlands: Uitgeverij 010, 1991).

2. B.R. Lawson, *Design in Mind* (Oxford, U.K.: Butterworth Architecture, 1994).

3. D.A. Schön, *The Reflective Practitioner: How Professionals Think in Action* (London: Temple Smith, 1983).

4. Lawson [2] p. 38.

5. B.R. Lawson, *How Designers Think* (Oxford, U.K.: Architectural Press, 1997).

6. G. Sturt, *The Wheelwright's Shop* (Cambridge, U.K.: Cambridge Univ. Press, 1923).

7. Lawson [2] p. 42.

8. M. Boden, *The Creative Mind: Myths and Mechanisms* (London: Weidenfeld and Nicolson, 1990).

9. B. Lawson and S.M. Loke, "Computers. Words and Pictures," *Design Studies* **18** (1997) pp. 171–184.

10. Lawson [2] p. 91.

11. Lawson [2] p. 38.

12. R. Aish, "Prospects for Design Participation," *Design Methods and Theories* **11** (1977) pp. 18–27.

13. B.R. Lawson, "Parallel Lines of Thought," *Languages of Design* **1** (1993) pp. 357–366.

14. M. Van Norman, "A Digital Modelshop: The Role of Metaphor in a CAAD User Interface," *Design Computing* **1** (1986) pp. 95–122.

15. See, for example, the work of Chuck Eastman on watching designers try to create bathrooms. C.M. Eastman, "On the Analysis of the Intuitive Design Process," in G.T. Moore, ed., *Emerging Methods in Environmental Design and Planning* (Cambridge, MA: MIT Press, 1970).

16. J.P. Riley and B.R. Lawson, *RODIN: A System of Modeling Three Dimensional Roof Forms* (Brighton, U.K.: IPC Press, 1982).

Bryan Lawson is Dean of the Faculty of Architectural Studies at the University of Sheffield, U.K. He is both a qualified architect and a psychologist. He leads the "processes" research group at Sheffield and has studied the design process and published widely on the nature of design thinking.