Seeing through the Machine: Creativity, Visualization, and Computer-Aided Drafting

Ann Johnson
University of South Carolina

In 1969, Dr. H. G. Conway, a former president of Britain's Institution of Mechanical Engineers and the managing director of Rolls Royce's Bristol Engine Division, gave a lecture at Heriot-Watt University in Edinburgh. The substance of his lecture, titled "Creativity and Innovation in Engineering" consisted of an impassioned plea to value engineering creativity, largely by emphasizing connections between engineering creativity and drawing.[1] To Conway, creativity in engineering consisted largely of visual dimensions, including visual imagination, rather than the rational intellectual processes he saw being emphasized in engineering education. But ironically, in 1969, Conway's comments were made as engineering stood on the brink of a major endeavor to shift its visual nature. Computer-aided analytical tools, such as finite element methods, had already been introduced in the 1950s, and graphics software would enter engineering work in the 1970s.[2] In the 1980s and 90s graphical tools escalated in both their utility for engineering design and in their market saturation.

These new practices, dependent on the co-evolution of new software tools and increasing computing capacity, significantly changed both the social and epistemological dynamics of engineering. Given the widespread agreement about the centrally visual nature of engineering creativity, computer-aided drawing tools also bore on the creativity of engineers. While many have argued this effect to be negative, most eloquently Eugene Ferguson in *Engineering and the Mind's Eye*, here I argue more this change cannot be seen as simply a deficit, but rather must be perceived as a profound change in the nature of engineering practice, including its creative dimensions. While many of Ferguson's and others' attacks on computer-aided design and drafting have been valid in themselves, the computer's overwhelming adoption is also telling. If hand methods of drawing can be seen as a part of a paradigm or disciplinary matrix, qua Kuhn, then a shift to a new paradigm of computer-based practices must be explained in terms of a new ability or capacity offered by computer tools—new paradigms are only taken up when they solve problems which have become pressing on a community. As Kuhn and Edward Constant have shown, shifting paradigms proves to be a messy business, which, while in process, results in a deep rift within the community in question—many of whom oppose the new ways of knowing.[3] However, now that the shift to computer-based engineering design is largely complete, we can see both the incentives and detriments of the shift to computers in engineering more clearly. As a result, in seeing only the losses, Ferguson and other attackers actually underestimate the nature of the shift to computer-aided engineering. As a result, I claim here that the benefits, as well as the deficits, profoundly reconfigured the way engineers see,

---

[1] H. G. Conway, "Creativity and Innovation in Engineering" 1969 Betts Brown Memorial Lecture at Heriot Watt University, 21 January 1969, published in *Creativity and Innovation: The Heriot Watt University Lectures* (Edinburgh: Heriot-Watt University, 1969), 27-40.
[2] For the introduction of computer-aided engineering analysis, see Ann Johnson, "From Boeing to Berkeley: Civil Engineers, the Cold War, and the Development of Finite Element Analysis" in M. Norton Wise, ed., *Growing Explanations: Historical Perspectives on The Sciences of Complexity*, (Durham: Duke University Press, 2004).
[3] Edward C. Constant, *The Turbojet Revolution* (Baltimore, Johns Hopkins University Press, 1980). Thomas Kuhn, *The Essential Tension* (Chicago, University of Chicago Press, 1984).

affecting the social, epistemological, and creative dimensions of engineering design. After a full generation of engineers had been trained and practiced in computer-aided design, this new way of seeing, and implicitly of knowing, amounted to no less than incommensurability between the computer users and their non-computer-savvy predecessors.[4]

I. The Nature of Engineering Creativity

Creativity is clearly a longstanding concern in the history and philosophy of engineering, though it has long resisted definition and analysis. Part of the difficulty in categorizing and breaking down engineering creativity stems from its multidimensional character—engineering creativity means many things to many different engineers. Many of the more famous investigations of creativity have focused on defining its many dimensions. Arthur Koestler, in his 1964 study *The Act of Creation*, highlights the irrational and unpredictable character of creative thinking, arguing that creativity depends on thinking on multiple levels simultaneously.[5] Others, including Herbert Simon in *Sciences of the Artificial*, work to define creativity in such a way as to make it replicable. In this pursuit, Simon seeks to break down creativity into characteristic qualities, such as novelty, unconventionality, persistence, and emergence from an ill-defined problem.[6] A third approach has been to focus on the social dimensions of creativity, such as Mihalyi Csikszentmihalyi's socio-cultural model. Csikszentmihalyi's analysis focuses on the process by which creativity is recognized and validated by appeal to larger social and cultural factors—thus situating creativity in a particular historical and cultural milieu.[7] There are common elements in these disparate accounts of creativity—from incorporating novelty to determining the kinds of problems most likely to spawn creative solutions.

Conceptualizing creativity in engineering is further complicated by its central relationship to engineering design—a concept which has proven even harder to constrain. While most scholars of engineering place design activities at the intellectual center of the discipline, few have been able to illuminate engineering design except through a method of case studies as exemplified by Walter Vincenti's *What Engineers Know, and How They Know it*. For Vincenti, the critical and often overlooked dimension of design is its knowledge-producing capacity. He writes, "Day to day design practice not only uses engineering knowledge, it contributes to it."[8] This epistemological characteristic has strong implications for the issue of creativity in engineering design, since means that engineering designers work in a loosely or roughly defined field rife with uncertainties and unknowns, requiring continual shaping and modification—just the kind of ill-defined environment that Koestler and Simon thought most contributed to creative thinking. Louis L. Bucciarelli's notion of engineering design is related to this in that Bucciarelli argues engineers are collectively involved in this on-the-fly reconception process, and thus the social arrangements between them, their interested positions, and limited visibility, in both conceptual and literal senses, must be accounted for.[9] Bucciarelli explains this social and epistemological dynamic using

---

[4] See Ian Hacking, *Representing and Intervening* (Cambridge: Cambridge University Press, 1983) 68, for an account of incommensurability based around ways of knowing the world, as opposed to Kuhn's theory-based incommensurability.

[5] Arthur Koestler, *The Act of Creation* (NY: Penguin, 1964).

[6] Herbert Simon, *Sciences of the Artificial*, 2nd Edition (Cambridge: MIT Press, 1981).

[7] M. Csikszentmihalyi, "Society, culture, and person: a systems view of creativity" in Robert Sternberg, ed. *The Nature of Creativity* (Cambridge: Cambridge University Press, 1988), 325-339.

[8] Walter Vincenti, *What Engineers Know and How they Know it* (Baltimore: Johns Hopkins Press, 1990).

[9] Louis L. Bucciarelli, *Designing Engineers* (Cambridge: MIT Press, 1994).

his term "object worlds," explaining that engineers view the large systems they're designing through the artifact component they're most closely working on or tied to. As a result, all engineers in the design process have somewhat different understandings of the system and the design process. Creativity arises in this field of mixed understandings.

Bucciarelli's notion of object worlds also highlights another important dimension of creativity for engineers—the visual dimension of it. Conway, the Rolls Royce engineering referred to at the beginning of this paper, claimed, in his lecture on engineering creativity, "Creativity is an engineer is an ability or aptitude, probably basically innate, which allows him to think of, dream up, visualize, or imagine new or unusual solutions to problems."[10] Conway's definition clearly implies a visual and minds' eye dimension to creativity. This point is made even more strongly by Eugene Ferguson, who writes,

> In order to produce a new machine, structure, or other technological artifact, two separate but closely related processes are generally required. In the first, engineering designers convert the visions in their minds to drawings and specifications. In doing so, they solve an ill-defined problem that has no single "right" answer but has many better or worse solutions. Engineers learn a great deal during the process of design as they strive to clarify the visions in their minds and seek ways to bring indistinct elements into focus.[11]

Here Ferguson is using the axiomatic language of creativity in discussing ill-defined problems, but the practices of the engineers he describes are oriented toward the visual and toward communicating visually. Rudolf Arnheim argues even more strongly for a visual concept of creativity, by claiming flatly that visual things cannot be expressed in words, a point echoed by Richard Feynman.[12] For Arnheim, looking at the creative dimensions of art, verbal analyses could actually paralyze intuitive creation, a concern Ferguson shares for engineering design.

Given the emphasis on visualization in creative design for engineers, postulated by both practitioners and scholars of engineering and creativity, it seems only natural to look at the historical process of developing and adopting a new visually oriented set of tools and practices to understand how creativity works in both social and epistemological ways, emphasizing its dynamic of changing over time. This is not to say that creative works of the past somehow lose their creative edge, since as Csikszentmihalyi argues, creativity also requires an historical and cultural context. It is, however, to argue that given changing engineering practices, the routes by which creative thinking can occur can vary significantly. In the last century, nothing has changed engineering practices--that is what engineers do on a day-to-day basis--to the same extent, and across as many disciplines and fields, as has computer-aided engineering.

The changes I focus on here began in the 1980s, although as this shift hardly happened overnight, given its social and epistemological significance. Today, in 2004 the shift in nearly complete; few firms designing anything—from engineered products to architecture to interior design—use "hand" drawing for anything but conceptual sketches.[13]

---

[10] Conway, 29.

[11] Eugene Ferguson, *Engineering and the Mind's Eye* (Cambridge: MIT Press, 1992), 3.

[12] Rudolf Arnheim, *Art and Visual Perception* (Berkeley: University of California Press, 1954). Richard P. Feynman, *Do you care what other people think?* (New York: 1988), 54.

[13] This is not to dismiss the importance of conceptual sketches, especially in light of the role they play in creativity. Many articles in the journal *Design Studies* focus on the importance of sketching and the difficulty of

But the timing of the shift also begs for historical inquiry. The move to computer-aided design required the evolution of new technologies, both software and hardware. However, information technology, both hardware and software, was not impersonally delivered to an anonymous, generic user. Rather, the users were actively involved in the development of the hardware, and particularly the software—by developing new uses and techniques, as well as requesting and even demanding particular features and capacities from the information technology. Therefore, as the community of users expanded over more than a decade, the technology developed to better fit the demands of the users. Simultaneously, the growing number of users also developed groups to both share techniques using the new technology, but also to guide the development of the technology through informing the companies producing the software and, at times, pressuring them to make particular kinds of changes. This dynamic is a complicated one with a number of individual and corporate actors, hierarchies, and competing interests. But part of the initially slow adoption of computer-aided design is largely attributable to the co-evolution of the technologies, users, and skills. As a result of this co-evolution, the speed of adoption of computer-aided design has not been regular, instead escalating with the increasing utility of the software and size of user communities.

The complex historical development of technology and use is mirrored by the complicated changes in the social relations between different people involved in the production of designs. Usually seen as the "deskilling" of the draftsperson, as described in Kathryn Henderson's *On Line and On Paper*, the adoption of computer-aided design tools did more than eliminate the intellectual and company space of the draftsman, it greatly reconfigured the social process of design.[14] Drafters lost their positions, as computer programmers then computer draftspeople came into the corporation. As Henderson shows, the knowledge of the draftsperson was lost in many cases; the computer programmers and computer drafter simply did not have the same knowledge and did not play the same role in the design process. But at the same time, knowledge was gained, not just of computers but of the kind of information computers can produce. Computer-aided drawing has facilitated new design practices, which bear directly on the nature of engineering creativity in the production of new artifacts. For example, the ease of making changes on the computer means designs remain flexible for a longer period—in some sense expanding creativity from the conceptual design phase into the more detail-oriented fixing of the design.

Fitting this new knowledge into an intellectually and socially reconfigured process of designing has taken some 25 years. In some cases, computer-aided tools have made it possible for engineers to produce all their own drawings.[15] But this doesn't mean that collaboration ended. Rather the nature of collective work and of the collaborators themselves changed. Once, the drawing, as an artifact itself, changed from something produced by hand, it reflected the collective design work of the engineer and draftsperson. As a computer-plotted artifact, it reflects the collaboration between several engineers, all of whom can work relatively seamlessly on the same drawing—or least the same file which will be used to produce a plotted drawing.

---

using computers for sketching and conceptual design. However, my focus here is on the area where CAD has been widely, if not universally, adopted, that is, for formal drafting and mechanical drawing.

[14] See Kathryn Henderson, *On Line and On Paper* (Cambridge: MIT Press, 1998).

[15] There seems an obvious parallel here to changes in the office workplace in the face of office "productivity" software. Positions that were exclusively secretarial, such as typists, have vanished, while at the same time, far more middle managers and bureaucrats are expected to produce their own documents.

In addition, drawings play a crucial role mediating between engineers and the production floor. In the days of a drafting room, drafters played a crucial role in negotiating design details between the design engineers and the production engineers. Without traditional draftspeople, engineers using computers to produce their own drawing would have to include details on those plates that previously they relied on traditional drafters to provide. This expansion of responsibility for the engineers was another area which lay behind the rocky shift from traditional to computer-aided drafting. But, once again, admitting the loss of a collaborative but hierarchical relationship between design engineer, draftsperson, and production engineer, the new system has also generated new means of collaboration. Again, because the computer file facilitates collaboration on the same drawing, design engineers can now give the file to production, where another generation of changes can be made on the computer before plotting. On the whole, the functions of the traditional drafter have been relocated, not eradicated—acts of negotiation must continue, though the negotiators' titles change.

Furthermore the hierarchical collaboration that Henderson depicts is a corporate model, requiring a firm of a minimum size. What about the hundreds, or thousands, of firms too small to have such a hierarchical arrangement of labor? John Walker, the entrepreneur who oversaw the development of AutoCAD, claimed, "the United States had 600,000 manufacturing enterprises, 85% of which had 10 or fewer employees and did all their drawings manually." [16] By the turn of the 21st century, these firms too relied on computer-aided drafting software, but their process of adoption involved shifts to computer software within an individual's work practices, rather than the kind of social reconfigurations Henderson describes. To these users, I also ask how the computer has changed the way they think.

II. Making AutoCAD and its Community of Users

The leading professional drafting software is a package called AutoCAD, produced by Autodesk.[17] AutoCAD was introduced in 1982 and quickly became an overwhelming market leader. By 1987, when the shift to computer aided drafting was just beginning to get underway, Autodesk had already sold 100,000 copies of AutoCAD.[18] As a result, while AutoCAD is certainly not the only option in the now competitive market of drafting software, its success in colonizing the field has been impressive. Since it would be impossible to examine the development of all of the different drafting software packages, as well as the development of their user bases, I will focus here solely on the market leader.

On January 12, 1982 John Walker, the head of Marinchip Systems, a company in the San Francisco Bay area that sold complete computer systems, wrote a business proposal.[19]

---

[16] Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* (Cambridge: MIT Press, 2004), 245.

[17] John Walker, Autodesk's founder who is now retired, has created an online archive, called the Autodesk file, of Autodesk memoranda and other official documents and correspondence at www.fourmilab.com/autofile. It is downloadable in both Adobe Acrobat and Postscript formats (so it can be printed as a 900+ page book), as well as being viewable as an indexed website. Here, I refer to the URL describing each particular document Walker has posted, giving the author, title, and date of the original document where available, just as one would for a traditional archival source.

[18] John Walker, "Cosmic Perspective" June 29, 1987. http://www.fourmilab.com/autofile/www/chapter2_58.html

[19] John Walker, "Marin Software Partners Working Paper," January 12, 1982, http://www.fourmilab.com/autofile/www/chapter2_2.html#SECTION0020000000000000000.

Walker felt that there wasn't much future in selling and supporting computer systems. There was simply no way to compete with the advertising and support capabilities of large corporations like IBM and DEC. Walker wanted to get out of the computer system business and venture into the software business. He wrote, "the software business is very different. First of all, a software package can be produced out of pure effort, with only the capital needed to finance the machine and pay the programmer. Unlike hardware, the big vendors of mass market machines are utterly ignorant regarding software, and manufacturing software is as simple as copying disks."[20] Walker's proposal was to set up a company called Marin Software Partners (MSP) that would allow a group of programmers to buy in with their own money and then, usually as moonlighters, produce code for new programs that MSP could sell and support. MSP would develop and sell many different kinds of software, driven as much by programmers' interests and whims as by the demands of the marketplace. Walker dreamed up this plan at the end of a moment in the history of software than Martin Campbell-Kelly and William Aspray have called software's "gold rush era," when hundreds of software start-ups were formed.[21] This crucible of software company formation happened between the late 1970s and early 80s and was a period of great opportunity and growth, when financial barriers to entering the software business were relatively low. But the middle 1980s, perhaps as early as 1983 starting a software company require far more financial resources to compete with the already established companies, from Microsoft to Lotus to Software Arts (VisiCalc) to gaming software manufacturers. These new companies, all formed between 1978 and 1982, produced the software that made microcomputers, of which there were dozen of different platforms or operating systems, useful. Walker saw this explosion of software producers as models and incentives for his own venture into microcomputer software.

Walker laid out four different types of software MSP could profit from: guerilla programs, or subroutines for existing software produced by other companies; new closed applications for microcomputers; software tools, such as word processors or sorting programs; and emulators to allow software and programming languages to cross platforms.[22] Walker's company had to write programs for multiple systems to ensure that, as the microcomputer market developed Walker's programs were not tied to platforms that ceased to exist. Walker wrote, "we simply can't afford to bet on one system to the exclusion of others."[23] Walker urged his partners to use high-level languages like C and Pascal that were available on many microcomputers. MSP would develop as many different software packages for as many different platforms as possible, and then see what sold. Walker aimed to make MSP one of the top 5 names in microcomputer software. He enlisted 15 programmer/investors in his initial corporation, and on April 26, 1982 incorporated his company, renamed Autodesk, in the state of California.[24]

---

[20] John Walker, "Working Paper--Background," January 12, 1982, http://www.fourmilab.com/autofile/www/section2_2_2.html#SECTION0022000000000000000

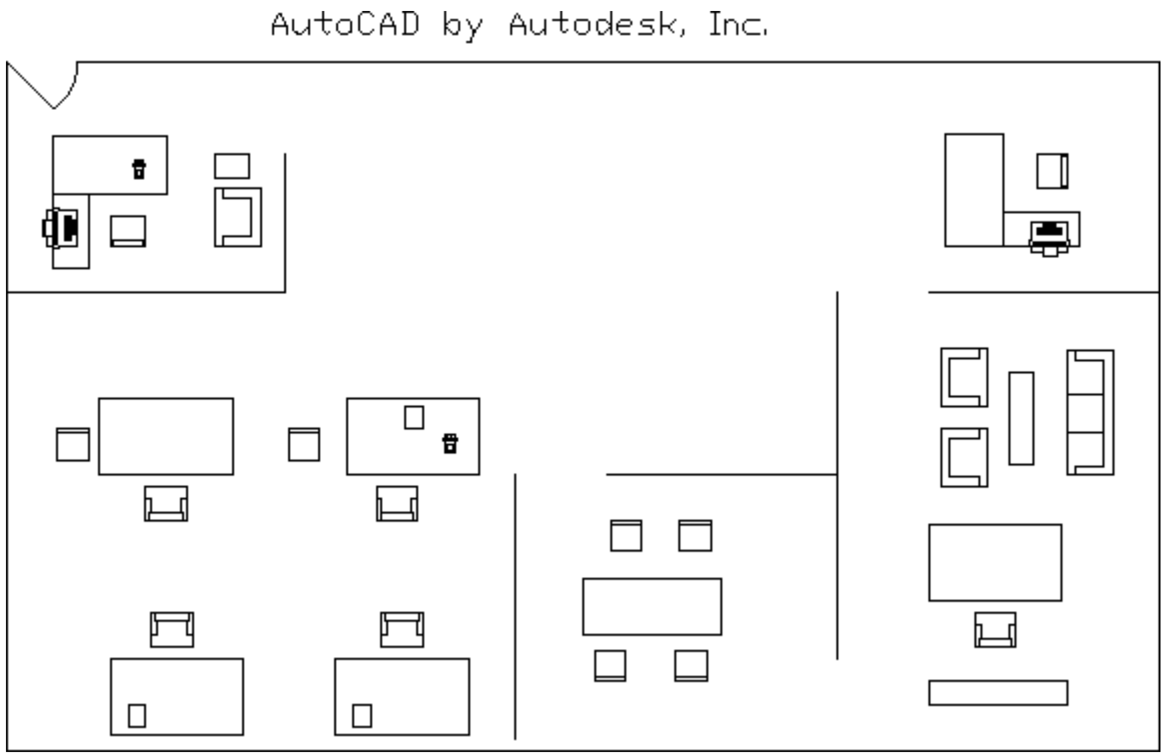
[21] Martin Campbell-Kelly and William Aspray, *Computer: A History of the Information Machine* (NY: Harper Collins, 1997) 259-261.

[22] John Walker, "Working Paper—The Nature of Potential Products," January 12, 1982. http://www.fourmilab.com/autofile/www/section2_2_10.html#SECTION0021000000000000000

[23] John Walker, "Working Paper—Hardware and System Strategy," January 12, 1982. http://www.fourmilab.com/autofile/www/section2_2_11.html

[24] John Walker, "Initial Stock Distribution, no date. http://www.fourmilab.com/autofile/www/chapter2_8.html#SECTION0080000000000000000

One of the about 6 products that Walker focused on developing was a drawing program for the TI 9900. The program, initially called Interact, had been written by Mike Riddle, one Walker convinced to be one the original partners in Autodesk. Walker & Riddle wanted to extend Interact to a number of different platforms. Walker & Riddle planned to produce a microcomputer drawing system—hardware and software—for about $10,000 to 15,000—yet they planned for it to be competitive with minicomputer CAD systems selling for $70,000.[25] From the beginning Walker's plan was to introduce his company at the COMDEX computer trade show in Las Vegas in November of 1982 by demonstrating Interact, which he first renamed Micro CAD then subsequently AutoCAD, and a data sorting program called Autodesk (which is where the company's name came from). While Walker never imagined, at least before 1983, that Interact was Autodesk's golden goose, he reacted immediately to the demand for what Autodesk's advertisements called a "word processor for drawings."



The demo drawing used to show off AutoCAD's capabilities at the COMDEX 82 Computer show[26]

The response to the COMDEX display was phenomenal. Dan Drake, Autodesk's Vice-President, wrote the following about the COMDEX experience, "It was apparent that AutoCAD was a hot product. At the biggest show in the industry, it had the field to itself, with no direct competition at all. The main problem before the meeting was how to cope

[25] John Walker, "Autodesk Products: Micro CAD" April 28, 1982. http://www.fourmilab.com/autofile/www/subsection2_9_12_1.html#SECTION00912100000000000000

[26] From Dan Drake, "Information Letter 4 – Technical Plans," April 2, 1982

with success."[27] By the summer of 1983, Walker had shifted the focus of the company away from all other products to constantly improving and supporting AutoCAD. Considering Walker's original conception of an ad-hoc company to produce numerous programs, the shift to a company dedicated to the production of a single package—AutoCAD—constituted a major shift in the company. As early as June 1983, Walker claimed, "The original company is dead. We are now a classic small company with one product rather than the imagined company developing many products."[28] With this total emphasis on drafting software, Autodesk also faced a problem—it was a company of programmers, not of draftspeople. Particularly since Autodesk employees needed to demonstrate AutoCAD to potential clients, it became apparent to Walker and the company that they were not particularly aware of how engineers and others actually used CAD software. Consequently, the disjunction between CAD software and drawing practice was obvious within a year of the introduction of AutoCAD software. When a company requested help in setting up a class for 200 draftspeople in AutoCAD, Autodesk was at a loss for how to do this and called on the expertise of its clients—the experts in actually using the program.[29] In 1983, Drake wrote, "We seem to need an AutoCAD Jockey. . .We should find people who are especially good with AutoCAD and make them specialists in making the product look easy and impressive. . . Even if they can't go around giving demonstrations, they can create drawings and set up demos for other people. This was agreed to be a good idea, but so far we lack anyone to take on the job."[30]

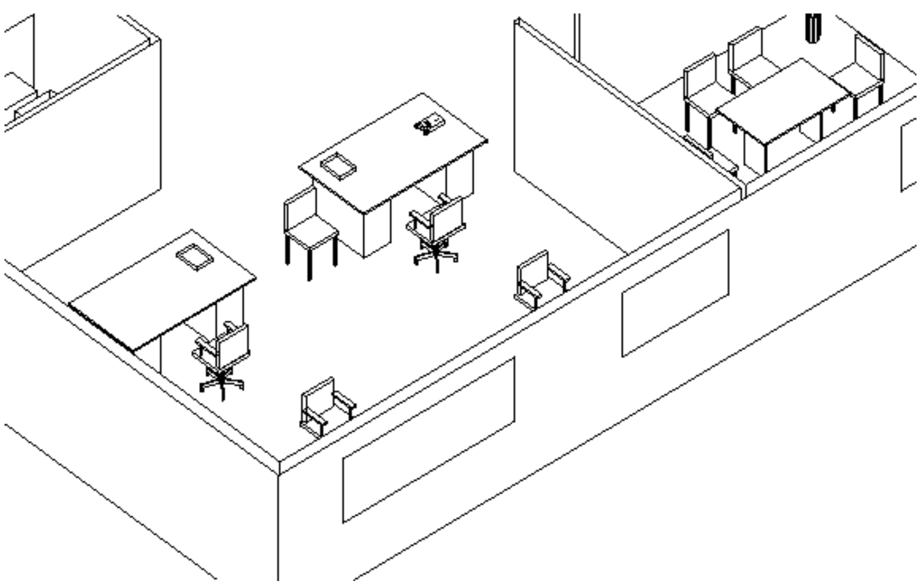
Walker repeatedly called on the users of AutoCAD to fill this lacuna. Walker created numerous channels for users alert the company to problems with the software, to request new features, and to contact one another with suggestions for how to use the program. Walker wanted to create a community of users, who would support the program and help improve, as well as diffuse it. Neither Walker nor his programmers knew enough about CAD practices to do this themselves—as a result a unique relationship was set up between Autodesk and AutoCAD users—a relationship which more than anything else marks the way Autodesk continues to do business to this day and which hardened Autodesk's commitment to an open, user-modifiable architecture. In addition for the first time in the summer of 1983 Walker hired a graduate student in architecture to come to Autodesk for the summer, use the program, advise the programmers, and produce good sample drawings for advertisements. Malcolm McCullough was the first, and as far as I've found the only, instance of Autodesk hiring anyone skilled in drafting.[31] AutoCAD did enlist its users in a collective effort to both improve the program but also create a collaborative community of users to share insights into using AutoCAD. To do so, Autodesk paid for the production of a bi-monthly users' group newsletter called *The CADalyst*.

However, Walker wasn't satisfied with simply responding to the requests of users. He also planned new features for the software aimed at convincing managers to buy the software and decide that their design teams would benefit from using it. To establish

---

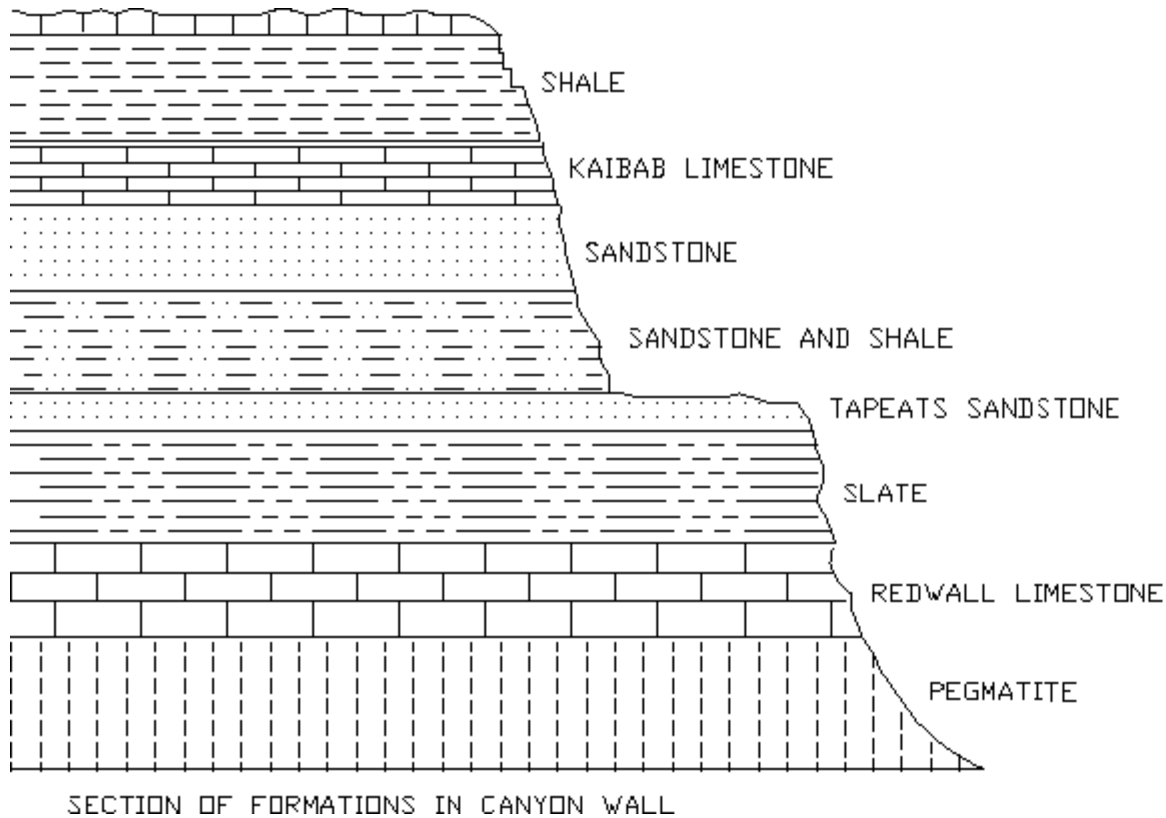
[27] Dan Drake, "Meetings: December 1982-January 1983," no date
http://www.fourmilab.com/autofile/www/chapter2_17.html#SECTION00170000000000000000
[28] Kern Sibbald, "June 1983 Meeting," June 7, 1983.
http://www.fourmilab.com/autofile/www/chapter2_20.html#SECTION00200018000000000000
[29] Kern Sibbald, "June 1983 Meeting."
[30] Dan Drake, "February 1983 Meeting: Trade Shows," no date.
http://www.fourmilab.com/autofile/www/section2_18_4.html#SECTION00184000000000000000
[31] John Walker, "Electric Malcolm," September 14, 1983.
http://www.fourmilab.com/autofile/www/chapter2_27.html#1837

AutoCAD as a market leader with any potential for longevity, he also needed to expand the number of users. In the summer of 1983 he wrote a memo asking the AutoCAD programmers to consider how to add-on a three dimensional capability. He did this not to fulfill users' needs—no one was asking for three-dimensional capability since what he was suggesting wasn't really an existing aspect of mechanical drawing practice, which commonly communicated three-dimensional object using multiple orthogonal views.[32] Walker claimed that a demo of a rotating cube in space was more impressive than zooming in on a two-dimensional drawing adding, "3D is important more from a marketing perception standpoint than a technical one."[33] But Walker did not want to rewrite the program; he wanted the 3D features to be added on. This meant that they were not true 3D solids, but rather "wire frame" extrusions—where the appearance of 3D objects would be created out of planar shapes. For most AutoCAD users this was a useless feature. But Walker insisted that the slogan "AutoCAD, now with 3D" was a better advertising hook than the more mundane but more useful "AutoCAD, now with cross-hatching and dotted lines."[34] Walker considered these two features equivalent programming tasks. Still the 3D add-on could be sold for an additional $1,200, while the dotted lines were simply added into the existing $2,500 package. Developments like this drove AutoCAD programming, and therefore its use, in particular ways that were occasionally unrelated to current mechanical drawing practice. In hindsight, Walker saw the rush to introduce three-dimensionality as a mistake, driven by marketing concerns. However, Walker's basic intuition was correct—CAD did not need to follow drawing conventions rigidly, because it would create new ones.



AutoCAD drawing from 1983, showing the extrusion technique for representing 3D.[35]

[32] Three-dimensional representations typically distort the size and/or shape of objects, as in perspective drawing or isometric projections—something inherently undesirable for mechanical drawing.
[33] John Walker, Low Rent 3D" September 5, 1983. http://www.fourmilab.com/autofile/www/chapter2_26.html#SECTION00260000000000000000
[34] Walker, "Low Rent 3D."
[35] From Walker, "Low Rent 3D."



Dotted Lines Demonstration Picture 1983[36] (note the machine-like lettering)

The years 1983-86 were growth years for AutoCAD, capped by their triumphal initial public offering, which valued the 4-year old company at $70 million.[37] But in these first three years Autodesk's successful marketing of AutoCAD should be contrasted with the realities of using the program. While Walker's description of AutoCAD as a word processor for drawing brings to mind the conversion from typewriter to word processor, one must remember that with drawing the transition is much more radical, given that no typewriter analogy exists for mechanical drawing. So moving from hand drawing to CAD was like asking a medieval scribe, who had never seen a typewriter, to use a word processor—the technology was foreign and the manual and mental skills needed to use it were utterly unconnected with the former techniques used to produce drawings. In addition, today CAD drawings are produced using a mouse, digital pen or other manual input device, at least requiring the movement of the hand. In the early 1980s, computer drawings were input through a keyboard as data—without any common ground with the physical motion of drawing. In the manual for the 1986 program, a short brochure was included to inform users "how to get started in AutoCAD without reading the manual." AutoCAD instructions were typed into a command line. So to draw a polyline (or line with multiple segments), one typed the following into the command line (italic text was entered by the user):

[36] From Dan Drake, "The Deal on the Table," no date.
http://www.fourmilab.com/autofile/www/chapter2_32.html#SECTION00320000000000000000
[37] John Walker, "Initial Public Offering" April 16, 1985.
http://www.fourmilab.com/autofile/www/chapter2_38.html#SECTION00380000000000000000

Command: *pline*
From point: *2,2*
Current line width: *0.0000*
Arc/Close/Halfwidth/Length/Undo/Width/<endpoint of line>: *@6<90*
Arc/Close/Halfwidth/Length/Undo/Width/<endpoint of line>: *@6,0*
Arc/Close/Halfwidth/Length/Undo/Width/<endpoint of line>: *8,2*
Arc/Close/Halfwidth/Length/Undo/Width/<endpoint of line>: *close*

Outside of the obvious fact that no one could really use AutoCAD without reading at least some of the manual, it is also obvious that the act of drawing on the computer in 1986 had more to do with computer programming than with drawing.[38] This posed a real challenge to integrating the program into a drafting environment—"drawing" on the compute required a completely different skill-set, perhaps even a different mindset and personality than did traditional mechanical drawing. While there were certainly some converts from traditional drafting to CAD, the chasm between the two skills most certainly claimed some victims. As AutoCAD and other drafting program caught the eye of managers, many traditional drafters found their jobs anywhere from radically transformed to axed. This phenomenon, certainly true in its own right, leads to the "deskilling" argument in both Henderson and Ferguson, which accurately describes what is lost with CAD without any claim about what is gained. However, Walker and Autodesk never emphasized how radical a change this was, showing both their ignorance of drawing skills and the fact that they were looking to the post-transition future. As early as 1985, when "drawing" in AutoCAD clearly resembled programming, Walker wrote, "more than 1,000 dealers, software houses, and OEMs have discovered that computer-aided design isn't an esoteric product for the Fortune 500, but an everyday tool as fundamental to people who draw as a word processor for people who write."[39]

1986 brought a new set of programming challenges which would move the process of drawing in AutoCAD considerably closer to actual drawing. Responding to the 1984 release of the Macintosh, Autodesk and others immediately saw promise in the graphical user interface, or GUI, the what-you-see-is-what-you-get or WYSIWYG display, and the mouse as a standard data input device. These three developments, somewhat painfully adopted and accommodated by AutoCAD over the 1980s, helped overcome one major objection that many firms had to computer-aided drafting—its ugliness. The drawings above show the rough, machine-drawn character of computer drawing in the 1980s. For firms making their living designing—from architecture to interior design to industrial design--ugly drawings were more than an superficial problem, they undercut the qualities of good design successful designers were trying to sell to their clients. To grab this market AutoCAD drawing needed to look less computer-produced. Design drawings, even of strictly functional parts like gears, still needed to look attractive, and traditional drafters infused their drawings with style, from plate layout to line quality to, most obviously, lettering. When AutoCAD required displayed green lines on a black screen there was no visual analogy between hand-drawn plates and screen appearance. Both screen displays and plotter outputs remained unable to draw smooth curves, essential for drawing, but a particular problem for

---

[38] "How to get started in AutoCAD without reading the manual" AutoCAD publication TD113-001, July 28, 1986. Available at http://betaprograms.autodesk.com/history/area51.htm
[39] John Walker, "Number One" March 1985. http://www.fourmilab.com/autofile/www/chapter2_36.html#SECTION00360000000000000000

lettering (see the lettering on the "geological" plate above—designed as a demo for AutoCAD's line qualities).

These problems were being addressed as the personal computer moved to a white screen and a windows environment, both changes introduced to a wide market by the Apple Macintosh, though with clear predecessors such as the Xerox Star.[40] For AutoCAD this posed serious problems. AutoCAD had always been command driven, giving the user a great deal of control over how she manipulated the program. The event driven menus and dialogue boxes of the Macintosh would take away some of the user's control. Walker wanted AutoCAD to be the first open-architecture, event-driven program. Using Auto Lisp, a lisp programming utility that had been added to AutoCAD in 1985-6, the user could essentially reprogram the menus and dialog boxes. AutoCAD aimed to be both user-friendly for the novice and customizable for the experienced designer. In software design, as in some other products, power and usability can run at cross-purposes, as the program becomes more powerful it also becomes more difficult to use, requiring more programming-like skills for routine use. Autodesk claimed to be trying to walk a fine line, between making the program customizable by LISP-programmers but instantly useable by the LISP-illiterate. Still, convincing users of the program's ease of use was often a tough sell. AutoCAD's advertisements frequently discussed the program's ease of use, a position which would be unnecessary unless the company was trying to overcome a reputation of being difficult. Autodesk also fund that it needed to not only support user groups, but also to sponsor training in use of the program, something the company continues to the present day. Walker has also admitted some of the programs' notoriously difficult features, such as "I believe that no commercial software product with comparable installed base and revenues has a user interface remotely as bad as AutoCAD's."[41]

In light of the changes of the late 1980s, difficult as they were for Autodesk to implement, AutoCAD was on its way to becoming a drawing program, instead of a way for programmers to draw. With AutoCAD Release 9 in 1987 (but not before), AutoCAD was poised to change general design practice. Release 9 introduced true three-dimensional solids, instead of an add-on extrusion feature. Release 10, in 1988-9, could generalize an operation to any plane in space, making the screen display truly three-dimensional. Using this feature an object could be viewed from any angle, and changed from while in that perspective. As the display grew more sophisticated, (i.e., it began to look more like an actual drawing), the ability to work on a two-dimensional picture from an infinite number of perspectives in three dimensions was an operation not possible in the same way on traditional drafting. Now two users could sit at a terminal and see the same object from the same perspective, rather than trusting their minds' eyes and intuition that they were in fact imagining the same perspective. Visual communication was on the screen; it was made visual, not the conversion of visual in the mind's eye into verbal communication and reconverted into another mind's eye.

Not coincidentally, the 100,000th copy of AutoCAD was sold in 1987, amounting to a total of $200 million in sales over the first 5 years, with another $100 million to follow in 1988 alone.[42] By 2000, AutoCAD sales topped $100 million in Europe alone. AutoCAD is

---

[40] See Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog*, 246-263.
[41] See John Walker, "Technological Leadership: User Interfaces" August 23, 1988. http://www.fourmilab.com/autofile/www/subsectionstar2_68_0_7.html#SECTION00680700000000000000
[42] John Walker, "Statement for the $100 million Meeting," December 13, 1988. http://www.fourmilab.com/autofile/www/chapter2_72.html#SECTION00720000000000000000

clearly established as the leading computer-aided drafting program. While AutoCAD's development has had its ups and downs since the late 1980s, it has remained the market leader, due it part to its own inertia, but also due to the company's diligence in continually upgrading and adding features to the program.

[Insert photo-realistic, current-generation AutoCAD drawing here]

III. CAD's effect on Visual Creativity: Seeing through the Machine

The history of AutoCAD's early development is important in assessing AutoCAD's contribution to the changes in engineering visual practices, because AutoCAD was not produced to duplicate mechanical drawing operations. In one sense, then, AutoCAD acted as an exogenous influence on drawing and visualization practices. This was a destabilizing impact, resulting in the social shifts Kathryn Henderson describes in *On Line and On Paper*. Destabilization came as a result of the AutoCAD programmers' limited understanding of mechanical drawing practice. They were building a new process model of how to produce mechanical drawings, not trying to computerize existing drafting practice, about which they knew woefully little. Clearly, they succeeded in generating a new socio-epistemological process of making drawings, but this is the new paradigm which has had a significantly destabilizing effect. While this kind of destabilization was not the goal of AutoCAD, avoiding it was not AutoCAD's programmers' concern, either.

In addition, Walker's marketing plans--which sold AutoCAD's 3D images to managers and not its practices to drafters--was a tool for diffusing the software. Walker's strategy also exacerbated its destabilizing effects, since AutoCAD was introduced into the design firm by fiat. This fiat changed the hierarchy of the design process not only by adding a new category of computer operators, but also by compromising the expertise of the people who could integrate computer practices into traditional mechanical drawing practices.[43] To a manager, AutoCAD promised greater productivity, because it would allow unlimited changes to drawing files without redrawing whole plates. Managers, who could make decisions about the use of software, wanted to hear this—the analogy of the word processor for drawings was attractive to them. As a result, from the empowered position of the manager, AutoCAD offered obvious advantages over traditional drafting, even if it set up considerable obstacles to users. Paradigm shifts typically contain this dynamic of benefiting one concern at the expense of other concerns. Clearly the power to make this decision lay in the hands of managers, and drafters in large firms were unable to answer managers' concerns. It can certainly be argued that in the 1980s that AutoCAD did not actually increase productivity, since the learning curve of the program was so steep and its capabilities relatively low. But Walker's strategy did begin the turn to computer-aided drafting, and the capabilities the program has developed over the past 22 years were facilitated by the demands and insights of the program's users.

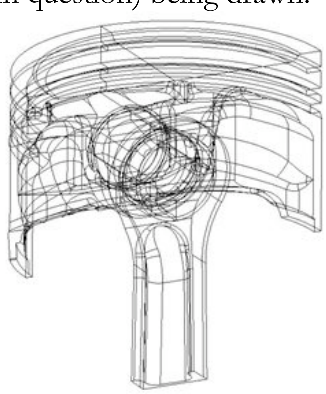
AutoCAD users were developing new practices from their initial encounters with the program. The first and most obvious change in how drawings are produced with the computer versus without is the ease of modifying elements of the drawing without completely redrawing plates. Since Walker repeatedly touted this aspect of AutoCAD in his claim of making a word-processing for drawings, and we are so familiar with the dynamics of word processors, understanding this shift is easy. As is true in writing, the constant modification to a digital file changes the basic nature of a "draft." Unless one consciously

[43] Henderson, 142.

chooses to save changes as a separate file, old drafts cease to exist unless printed, text being digitally subsumed into "the current" and often only version. This is as true for drawings as for text. But the meaning of rough draft can be rather different when one moves from text to drawing. While word processors can capture fragments, outlines, lists, and verbal sketches as well as formal and polished text, this analogy does not hold for computer-aided drafting. Computer-aided drafting does not lend itself to rough drafts, or more properly sketches, since drawing on the computer requires precise definitions of space, which is what sketching often lacks.[44] As a result, despite several attempts to produce sketching software, sketching by hand persists and has been fused into computer-aided design practice.

But the ease of change has also led to new conceptual practices; for example, the generation of large numbers of options. Capitalizing on the ease of producing numerous versions of a design, computer-aided drafting can significantly increase the number of variations considered, a phenomenon likely to generate greater creativity. Artificial intelligence pioneer Marvin Minsky, invoking science fiction writer Douglas Hofstadter, claimed that generating variations on a theme was the crux of creativity.[45] Programs like AutoCAD facilitate this behavior by lowering the effort needed to investigate options visually. Whereas Henderson criticizes computer-aided design for streamlining the design process and therefore reducing creative exploration, a number of engineers and architects have claimed that it has increased conceptual exploration. For example, Peter Eisenmann has used distinctly computer-generated forms to explore new architectural forms.[46] Peter Rice, the esteemed structural engineer responsible for unconventional structures from the Sydney Opera House to the Schlumberger tent, claimed that the computer allowed him to explore novel geometrical arrangements making possible some of this most creative work in membrane structures and fabric roofs.[47]

In addition to its facilitation of variation, computer-aided drafting also changes the conceptual process of producing drawings in a variety of ways. Producing drawings on a computer allows the drafter to view the artifact from an infinite number of perspectives. I will call this phenomenon *seeing through the machine*, which has an intended double meaning. The computer drafter does, in fact, see the artifact being designed "in" the computer screen, which is, of course, itself a machine. But by being able to move through virtual space and move to any perspective, the drafter can also see through the machine (or whatever artifact in question) being drawn.



[44] Casper G. C. van Dijk, "New Insights in Computer-aided Conceptual Design" *Design Studies* **16** (1995) 65.
[45] Yu Tung Ling, "Creativity or Novelty" *Design Studies* **21** (2000), 272.
[46] R. Hanna and T. Barber, "An Inquiry into Computers in Design: Attitudes before—Attitudes after" *Design Studies* **22** (2001), 258.
[47] Peter Rice, *An Engineer Imagines* (London: Artemis, 1994), 95-6.

Seeing the object of design from new perspectives means that the computer screen has become a new space for design, one which does not have to conform to physical realities such as gravity. This alternate, dynamic reality provides a new creative space for design, since conceptually removing real-world constraints can often spur new trajectories in thinking about an artifact. Students learning CAD often claim that this new photo-realistic three dimensional space is one of the major advantages of using CAD systems.[48] At the same time it also proves to be one of the more difficult aspects for traditional drafters to assimilate mentally. Whereas one used to drawing in three orthogonal views is conditioned to determine what view a particular feature is seen from, moving effortlessly through computer-screen space does not require specific questions about viewpoint. In fact, the photo-realism of seeing on the screen often means one completely loses track of the "tricks" needed to show three dimensional space in a two dimensional medium.

I discovered this myself in teaching finite element analysis to students who had learned how to draft on a computer—their facility in three-dimensional screenspace was completely different from my own. I found myself constantly questioning my students about their vantage point as they moved through their models on the computer screen, asking repeatedly, "now, where are you relative to the object?". Neither my three-views orientation nor their "video-game" way of seeing was more creative, but an obvious disconnect existed between these two ways of seeing. Seeing the object of design in different ways led to asking different questions about the object of design, and ultimately to different designs. New visual interactions are clearly facilitated by this new space. For example, screenspace makes it much easier to perceive and understand overlapping and intersecting three-dimensional forms. These forms have been exploited in computer-aided design projects, shown in particular by the work of Peter Rice.[49]

IV. Conclusion: Screenspace and the Mind's Eye

According to Terry Liddament, drawing is not an act of transcribing what one sees in one's mind's eye.[50] Liddament has written that seeing in one's mind's eye is not an imaginative act, but rather something more akin to drawing in one's head. The tools one has for expressing imagery are a part of understanding the world visually. If one takes this argument seriously, then seeing in the mind's eye cannot be prior to developing the drawing skills--whether traditional or computer-aided--needed to understand visually. Therefore, if computer-aided drawing skills are different from traditional drawing skills, they will then change the way one visualizes before or without drawing. Liddament's argument suggests that different drawing tools generate different ways of seeing and therefore understanding. While this seems radical, examining the development and adoption of CAD tools is a way to test this process.

Whatever position one takes on Liddament's argument about perception, seeing on the computer screen does affect the social dynamics of visual creativity. Even in teaching students with a much greater facility in screenspace, the ability to show another person definitively what you are seeing is a significant change from a kind of collaborative imagining. Prior to the technical breakthrough of having two engineers looking at the same artifact from the same perspective on a computer screen, conversations, buttressed with "cocktail napkin" drawings, but more importantly parallel images in the collaborators' mind's

[48] Hanna and Barber, 273.
[49] Rice, 103.
[50] Terry Liddament, "The Myths of Imagery" *Design Studies* **21** (2000), 589-606.

eyes, were a staple of design collaboration. There was also tremendous room for error in this social situation—one never really knew if the other person was imaging the same design from the same perspective. Conversations usually ferreted out the most obvious miscommunications, but the insufficiency of verbal discussions of visualizations was always problematic. Screenspace acts a corrective, eliminating the need to discuss images in verbal terms. This shared visual experience changes the social dynamics of creativity, allowing more time to be devoted to variation and less to communication. This obviously has both advantages and drawbacks. It does lead to what Walker promoted and Henderson and Ferguson disparage as streamlining. For Henderson and Ferguson the loss of conversation is a detriment, since conversation, even when indirect, fuzzy, and confused, can lead to better design. For Walker and CAD users today eliminating a major channel for visual miscommunication allows more time to be spent on design progress. Both attitudes seem naïve, and in fact show the shift to computer-aided design is paradigmatic, with its attendant conceptual losses and gains.

A more nuanced analysis of the new social space of computer-aided drawing comes from Louis Bucciarelli.[51] Bucciarelli argues that the CAD drawing, like its traditionally-draw predecessor, acts as a mediator between object worlds. But it still must be interpreted within object world mindsets. In other words, the same token drawing has different meanings to different users, because they are trapped within a particular object world perspective. But computer-aided drawings complicate this a bit because of the social dynamics of collectively producing the drawing in the first place. Because people from different object worlds can collectively produce a single drawing by passing around its digital file, the screen image is therefore seen from multiple object-world perspectives. But changes made in one object world are carried into the next modifier of the drawing file. Therefore, the plotted drawing the represents multiple object world views, acting as a conceptual mediator. The CAD drawing is an artifact just like the object of design, but it sole purpose is to act as an instrument to collect and negotiate information between object worlds. Object world blindness persists in the age of computer-aided design, but the nature of the drawing as an artifact changes.

CAD users today are neither more nor less creative than designers who used traditional drafting methods. They are neither more nor less productive. They are neither more nor less error-prone. But they clearly see and think differently than their pre-computer-aided selves. Most design students who learn CAD feel that it enhances their design cognition.[52] But there is also a significant argument for losses at the hands of the new paradigm of computer-aided mechanical drawing, particularly in the earliest stages of a design, usually considered the least defined and therefore most creative moment in design. Investigating the changes in visual thinking attributable to a new tool and a new paradigm allow profound questions to be asked about the nature of engineering creativity. When creativity emerges from a new dynamic the creative dimension of engineered products changes. This chain of events returns us back to the Csikszentmihalyi socio-cultural model of creativity, situating creativity not in the mind of a single genius, but rather in a method of intellectual production. Since nearly all ways of understanding engineering design emphasize its social and collective nature, engineering creativity must also be based in that same collective process. Computer-aided design works in that collective process, but also disrupts it, creating, over time, a new paradigm of visual thinking.

---

[51] Louis L. Bucciarelli, "Between Thought and Object in Engineering Design" *Design Studies* **23** (2002), 225.
[52] Hanna and Barber, 273.