

# Drawing Inferences:

Building Geometric Models With Hand-drawn Sketches

By

Ewan E. Branda  
Bachelor of Architecture  
University of Waterloo  
Waterloo, Ontario, Canada  
October 1989

Submitted to the Department of Architecture in partial fulfillment of the requirement for the degree of Master of Science in Architecture Studies at the Massachusetts Institute of Technology, May 1998.

© 1998 Ewan Branda. All rights reserved

The author hereby grants to M.I.T. permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

Signature of the Author: \_\_\_\_\_  
Department of Architecture  
May 15, 1998

Certified by: \_\_\_\_\_  
William J. Mitchell  
Professor of Architecture and Media Arts and Sciences  
Thesis Supervisor

Accepted by: \_\_\_\_\_  
Roy Strickland  
Associate Professor of Architecture  
Chairman, Department Committee on Graduate Students

**Readers:**

Daniel Tsai

Research Associate

Graduate School of Design, Harvard University

William L. Porter

Professor of Architecture and Planning

Massachusetts Institute of Technology

**Thanks:**

At the GSD: Daniel Tsai. At the MIT Artificial Intelligence Laboratory: Patrick Winston, Kimberle Koile, Thomas Rickert, Pawan Sinha. In the MIT Computer Graphics Group: Jon Levene, Mok Oh. In the MIT Department of Architecture: Bill Mitchell, George Stiny.

## Drawing Inferences: Building Geometric Models with Hand-drawn Sketches

by Ewan E. Branda

Submitted to the Department of Architecture in partial fulfillment of the requirement for the degree of Master of Science in Architecture Studies at the Massachusetts Institute of Technology, May 1998.

### ABSTRACT

Architects work on drawings and models, not buildings. Today, in many architectural practices, drawings and models are produced in digital format using Computer-aided Design (CAD) tools. Unquestionably, digital media have changed the way in which many architects perform their day to day activities. But these changes have been limited to the more prosaic aspects of practice. To be sure, CAD systems have made the daily operations of many design offices more efficient; nevertheless, they have been of little use - and indeed are often a hindrance - in situations where the task at hand is more conjectural and speculative in nature, as it is during the early stages of a project. Well-intentioned efforts to insinuate CAD into these aspects of practice have only served to reveal the incongruities between the demands of designer and the configuration of the available tools.

One of the chief attributes of design practice is that it is action performed at a distance through the agency of representations. This fundamental trait implies that we have to understand how computers help architects describe buildings if we are to understand how they might help architects design buildings. As obvious as this claim might seem, CAD programs can be almost universally characterized by a tacit denigration of visual representation. In this thesis, I examine properties of design drawings that make them useful to architects. I go on to describe a computer program that I have written that allows a designer to build geometric models using freehand sketches. This program illustrates that it is possible to design a software tool in a way that profits from, rather than negates, the power of visual representations.

Thesis Supervisor: William J. Mitchell,  
Professor of Architecture and Media Arts and Sciences

## Drawing Inferences: Building Geometric Models with Hand-drawn Sketches

### ABSTRACT - 2

### DIGITAL DESIGN MEDIA AND THE VIRTUE OF REPRESENTATIONS - 7

You Should Be Able To Work With A Computer On Your Terms  
A Proposal For Building Geometric Models With Free-Hand Sketches

### WHY REPRESENTATIONS ARE IMPORTANT IN ARCHITECTURAL DESIGN - 9

Representations Lose Information  
Representations Facilitate Movement Through Diverse Modes Of Perception  
The Manufacture Of Representations Employs Perceptual Faculties

### SOME PROBLEMS WITH TRADITIONAL CAD SYSTEMS - 25

The Architecture Of Digital Design Media  
Cad Systems Propose A Direct Communion With Geometry  
Cad Systems Mimick The Production Of Real Objects  
Conventional I/O Methods Insulate Imagination From Perception

### THE NATURE OF ARCHITECTURAL DESIGN ACTIVITY SHOULD INFORM A PROPOSAL FOR A SOFTWARE TOOL - 30

Usually, Design Proceeds By Making Specific Proposals  
A Designer Moves 2 Steps Forward, 1 Step Backward

### A USEFUL DIGITAL DESIGN TOOL SHOULD EXPLOIT THE ENDOWMENT OF VISUAL REPRESENTATIONS - 32

It Should Respond With Its Own Interpretations  
It Should Allow For Easy Movement Between Conjecture And Constraint  
It Should Allow For Easy Transactions Between Methods Of Picturing And Methods Of Measurement  
It Should Cause An Engagement Of Visual And Motor Faculties

### BUILDING GEOMETRIC MODELS WITH SKETCHES: A Software Tool For Early Stages of Design - 33

Freehand Sketching Can Be A Way Of Creating Models  
Sketch Interpretation Involves The Possibility Of Misinterpretation  
Sketching Allows For Rapid Testing And Editing Of Ideas

**USING THE PROGRAM - 35**

**OVERVIEW OF THE PROGRAM DESIGN - 36**

Sketch Input

Image Processing And Vectorization

Scene Segmentation And Structural Relations

Regularizing The Scene

Three-Dimensional Scene Recovery

Displaying And Re-Drawing

**ASSESSMENT AND EXTENSIONS - 45**

**LIMITATIONS OF THE CHOSEN MODEL**

**POSSIBLE EXTENSIONS OF THE WORK**

Study Types Of Projection

Expand The Notational System

Keep A Record Of Sketches And Models

Support Multiple-Users

Make The Program Learn The User's Intentions And Gestures

Introduce Other Images

Build In External Constraints

Provide Better Drawing Methods

**REFERENCES - 49**



## Part 1

### DIGITAL DESIGN MEDIA AND THE VIRTUE OF REPRESENTATIONS

Architects work on drawings and models, not buildings. Today, in many architectural practices, drawings and models are produced in digital format using Computer-aided Design (CAD) tools. Unquestionably, digital media have changed the way in which many architects perform their day to day activities. But these changes have been limited to the more prosaic aspects of practice. To be sure, CAD systems have made the daily operations of many design offices more efficient; nevertheless, they have been of little use - and indeed are often a hindrance - in situations where the task at hand is more conjectural and speculative in nature, as it is during the early stages of a project. Well-intentioned efforts to insinuate CAD into these aspects of practice have only served to reveal the incongruities between the demands of designer and the configuration of the available tools.

One of the chief attributes of design practice is that it is action performed at a distance through the agency of representations. This fundamental trait implies that we have to understand how computers help architects describe buildings if we are to understand how they might help architects design buildings. As obvious as this claim might seem, CAD programs can be almost universally characterized by a tacit denigration of visual representation. This assertion may seem surprising given that digital media provide a facile means of producing and distributing visual material. But quantity and efficiency are not the sole criteria for useful representations: design drawings have certain properties without which they would be useless for conceiving and elaborating projects. I will discuss these properties and show that if digital media are to contribute to design practice in any profound way they will have to be reconsidered as extensions to this subtle and complex descriptive system. Though architects use a wide range of types of visual representations in practice, I will primarily discuss drawings.

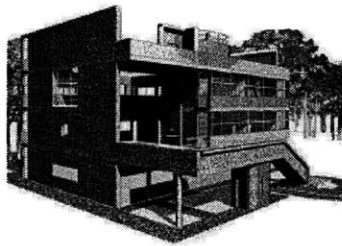
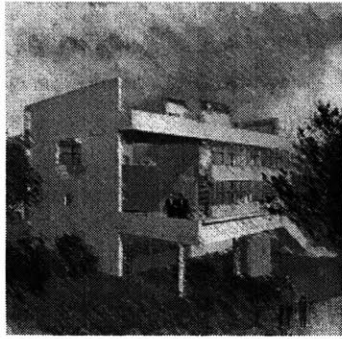
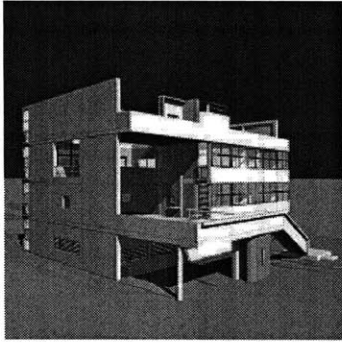


Figure 1  
Apologetic rendering. The simple rendering of the original geomtric model has been post-processed using the *Piranesi* rendering system.

Top: the original flat-shaded rendering.  
Middle: "The same scene, re-rendered in an impressionistic style."  
Bottom: "The same scene, re-rendered in the style of an engraving."  
Richens (1997).

## YOU SHOULD BE ABLE TO WORK WITH A COMPUTER ON YOUR TERMS

The descriptive tools that designers develop and refine for themselves comprise a notational system that allows for fluid and concise expression of ideas. But exploratory and probationary schemes are crushed in the rough grip of CAD. It is not for want of subtlety in graphic expression that CAD systems undermine the power of drawings as design tools. This misconception has resulted in a proliferation of software that attempts to recapture the "look and feel" of freehand drawing by clothing the obscene geometric database in a fancy-dress of squiggly linework and luxurious textures (figure 1). CAD falls short of freehand drawing because it obliges the designer to work *according to its terms*. It forces conformity to its own strict procedure for making visual inscriptions and negates almost all of the properties of design drawings that make them inseparable from the production of designed objects.

## A PROPOSAL FOR BUILDING GEOMETRIC MODELS WITH FREEHAND SKETCHES

Digital geometric modeling programs are an instructive form of architectural representation. They are not really a form of "virtual reality" - as their designers are wont to claim - as much as they are a limit-case scenario illustrating one of the key problems of graphic representation: how can three-dimensional space be represented on a two-dimensional surface? Allowing the user to effortlessly interact with a database representing three-dimensional geometry entirely using a two-dimensional medium (the computer monitor) is not a trivial problem. In fact, what is a problem in the design of modeling software is the same thing that makes visual representations important in design: in the act of making a representation of an existing or yet unrealized object the designer transports the object between spatial dimensions. In making a sketch of an imagined space, the transformation is from three dimensions to two and in making a model from a sketch it is from two dimensions to three. This transformational capacity that is an attribute of all visual representations is what makes them crucial to the practice of design. In this thesis, I describe the architecture of standard geometric modeling packages and argue that this particular configuration is incongruous with the requirements of a design



tool because it suppresses these important properties of visual representations. I also describe a computer program that I have written that allows a designer to build geometric models using freehand sketches. The program is designed to exploit the transformational qualities of visual representations. It illustrates the properties that a program might have if it is to extend the architect's collection of design methods and tools in any significant way.

## WHY REPRESENTATIONS ARE IMPORTANT IN ARCHITECTURAL DESIGN

The apparent ability of a representation to transform what it transmits is the source of its potency. In the act of describing something - whether real or imagined - we transform it: as an idea moves between states - between mental image, spoken word and visual inscription - it undergoes a metamorphosis with each transcription. We can say that it is in the space of projection between these different modes of representation where the parameters of a design are developed and refined. It is the ability to project ideas from one form to another - to convert ethereal speculations into graphic utterances - that allows a designer to work and designs to evolve. Creative activity cannot be assigned to the exclusive domain of either the internalized imagination or the external world, but to the space of transgression between them. Robin Evans says of pictures that "their inexhaustible mystery arises from the fact that they externalize an aspect of perception, or that they appear to externalize it, as if one were seeing the thought itself, which does not happen with words or numbers in the same way." He uses projection as the metaphor for the way the imagination constructs both ideas and concrete things. This notion of projection places importance on the space in between its source and its receiver. "Imagination is not held within the mind but is potentially active in all areas of transition from persons to objects or pictures. It operates, in other words, in the same zones as projection and its metaphors" (Evans 1995). There are several characteristics of representations - and of visual, architectural representations in particular - that give them this property of seeming to transform what they project across the gap between subject and object, and without which design would be impossible.

## REPRESENTATIONS LOSE INFORMATION

Drawings do not conserve information well: the act of describing an idea on paper involves substantial information leakage. This property has important implications.

### Pictures Liberate Their Subject From The Repressive Regime Of The Real

Norman Bryson observes:

“Two impulses, one to resurrect and one to renounce, seem between them to define the painting of the West. On the one hand, what Lévi-Strauss calls the ‘avid and ambitious desire to take possession of the object’, a desire which calls into being all those refinements within the technology of reproduction which for antiquity, as for the Renaissance, constituted painting’s progressive history; and on the other hand, an impulse which runs counter to the first, demands a diminution or sacrifice of the object’s original presence, and strips away from its unwanted repetition aspects which impede the release of ‘aesthetic emotion’” (Bryson 1981).

This second impulse temporarily frees the object from the imperatives of the real world. Although this passage refers to the history of painting, it reveals something important about design practice: propositions for new configurations of the world can only occur outside of the context of a dominant model of reality. A designed object comes into being and evolves in a process that has a distinctive pulse: the imagined object’s set of possible configurations expands when released from the obligations of the real and contracts when subjected to them. Design is made possible by of this subtle interdependence of unimpeded speculation and constraining rationality.

The premises behind the ongoing project of “virtual reality” – which the designers of most software design tools claim to support – are incompatible with these requirements. At first glance, it may seem that tools such as “photo-realistic” raytrace rendering software constrain designs unreasonably because they embed a design too deeply in the parameters of the real world; rather, it is not their realism that constrains (indeed,

these programs forfeit most of the most meaningful aspects of the real) but is the belief on the part of their designers that these tools provide a “trap door” to reality – that a designer can reach through the window of the program and grasp the real-world parameters of the object. In explaining the philosophy behind the design of *formZ*, a popular modeling package, its designers state:

“3d entities and configurations are best created and visualized directly in 3d, rather than through their 2d representational conventions. The system offers extensive tools that facilitate, even enhance, the generation of 3d models directly in their 3d world environment” (From the *formZ User’s Manual*).

Even if these programs did provide this facility, I question the legitimacy of a proposal claiming that designs are best elaborated when they are embedded as deeply as possible in the parameters of the real. “[A]rt begins where an artificial barrier between the eye and the world is erected: the world we know is reduced, robbed of various parameters of its being, and in the interval between world and reproduction, art resides” (Bryson 1981).

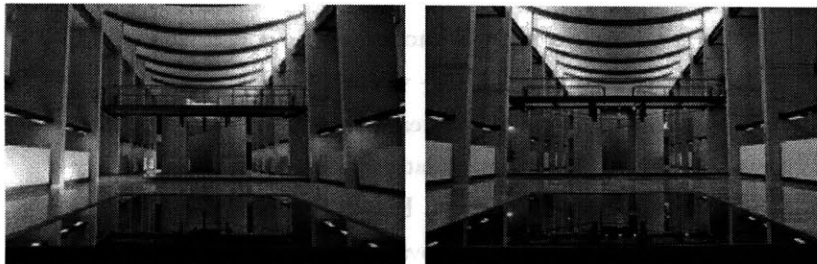
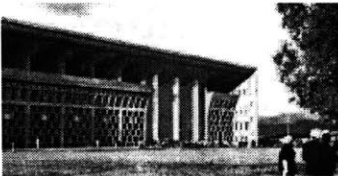


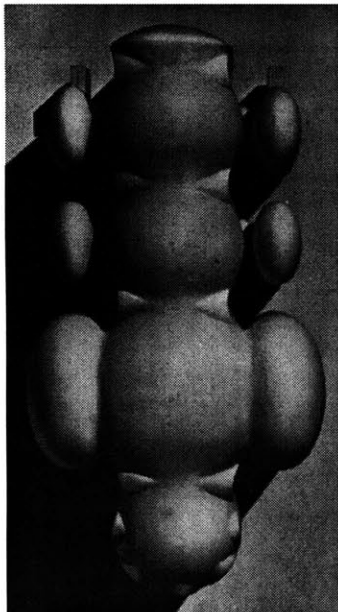
Figure 2  
Sample images from an advertisement for the *Lightscape* rendering system. At left: the computer rendering; at right: “the real thing.”

#### Drawings Require Re-Interpretation

If making a representation of an object forces a renunciation of its real attributes then reading and rebuilding a representation obliges a reconstruction of those attributes. Because it involves a great deal of re-drawing, design is an ongoing project of stripping away and then reconstituting an object’s real-world properties. In the act of building a new representation, the designer reinterprets the designed object: new parameters are added while others are discarded.



**Figure 3**  
Profitable re-interpretation of a sketch. Top: Le Corbusier's sketch of the Pocile wall at Hadrian's Villa, Tivoli. Bottom: His Palace of Justice at Chandigarh. In Boesiger (1957). This comparison was made by Graves (Graves 1981).



**Figure 4**  
Paranoid-critical activity in action. Luigi Moretti's plaster cast of the interior of Guarini's Santa Maria della Divina Provvidenza, 1952. From Evans (1995).

But why is this reinterpretation a necessary and useful part of design? In an unrelated article, Patrick Winston recounts being asked "if I ever had the experience of explaining an idea to someone, only to have the idea misunderstood into an idea that was actually better" (Winston 1997). This notion of a profitable, opportunistic misinterpretation gives us a clue as to the traits that any design medium should have.

#### REPRESENTATIONS FACILITATE MOVEMENT THROUGH DIVERSE MODES OF PERCEPTION

An important characteristic of design drawings is that they encapsulate a multiplicity of perceptual modes in one image. A design drawings allow the designer to rapidly and effortlessly transgress perceptual boundaries: they allow designer to keep more than one pot on the stove. A design is always in a state of flux, which the medium has to describe and encourage.

#### Pictures Can Make the Unreasonable Seem Reasonable

Salvador Dali proposed a method for creating images that exploited the ability of a picture to carry an object from one mode of perception to another. His so-called "Paranoid-Critical Method" has two distinct phases:

"...(a) the synthetic reproduction of the paranoiac's way of seeing the world in a new light - with its rich harvest of unsuspected correspondences, analogies, and patterns; (b) the compression of these gaseous speculations to a critical point where they achieve the density of fact... Paranoid-critical activity is the fabrication of evidence for unprovable speculations and the subsequent grafting of this evidence on the world, so that a 'false' fact takes its unlawful place among the 'real' facts" (Koolhaas 1978).

The ability to act as "objectifying 'souvenirs' of tourism", is an important property of design drawings. It is simplistic to complain that digital design tools constrict intuitive, creative activity because of their "rationalization" of the drawing production process; indeed, these drawings that would be meaningless without this crucial, objectifying function.

Samuel Edgerton identifies a similar property of visual images

(Edgerton 1980). He proposes that the scientific revolution, with its dependence on images that structured the way of seeing the world, had its roots in Renaissance art: “[T]he mathematical aspect of Renaissance art ... allowed it to be used as a special visual language, more communicative than oral or written language, particularly when describing tangible objects.” The rationalizing aspect of these images can be seen in the engraving of the Annunciation made as part of a handbook used by Jesuit missionaries (figure 5). This image renders the real as fantastic - clouds bearing angels - and the fantastic as real - the narrative of the Annunciation sectioned, laid out, and labeled as if on a dissecting table. The artist presents a tableau in which the unreal has been fully naturalized within the same pictorial space as was used to present a dissection of a cadaver or a cutaway view of a mechanical apparatus (figure 5).

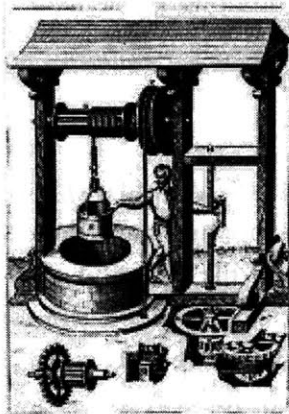


Figure 5  
Left: Agostino Ramelli, *Engraving of a Windlass Pump*, 1588.

Right: Hieronymus Nadal, *Annunciation*, 1593.

From Edgerton (1980).

#### Architectural Drawings Have Both Metrical And Pictorial Properties

Orthographic projection is an important chapter in the development of Western representational schemata. In what is essentially a history of orthographic representation, Peter Booker makes the distinction between

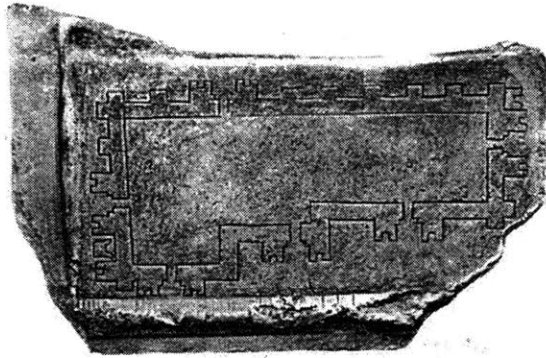
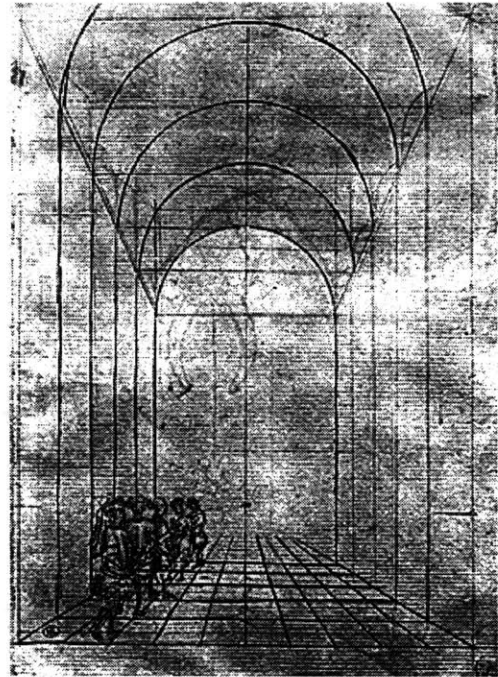


Figure 6

Left: Metrical drawing: Drawing board from statues of Gudea of Ur, c.2130 B.C. In Booker (1963).

Right: Pictorial drawing: Pisanello, Drawing of an interior. In Lotz (1977).



earlier engineering drawings that are “true shape drawings” and later, “pictorial” images (figure 6). As a description of the measurable attributes of an object, the line in a true-shape drawing acts as the equivalent of the rigid measuring instrument used in the real world, a specification for the measuring out of Euclidean space. It acts as a mirror or tracing, an image that is isomorphic with the object it represents. In contrast, the pictorial image transforms the shape of the object as it represents it. Perspective projection is an example of this type of representation. It is in the Renaissance that the distinction between these two modes of graphic representation is explicitly described. In his *Ten Books on Architecture*, Alberti draws a clear distinction between “painter’s perspective” and architect’s drawings, and in doing so sets down one of the canonical early Renaissance ideas about the representation of space. According to Alberti, shading and foreshortening were strictly the realm of the painter. For the architect the laws of *proportio* and *divisio* were only reliably conceived and conveyed in terms of metrical modes of drawing which did not introduce untruthful distortions (Alberti 1955, Lotz 1977). The *a priori* assignment of metrical

drawing to the domain of instrumental action and pictorial drawing to the domain of aesthetic practice is the departure point for subsequent development of technical drawing.

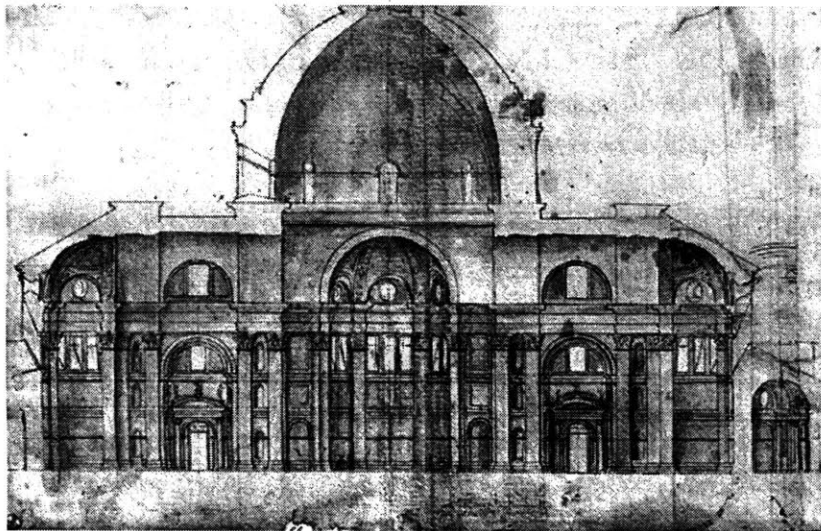


Figure 7  
Antonio da Sangallo the Younger,  
project for St. Peter's. In Lotz  
(1977).

### Sectional drawing

The sectional drawing is the condensation *par excellence* of these two distinct perceptual and representational modes. The sectional view's effectiveness as a design representation stems directly from its double duty as objectified measurement and pictorial presentation (figure 7). The roots of the thought-experiment that would become sectional drawing can be found in Renaissance drawings of classical artifacts in which their hypothetical partial ruination becomes a window to the interior (figure 8). Of this type of drawing, Jacques Guillerme and Hélène Vérin say "[its] merit, indeed its purpose, lies in presenting a drawing of the mental operation which embraces, all at once, the interior and exterior of the edifice as well as the thickness that separates them." In their words, "archaeological curiosity" armed with "the demands of proportion" are what give birth to the section (Guillerme and Vérin 1989, Rudwick 1976).

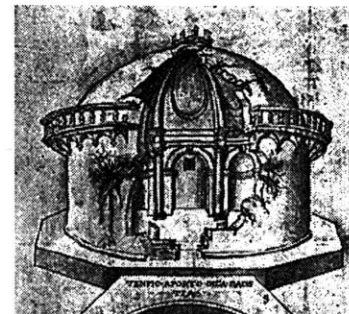


Figure 8  
Giuliano da Sangallo, "Antique  
Circular Temple." In Guillerme and  
Vérin (1989).

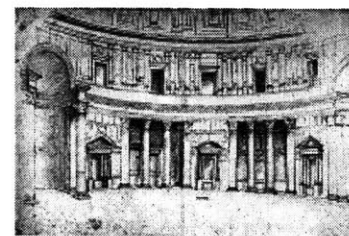


Figure 9  
Sketch of the interior of the  
Pantheon (attributed to Rafael). In  
Lotz (1977).



Wolfgang Lotz outlines a development of the sectional drawing in which the rigid segregation of painter's perspective and architectural drawing dissolves into the amalgamation that would become the canonical form of the sectional representation of space. (Lotz 1977). He describes a historical narrative in which the viewer of the "deep" interior perspective view gradually backs away from the space until the perspective projection has become a flattened orthography (figure 9). The Renaissance sectional drawing is still a collapse of pictorial and metrical descriptions, with the lines and surfaces representing the cut material being a metrical, "true-shape drawing" and the surfaces beyond viewed as a pictorial projection, a pathological perspective view.

#### *Gaspard Monge and Projective Geometry*

But the renaissance section drawing is only proto-orthography: its explicit reference to the metrical procedures of cutting and measuring have yet to be broken by Gaspard Monge's resurrection of Desargues projective geometry. The demand for a reliable instrumentality of visual inscriptions would be met most effectively by Monge's system of descriptive geometry. His aim was to provide a mechanism by which complex three-dimensional spatial problems could be resolved using procedures carried out within the plane of two-dimensional representations. His system relied on two polar but interdependent properties: first, an unambiguous correspondence between the real object and its image was essential if the system was to be instrumental in any way - manipulation of a real object using an image is only as good as the reliability of the correspondence between them; second, this correspondence was ensured by employing a system of projection that lent the inscription the quality of a photosensitive surface. In this way, Monge's schema captured objects onto surfaces using a configuration that relied on an interdependency of metrical and pictorial modes of representation.

With the publication of Monge's *Géométrie descriptive* in 1795 and his classes at the École Polytechnique, there is a significant shift in the meaning of the orthographic drawing. The surface of the inscription conceived of as an amalgamation of knife blade and painter's canvas is reinterpreted as a proto-photographic plate. The metrical qualities of the ob-

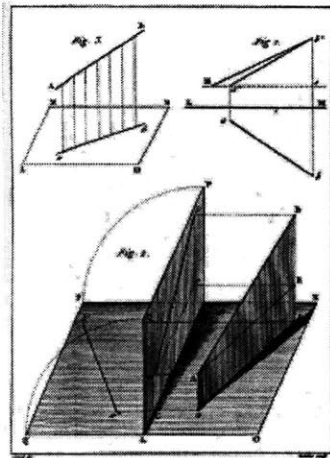


Figure 10  
Gaspard Monge, Plate 1 from the  
*Géométrie descriptive*, 1795.  
In Booker (1963).



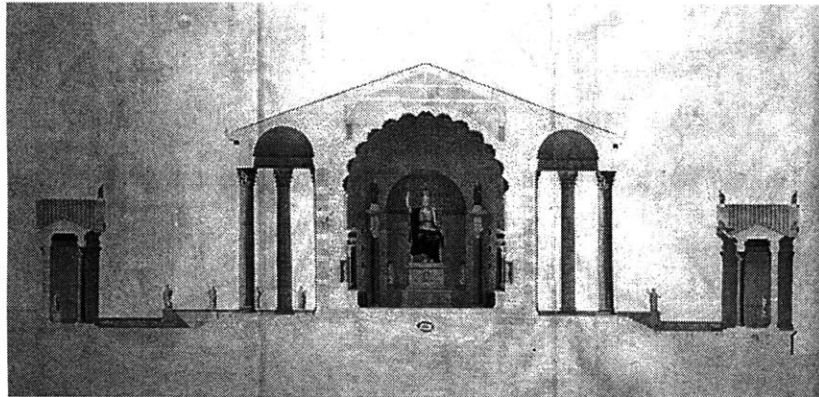
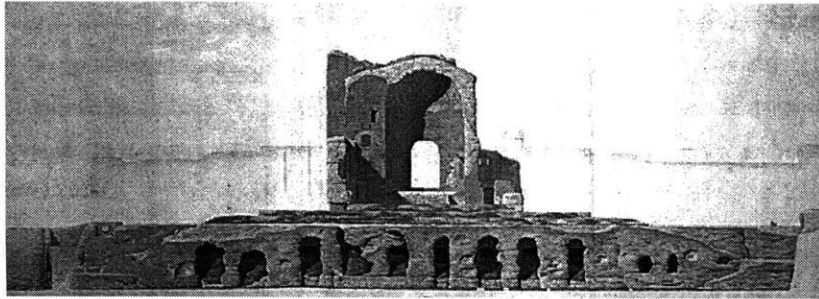


Figure 11  
L.Vaudoyer, "Temple of Venus and  
Rome," 1830. Top: existing  
condition. Bottom: reconstruction.

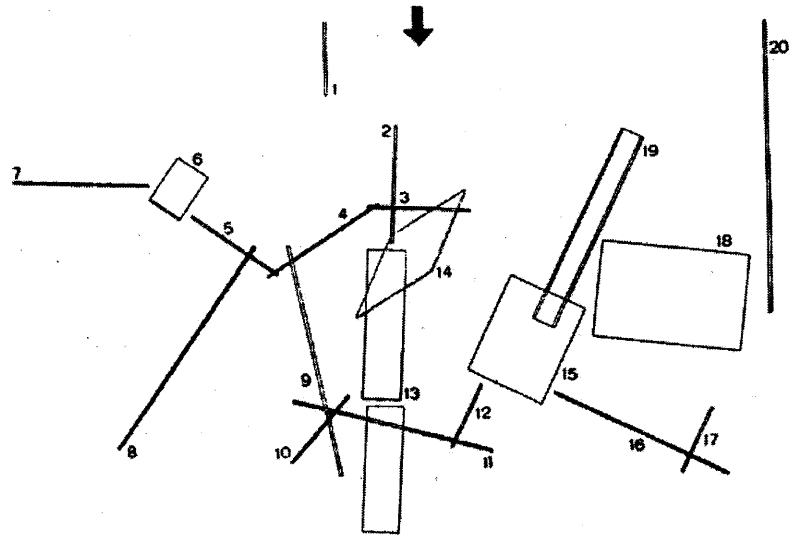
ject projected on it are evaluated in purely optical terms. With Monge, the inscription no longer refers to the physical act of cutting or measuring. It is the inscription itself that is measured.

#### *The École des Beaux Arts and Imaginary Archaeology*

The École des Beaux Arts posed a challenge to the technological ideas being taught in the École Polytechnique. Rather than abandoning Monge's projective modes, their students produced representations of classical ruins in Rome in which orthographic projection acts openly as both picture and description. It is a conceptual net taken into the field and cast over the decaying ruins (figure 11). Its effectiveness in its ability to act as a (mirror or overlay) derives directly from its orthogonality and thus from its character as a surface that is the meeting point of objective surveyor's measurements and the pictorial portrayal of materiality and atmospheric effects. This rationalizing image provides, in Bruno Latour's words, an

“optical consistency” that allows for both the objective recording of classical ruins and by extension their equally “objective” reconstruction to occur within the same pictorial space (Latour 1980). The ruins are represented as bared to observation in the bright light of reason with their

Figure 12  
 Combinability of inscriptions.  
 From a single image we are able to: understand what this space would actually look like if seen from above; precisely determine the relative locations and sizes of objects by measuring the drawing; cause actions to be performed on a group of real objects that exist - or will exist - “out there” in the world. Ernesto N. Rogers, Vittorio Gregotti and Giotto Stoppino (1951) *Plan of the Man-sized Architecture exhibition*, Milan. In Polano (1988).



shadows, calculated with the same techniques of projective measurement, and their meticulously rendered surfaces soliciting confidence in the author as an impartial conveyor of reality. The pictorial mode of these images is thus intimately bound up in their authors’ project of quasi-archeology.

#### Architectural Drawings Are Both Symbolic and Figural Languages

Design drawings are both a language of symbols and a figural descriptive system. We can understand this distinction between “symbolic” and “figural” by considering a controversy resurrected by cognitive scientists in the early 1970’s known as the “Imagery Debate” (Block 1981). The object of this conflict were mental images and the way in which we perceive them. The debate was not so much about the question of whether humans build mental images – there was a general consensus that this did occur; rather, the argument was about their precise mechanism of representation. Broadly speaking, there were two sides to the debate: the “pictorialists” proposed that mental images represented in more or less

the same way that external pictures do (however that might be); the “descriptionalists” claimed that these images in fact do not communicate as pictures but along the same lines as non-imagistic representations such as language, in other words in symbolic terms.

A design drawing can be seen as a surface where symbols and pictures commingle, where the demarcation between them is rendered indistinct; more precisely, the elements in the design drawing are often themselves both symbolic and figural representations of the thing they represent. This condensing of symbolic and pictorial modes of representation is what gives architectural drawings their crucial property of combinability. With reference to this aspect of instrumental visual images Bruno Latour observes:

“To link geology and economics seems an impossible task, but to superimpose a geological map with the printout of the commodity market at the New York Stock Exchange requires good documentation and takes a few inches. Most of what we call ‘structure’, ‘pattern’, ‘theory’, and ‘abstraction’ are consequences of these superimpositions” (Latour 1980).

This attribute of combinability allows formal propositions and abstract data to exist on the same surface (figure 12).

#### Architectural Drawings Are Both Indexical And Literal

When speaking about her pencil drawings of the night sky Vija Celmins describes a process of control achieved through this translation.



Figure 13  
Vija Celmins, Star Field III, 1983.

The act of drawing allows the sky to simply become black graphite. Rather than her drawings resulting from visual observation of the night sky “they came out of loving the blackness of the pencil”. Thus, in response to the challenge that she is “trying to control something as big as the entire cosmos,” she is able to reply that “I am only interested in controlling the space in front of me” (figure 13). The instrumentality of architectural drawings owes a lot to this potential that drawings have to be read at one and the same time as referential and as concrete marks on a surface. Design would be impossible if the lines inscribed on the surface of a drawing incessantly proclaimed their referentiality, if they never simply dissolved into markings of graphite or stains of ink. The weight of each line drawn would be too much to bear.

#### **Pictures Allow Surfaces to Transform Structure**

“[A]rchitectural form is increasingly released from constructionally or geometrically dictated vocabulary. With the tendencies towards de-confinement and increased density, the constructional hardware somewhat recedes in favour of the immaterials neglected until now; suggestive sensory qualities of light, movement, sound, and colour are moving to the centre of architectonic expression.”

This claim of the release architectonic form from geometry assumes an a priori disjunction between internal geometric substructure and external surface appearance, a problem that CAD media bring to the foreground. As with the above discussion of metrical and pictorial modes of representation, what is relevant here is how the inscription made by the designer acts as a surface on which these two distinct notions are resolved; that is to say, it is the place where both internal and external forces are expressed.

#### *Goethe's Theory of Colors*

Goethe's colour theory suggests that a subjective form of vision destabilizes the causal relationship between internal geometric structure and outward appearance. He provides a graphic example of this in his

*Theory of Colors* (figure 14). In this example, subjective vision not only alters the perception of geometric properties, but affects the construction of geometric objects. That is, given the image of a black and white circle of equal diameter, “if the black circle be made larger by so much, they will appear equal” (Goethe 1840) This instability of geometry at the hands of subjective perception is due, as Robin Evans points out, to the fact that “[r]eflection, luster, refraction, luminosity, darkness, colour, softness, absorption, liquidity, atmospheric density, instability of shape: these and a host of other properties jeopardize perceptions of metric uniformity”.

#### *Durand and the Hegemony of Configuration*

But the problem of internal geometry as it relates to external appearance of designed artifacts comes to the foreground with the problematic figure of Jacques-Nicholas-Louis Durand. In 1819, nine years after the publication of the *Theory of Colors*, Durand published his *Précis des Leçons d'Architecture*, a synthesis of the ideas which were at the core of his courses at the École Polytechnique (where Monge's descriptive geometry had been first introduced twenty years earlier). These ideas were eventu-

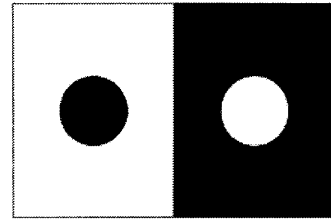


Figure 14  
“A dark object appears smaller than a bright one of the same size.”  
In Goethe (1970).

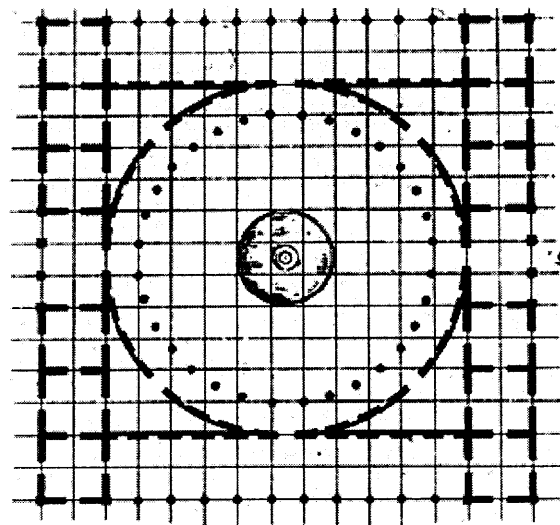


Figure 15  
J.N.L. Durand, plate from the  
*Précis*. In Benevolo (1971).

ally to define an attitude towards the production of built form that would form the paradigm for both contemporary design practice and computational design systems. At the heart of Durand's teachings was a belief that underlying geometric "disposition" was instrumental in determining the economic performance of a project. For Durand, economy and efficiency took the form of moral imperatives, the only truly legitimate values of a design. With the *Précis*, geometry was divorced from specific issues of built form and became lodged in a rationalizing, positivistic methodology.

One way in which Durand's problematic ideas can be understood is in terms of his use of drawings. He used drawings as extensions of self-referential, autonomous rule systems. Geometry and drawing are transformed into instruments of control in the application of immutable rules. His drawings use geometry as a set of regulating lines, numerically controlled systems around which physical form would accrue (*figure*). Evans observes that for Monge, the grid functioned as a conceptual net in which were caught the elaborate curved shapes that had, before that point, been impossible to realize; with Durand, on the other hand, the grid *is* the content: the metric properties of the map become the properties of the items being mapped. But a rule-based system of design such as Durand's could not tolerate the instability of a subjective basis for perceiving geometry; that is, if geometry was to be seen as a manifestation of the application of immutable rules, rules not based on the insubstantial numerology of the Renaissance but on the indisputable authority of the laws of *economy*, then the geometry itself had to be immutable. This requirement resulted in architecture's enforced isolation from contemporary ideas of visuality as a subjective phenomenon and from any notion of the perception of built form as receivable through the senses (Crary 1990). But once the appearance of a structure is no longer a manifestation of an internal geometric order it becomes relegated to the domain of "culture"; that is to say, of style (Benevolo 1971). This new arbitrariness of appearances was the result of this view of geometry as immutable substructure and paradoxically rendered the meaning manifest in surfaces as both a consequence of and independent from internal configuration.

## THE MANUFACTURE OF REPRESENTATIONS EMPLOYS PERCEPTUAL FACULTIES

“The painter ‘takes his body with him’ says Valéry. Indeed we cannot imagine how a *mind* could paint. It is by lending his body to the world that the artist changes the world into paintings. To understand these transubstantiations we must go back to the working, actual body – not the body as a chunk of space or a bundle of functions but that body which is an intertwining of vision and movement” (Merleau-Ponty 1964).

Thought is inseparable from perception. But modernity has seen “thought” generally identified with language. Barbara Maria Stafford argues that, at the beginning of this century, Ferdinand de Saussure’s structuralist project laid the groundwork for this linguistic bias: “The totemization of language as a godlike agency in western culture has guaranteed the identification of writing with intellectual potency.” Moreover, “most damagingly, Saussure’s schema emptied the mind of its body, obliterating the interdependence of physiological functions and thinking” (Stafford 1996). Mark Johnson is more specific about the problem:

“Roughly, the gap is thought to exist between our cognitive, conceptual, formal, or rational side in contrast with our bodily, perceptual, material, and emotional side. The most significant consequence of this split is that all meaning, logical connection, conceptualization, and reasoning are aligned with the mental or rational dimension, while perception, imagination, and feeling are aligned with the bodily dimension. As a result, both non-propositional and figuratively elaborated structures of experience are regarded as having no place in meaning and the drawing of rational inferences” (Johnson 1987).

To produce and to consume images means to engage our sensory, perceptual faculties. If we accept the argument that thought – and imagination – are linked to perception then design activity necessarily demands the use of our sensory apparatus. But, without exception, digital design tools are built under the assumption that we think using a reasoning engine buried deep in our brains – that our visual, verbal, and motor func-

tions are semantically neutral and thus in no way color the information they transmit to the brain (Ullman 1991). As Patrick Winston points out:

“the inner conversation many (all?) people have when they solve problems may play the same role as a conversation with someone else. Processing thoughts expressed as word sequences must excite important thinking mechanisms buried in our language-processing hardware. Thus, the thinking lies in the language-processing hardware, not behind it.” (Winston 1997)



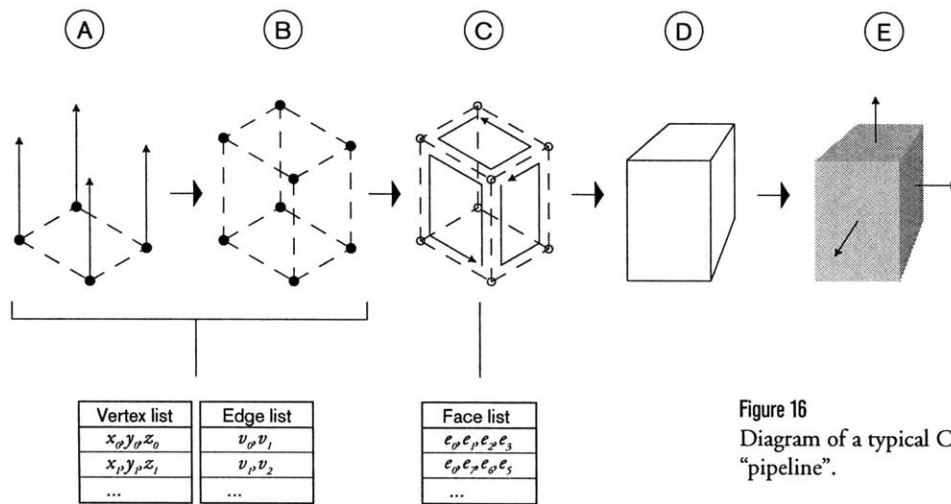


Figure 16  
Diagram of a typical CAD  
“pipeline”.

Modeling:

- A.  
Object creation: the user creates a 3d object using a given procedure, such as specification of a base polygon and an extrusion height.
- B.  
The 3d database: the representation of an object as a set of vertex points in 3d space. Edges are represented as connected pairs of vertices.
- C.  
Finding the surfaces polygons: faces of the object are represented as sequences of edge vertices.

Viewing:

- D.  
Polygon sorting and projection: the back faces of the object are removed and the faces are sorted by depth from the image plane. The points are then projected into 2d screen space.
- E.  
Rendering: the object faces are shaded by computing their angle relative to the light direction.

## SOME PROBLEMS WITH TRADITIONAL CAD SYSTEMS

Having discussed some of the reasons why design would not be possible without visual representations we can see that CAD tools must of necessity respond to the demands we have placed on any architectural representation if they are to contribute meaningfully to design practice. It is instructive to examine the architecture of conventional CAD systems and to consider some of the ways in which CAD systems fail to provide design tools that are an improvement over traditional media.

### THE ARCHITECTURE OF DIGITAL DESIGN MEDIA

For the purposes of our discussion I will describe the architecture of a standard three-dimensional geometric modeling and rendering package (Foley 1996, Hearn and Baker 1997). These principles form the basis of all modeling and rendering programs used by architects (Mitchell and McCullough 1992). These programs can be divided into two main components: the geometric database – with tools for doing transformations on it – and the “rendering pipeline” that allows the display of the geom-

etry as lines, surfaces, or “photo-realistic” scenes. The construction and display of a geometric model proceeds in a more or less linear fashion. First, the user builds the geometric objects by specifying their dimensions and locations in a hypothetical modeling scene. The user edits the objects using standard Euclidean transformations – translation, rotation, reflection, scaling. Display attributes, such as color and surface texture, are then assigned to the objects. Finally, user displays the model according to selected visualization criteria such as viewing position and “camera” type. Granted, this process can be used with a great degree of subtlety and has given architects opportunities to visualize projects before construction in a novel way. There are three main characteristics of this procedure, however, that severely limit the usefulness of this way of building models and images: firstly, the process gives internal geometry hegemony over external appearance; secondly - and this is a corollary of the first point, the process is essentially unidirectional: objects cannot be visualized before they have been unambiguously defined in geometric terms; lastly, the input-output methods that these programs employ insulates the model from the effects of the visual and motor faculties of the designer.

#### **CAD SYSTEMS PROPOSE A DIRECT COMMUNION WITH GEOMETRY**

CAD systems present objects as manifestations of geometric procedures. By this I am not referring to the conventional “wireframe” mode of displaying objects that most geometric modeling systems use; rather, I am referring to a much deeper principle upon which all of these systems are built: users of these tools are asked to reach into a virtual space and manipulate the geometric properties of the objects therein, as if arranging objects on a table.

It is the tacit – and in some cases explicit – aim of the designers of CAD systems to offer a medium that is as transparent as possible, that recedes into the background and offers an unmediated interaction with the designed object. It is my claim that this objective is misguided. If we accept a model for the visual perception of objects that is rooted in subjective vision then in what way can we consider systems of digital visualization that are based on an inflexible dependency of appearance on internal structure as useful representations of the way a designer sees the work?

Geometric modeling systems assume an *a priori* authority of internal structure over outward appearance. Designs residing in the imagination are projected into a Cartesian space in which objects are represented as sets of vertices. The design assumes an increasingly closer correspondence with the “real” with each added level of dimensionality in the Euclidean representation. Points exist to define lines, lines to define surfaces, surfaces to define volumes. Rendering schemes for geometric models depends on the absolute stability of this geometric substructure. In this way, the rendering of appearance and the underlying geometry are kept at a safe distance from each other (particularly with regards to safeguarding against any possible contamination of the geometric substructure by a treatment of surfaces). The procedure known as “bump mapping” does introduce an apparent disturbance of geometry through visual surface information. In this process, a grayscale image is mapped to the surface of a geometric model. The rendering algorithm interprets the gray scale value mapped to a specific point on the surface as the value of a theoretical displacement applied to that point above or below its normal value. This surface disturbance only appears in the rendered image, however, while the underlying geometric structure remains unchanged. This can be seen at the edges of the object (figure 17).

We have seen projects in recent years that are well-intentioned attempts to make a virtue of what seems to be a necessity. Algorithmic, generative design software has been proposed as a way of introducing computers into the earlier stages of design (figure 18). These systems have

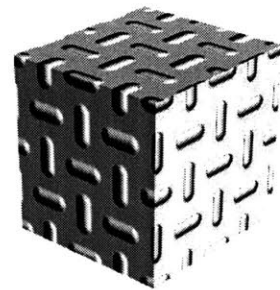


Figure 17  
Procedural bump-mapping.

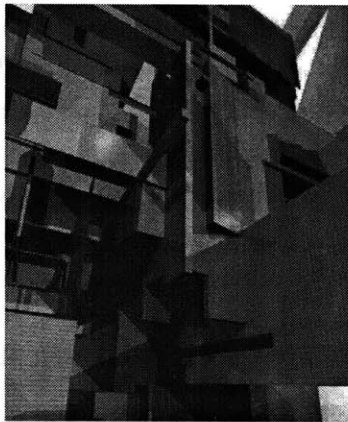
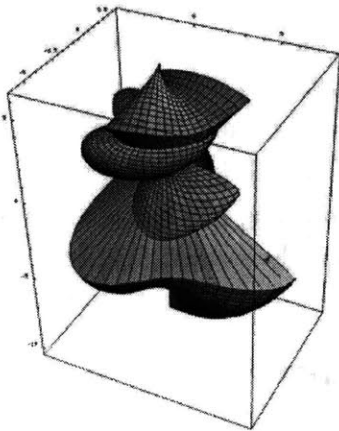


Figure 18  
Examples of designs produced by Marcos Novak using algorithmic techniques.

succeeded in generating forms that arguably would have been unimaginable – or at least impractical to produce – before the development of software design tools. In these systems, an algorithm assumes control over the creation and transformation of the shape geometry, with the designer encoding preoccupations into the software and supplying a “seed” object. But these systems deal only with the machinery that performs transformations on geometric objects. The broad scope of a process that encompasses manual input, geometric transformation, visualization and output of concrete images is compressed so that a sophisticated geometric manipulation engine is bracketed by a parsimonious input method and a trivial visualization system. The gaps between the system building blocks – the spaces of projection – remain ignored. Geometry not only remains unaffected by appearances but renders them irrelevant.

#### CAD SYSTEMS MIMIC THE PRODUCTION OF REAL OBJECTS

With digital geometric modeling systems design unfolds as a linear narrative modeled on the production of real buildings (figure 16). There is a strict production procedure to be followed and accompanying protocols to be observed. The process of construction of an actual building follows a progression from stable substructure to finished surfaces. Designers of CAD systems, unaware perhaps of the differences between the demands of design and exigencies of construction, have modeled their software on this same narrative: the design process commences according to strict rules of geometric stability and soundness, and finishes with a coloring and decorating of the building’s virtual surfaces. As every CAD user knows: the appearance of the final rendering will suffer if your geometric model is poorly built. As I have discussed earlier in this paper, an important function of design representations is to temporarily isolate a design from the contingencies of the real world. To ask architects to design a building in this manner is akin to asking an artist to construct a portrait by painting successive, anatomically accurate, layers of bone, muscle and skin.

#### CONVENTIONAL I/O METHODS INSULATE IMAGINATION FROM PERCEPTION

Geometric modeling and visualization systems disallow input and output (I/O) as constituent parts of the exercise of the imagination. With

reference to the schema illustrated in figure 16 it can be said that design software emphasizes the middle stages at the expense of the two ends. Geometric modeling software assumes an unambiguous input stream fed by a hand which simply reiterates a description of a predetermined design, resident in the imagination, in which object can not be probationarily approximated or cautiously circumnavigated, nor can it be allowed to be excavated from an accumulation of physical inscriptions. The alternative is not software designed to approximate “sketching” or of replacing keyboard input with such tropes as the pressure-sensitive stylus. This allusion to familiar and traditional techniques directs attention away from the site of imaginative activity: the zone of projection between the tool used for input and the tool used for modeling. Whether one uses a keyboard, a mouse, or a digital scan of a manually executed drawing, it is the concealed algorithm, converting continuous physicality into a digitally encoded mathematical representation that assumes control over the projective space of the imagination. Similarly with output technology, the exact characteristics of the physical material that forms the support for the inscription has less effect on its meaning than the hidden algorithm that transforms the manipulated image from the luminescent, rendered image on the screen to the matrix of data read by the output device. Thus, recent developments in physical prototyping technology tend to shift the emphasis away from the artifact and towards the author as the reader of the unequivocal meaning surrendered by a physical representation of a design.

There is a common thread among all of the criticisms I have mentioned: the problems I have presented emerge out of a transfer of power from the designer to the medium used to make images. The space of projection referred to by Evans in which much of the work of design gets done corresponds the gaps between the major components in the system architecture diagram (Evans 1986). The result is that the algorithms that convert hand gestures to geometry, that transform geometry into visual information, and that solidify visual information into concrete objects are kept hidden from the user. The problem is that these algorithms stake a claim to the area of design where most of the interesting work gets done.

## THE NATURE OF ARCHITECTURAL DESIGN ACTIVITY SHOULD INFORM A PROPOSAL FOR A SOFTWARE TOOL

Using the characteristics of representations that architects use in practice, we can start to form a picture of a useful computational design tool. But these criteria are an incomplete picture: the topography (?) of the design process changes from one architect to another. If we are to succeed in embedding a digital tool more meaningfully into design practice we have to take clues from the nature of the design methodology in question. In other words, I am raising doubts as to whether it is possible - or even desirable - to build a tool that is universally useful. My proposal for a software design tool is thus based on specific aspects of the way that I design buildings. I believe, however, that some of these aspects are instructive about architectural design in general, and thus about how we might design a better CAD tool.

### USUALLY, DESIGN PROCEEDS BY MAKING SPECIFIC PROPOSALS

Computer interpretation of freehand sketches is a classic computer intelligence problem (Negroponte 1975, Gross 1996). But the assumed properties of freehand sketches that inform these programs render them of limited use for architects. All too often, the designers of these programs assume that sketches are extremely loose, hazy approximations of unformed ideas – that designers cautiously close in on an idea by stalking it from an underbrush of brisk and noncommittal gestures (figure 19). Many of these programs focus, therefore, on interpreting the *intention* of the designer. (At what point does a blob become a rectangle? At what point does a triangle become a roof?) Doubtless, there are many architects who resolve design problems in the manner of a gradual solidification of approximations; nonetheless, I claim that a great deal of the explorations that typify design practice take the form of specific proposals with clearly readable formal properties (figure 20). Rather than a linear process of convergence, design often takes the form of a cycle of proposition, critique and counter-proposition.

## A DESIGNER MOVES 2 STEPS FORWARD, 1 STEP BACKWARD

A corollary to the designer's inclination to resolve problems by making specific formal proposals is the tendency of design to proceed with a characteristic, lurching gait that results from its cyclic nature. As it is difficult to engage in a meaningful discourse about a design that does not yet exist, proposals are necessarily developed beyond their "appropriate" level of detail; thus, the linear process that most CAD systems oblige of their users is of limited use in architecture design. Visualization and representation are as important – perhaps more so – when a design is malformed than after it has been resolved.

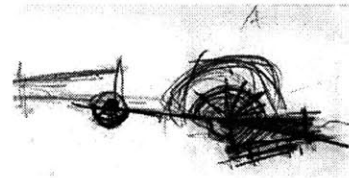


Figure 19  
Example of a design sketch, used by Gross and Do in a description of their program that interprets such sketches. Gross and Do (1996).

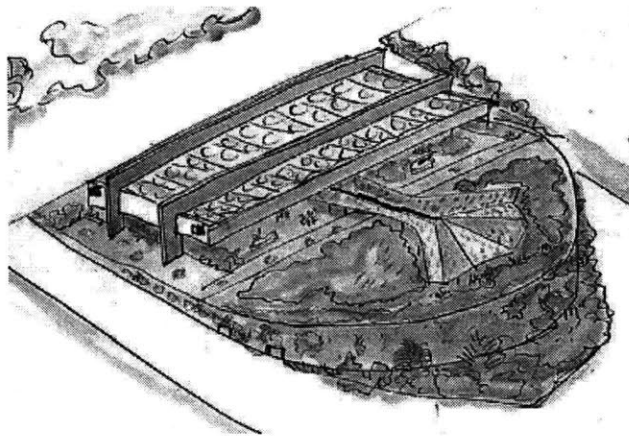


Figure 20  
Another type of early design sketch. Lina Bo Bardi, First Study for the São Paulo Art Museum, 1957. In Ferraz (1994).

## **A USEFUL DIGITAL DESIGN TOOL SHOULD EXPLOIT THE ENDOWMENT OF VISUAL REPRESENTATIONS**

We have outlined some demands for a software design system that could allow for a more meaningful integration into early design. Specifically, how can this be done?

### **IT SHOULD RESPOND WITH ITS OWN INTERPRETATIONS**

A useful tool responds with its own interpretation of the design. Drawings on layers of tracing paper reveal emergent solutions. A model viewed from a different position in physical space discloses unforeseen designs.

### **IT SHOULD ALLOW FOR EASY MOVEMENT BETWEEN CONJECTURE AND CONSTRAINT**

Designers rely on the freedom to make speculative proposals and the obligation to criticize these them using rational, real-world criteria. Visual representations have provided this facility in the past.

### **IT SHOULD ALLOW FOR EASY TRANSACTIONS BETWEEN METHODS OF PICTURING AND METHODS OF MEASUREMENT**

Design drawings are at the intersection of metrical and pictorial modes of representation. This property has allowed designers to develop designs synchronously in both aesthetic and instrumental terms.

### **IT SHOULD CAUSE AN ENGAGEMENT OF VISUAL AND MOTOR FACULTIES**

Much of the power of representations comes from their being manufactured objects. Building an image by hand engages the designer's visual and motor sensory faculties. The employment of sense perception is directly linked to imaginative thought.



## PART 2:

### BUILDING GEOMETRIC MODELS WITH SKETCHES: A Software Tool For Early Stages of Design

Digital geometric models are useful in architectural design. I believe that they could be more so if different means were provided to build and manipulate them. I am proposing a software tool in which freehand sketches are used as input and editing methods for geometric models. The user of the program manipulates models through the mediation of images drawn on the picture plane. By linking a traditional means of producing architectural representations with a geometric database I am proposing a tool that can respond to some of the demands I have set out in this paper.

#### FREEHAND SKETCHING CAN BE A WAY OF CREATING MODELS

Geometric models are built using strict procedures that map hand movements and mouse clicks to points in an imaginary three-dimensional space. We can insert a mediating representation between the hand and the model (figure 21). Thus, these same hand movements can be more



Figure 21  
Albrecht Durer, *The Painter's Manual*. From Ferguson (1993).

profitably used to create a two-dimensional drawing. The drawing becomes the interface between the computer and the designer: the designer works on the sketch while the computer reads the sketch. No additional information is provided to the machine: the sketch is all it has to work with.

#### SKETCH INTERPRETATION INVOLVES THE POSSIBILITY OF MISINTERPRETATION

The use of a two-dimensional drawing as an interface brings with it the possibility of misinterpretation inherent in any form of mediated communication. As I have discussed earlier in this paper, profitable misinterpretation is an important aspect of design. It allows designs to be dislodged from the confines of preconceived restrictions, from assumed parameters, from emergent constraints. Misinterpretation allows designs to evolve.

#### SKETCHING ALLOWS FOR RAPID TESTING AND EDITING OF IDEAS

Freehand sketching is efficient. Ideas can be expressed quickly and with minimal resources. Sketching provides a concise notational system – shadows, dashed lines, text labels – that allow for rapid revision and communication of proposals.

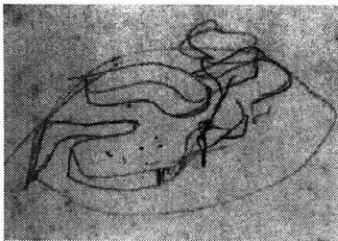


Figure 22  
Lina Bo Bardi, Sketch of the  
Chame-Chame House, 1958. In  
Ferraz (1994).

# USING THE PROGRAM

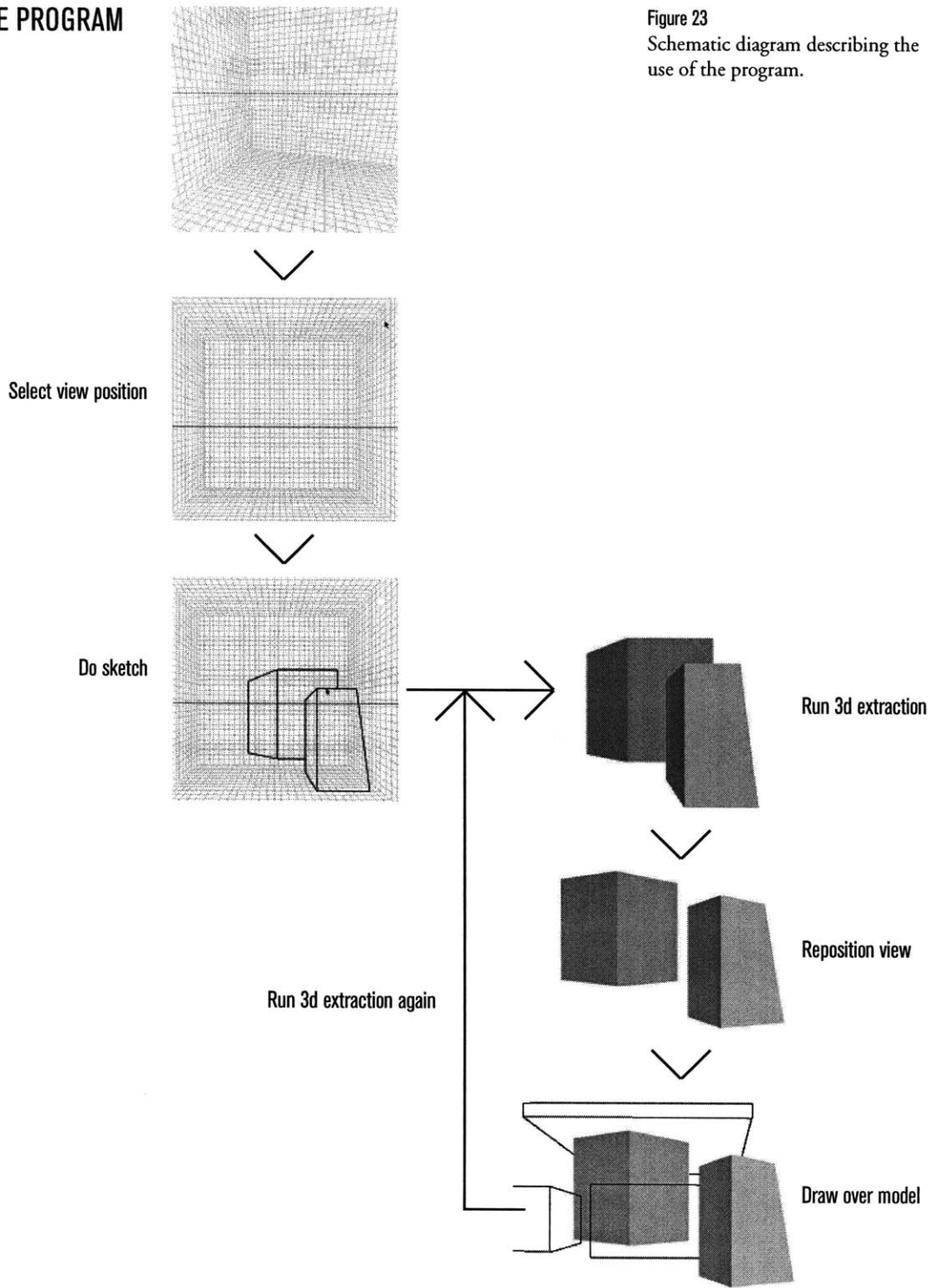
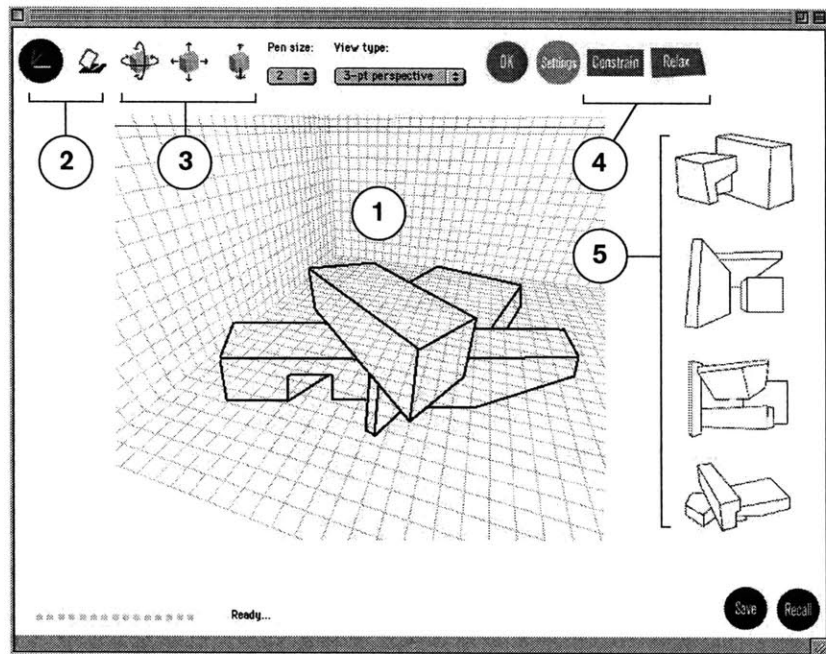


Figure 23  
Schematic diagram describing the use of the program.

Figure 24

The program's graphical user-interface.

1. Drawing area
2. Drawing tools
3. View selection tools
4. Regularity strictness controls
5. Sketch library



## OVERVIEW OF THE PROGRAM DESIGN

The program has the following components:

- Sketch input

- Preprocessing and vectorization of the sketch image

- Scene segmentation and analysis of structural relations

- Image regularization and cleanup

- Recovery of 3d vertex locations

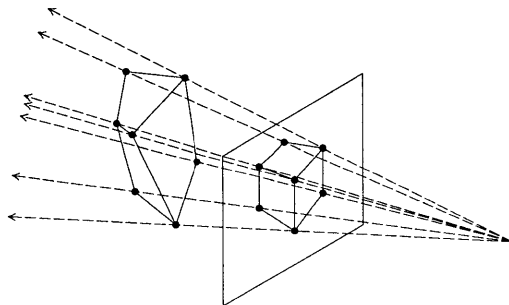
- Display of the 3d scene as a rendered geometric model

### SKETCH INPUT

The program records the user's sketch in a pixel buffer. It does not, as most CAD systems do, record the users gestures as a series of connected vertex locations. I made this decision because a pixel array contains no geometric information. Like marks of graphite on paper, pixels allows the user to make inscriptions, edit them, erase them and add to them independently of a strict procedure of geometric construction. Lines can be

drawn in any order and with any degree of definition. The program will ignore marks that are unresolvable, such as guidelines, hatching and notes. Only when the user invokes the 3d recovery routine does the program attempt to make any geometric sense of the drawing.

A major problem in the reconstruction of three dimensional scenes from single images is the recovery of the viewing position. A an object in a single image corresponds to an infinite number of possible shapes in three-dimensional space (figure 25). To overcome this difficulty, this version of the program requires that the user set the desired “camera” position before starting the sketch; however, it would be possible to use clues in the image to infer possible camera parameters thus allowing the use of a digitized sketch on paper to be used as a source image. This difficulty is overcome once the initial model is built: Modifications and additions to the model through subsequent sketching use the currently displayed view of the model as the known view position. I should point out that in inferring 3d scene descriptions from single, 2d images there is an inverse relation between the amount of knowledge available about the viewing parameters and the necessary assumptions about the regularity of objects in the scene; that is to say, the less we know about the viewer, the more we have to look for clues such as local symmetries, repetition and compactness among the objects in the scene (Kanade 1981, Shomar 1986).



**Figure 25**  
The difficulty of recovering a 3d object from a single view. Each point on the 2d image maps to an infinite set of points in 3d space. What we perceive to be a regular polyhedron in the image has an infinite number of possible shapes in 3d space.

#### IMAGE PROCESSING AND VECTORIZATION

When the user asks the program to analyze the sketch, the pixel buffer is first converted to a binary image. It is then processed using a morphological closing operation that eliminates small gaps between lines followed

by a morphological opening operation that deletes stray pixels and smooths noisy line edges. A one-pixel wide, 8-connected skeleton is then found using a derivation of the Zhang-Suen thinning algorithm (Parker 1997). The skeleton is analyzed to find the set of feature-points. These points correspond to pixels with either one, or three or more neighbors (analogous to line endpoints, branch points or meeting points of multiple lines). The connected chains of pixels between pairs of feature points are processed using a curvature-based corner detecting method (O’Gorman 1995). The output at the end of this stage is a connectivity graph of feature points with polygonal descriptions of their connecting lines. The binary image is preserved and stored in a sketch library.

### SCENE SEGMENTATION AND STRUCTURAL RELATIONS

The vertex graph is then analyzed using a combination of Mahabala’s vertex labelling algorithm and Guzmán’s scene segmentation algorithm (Guzmán). These algorithms work by first labelling all the feature points in the graph as endpoints, arrows, forks, t’s, k’s, x’s and peaks. The labelled vertices are then analyzed based on a set of heuristics derived from assumptions about object solidity and rigidity. It is assumed that all vertices are trihedral - that is, each vertex is the meeting point of three planar faces. The goal of these algorithms is to arrive at a description of a scene in which vertices and lines are assigned to discrete objects (figure 26). It should be pointed out that Guzmán’s algorithm does its work entirely in the domain of a 2d line drawing. No knowledge of the 3d scene is required, nor is any knowledge as to expected shapes of objects. For example, “T” vertices in a 2d image give clues to object occlusions in 3d space.

The segmented, 2d scene is then analyzed in order to define spatial relations between the objects identified in the previous stage (Winston 1975). I have assumed one basic type of relation that can be described as “occluding/occluded by”. Each occlusion relation is encoded as a link between objects in the scene. Labels are added to the relations when more precise conditions can be identified; for example, an “above/below” label is added when it can be determined with a reasonable degree of certainty that an

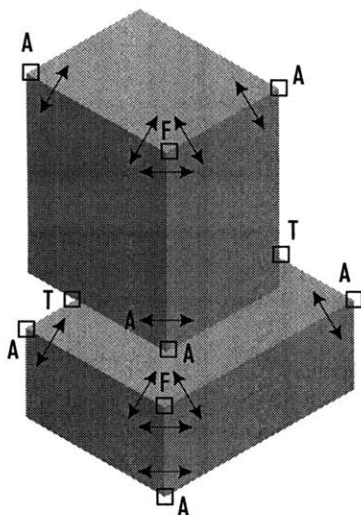


Figure 26  
Scene segmentation using Guzmán’s algorithm. The arrows indicate connections between faces resulting from the vertex analysis.

Vertex types:  
A = arrow  
F = fork  
T = tee

object is above or below another (Winston 1975). The set of objects and relations is then searched for vertices that meet certain assumed criteria about abuttal and contact; for example, it is assumed that, in the absence of information to the contrary, an arrow vertex of an occluding object that bounds a face of an occluded object touches the occluded surface (figure 27).

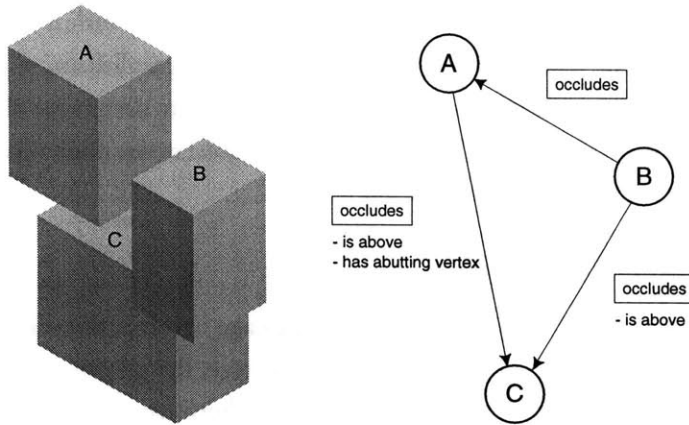


Figure 27  
Description of spatial relations  
between 2d objects.

Left: the segmented scene  
Right: the spatial relation graph  
showing occlusions and with labels  
for above-below relations and vertex  
abuttals.

I chose to use Guzmán's algorithm for various reasons. It recombines objects in scenes by using rigid, planar surfaces, an approach that I find intuitive and easy to relate to the way in which I sketch objects. In addition, it works with trihedral vertices. This means that it does not accept thin, "origami" constructions but demands that objects be well defined as polyhedra. I find that this approach, too, relates well to the way in which I think about objects that I'm drawing. Finally, and most importantly, Guzmán's algorithm does not rely on any knowledge about shapes it expects to find. This is significantly different than the approach that CAD systems take of building up designs from sets of predefined geometric primitives. This algorithm examines local conditions and makes no assumptions about overall shape.

At the end of this stage, we have as an output a description of a 2d scene in terms of a set of objects defined by connected vertices and related to one another by links describing spatial relations and abuttal constraints. It should be noted again that, at this point, the program is still working within the realm of a 2d image. Only now do we proceed to the 3d recovery stage.

## REGULARIZING THE SCENE

Prior to running the recovery routine, the user specifies how zealous the program should be in concluding that sets of lines are parallel and assuming that objects align with the viewing direction. By clicking on buttons labelled "Constrain" and "Relax" the user can automatically adjust the parameters by which the program makes these decisions.

The program first looks for parallel line candidates within objects and, if they fall within the specified tolerances for parallelism, adjusts them so they are parallel. Next, the program attempts to determine if any object in the drawing contains edges that run parallel to the main viewing direction and, if it does, locates it on this axis.

## THREE-DIMENSIONAL SCENE RECOVERY

Reconstructing a 3d scene from a 2d image involves encoding assumptions about a design vocabulary. In this stage, the program uses assumptions about 3d properties of, and relations between, objects and attempts to reconstruct a plausible scene. There are three phases in the recovery of the 3d scene:

### *Rebuilding Occluded Bodies*

The program first attempts to rebuild parts of objects that are occluded by other objects. It assumes that two lines disappearing behind an occluding object can be extended to their point or intersection if this point lies within the boundary of the occluding object. The program also assumes that three or more lines disappearing behind an occluding object and intersecting an area that is sufficiently small can be extended to a common point of intersection.

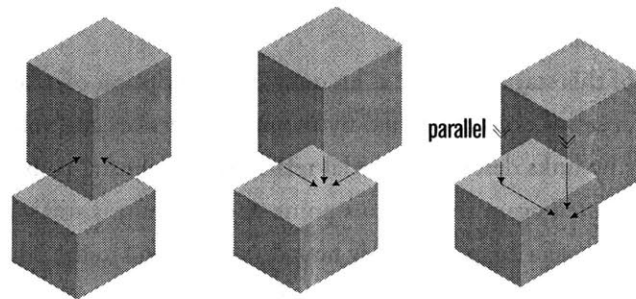


Figure 28  
The conditions under which an occluded body can be reconstructed.



### *Projecting Vertices into the Scene*

This step is the core of the 3d recovery algorithm. The program assumes that the scene has been drawn on an arbitrary ground plane that exists in the 3d scene description. The program also assumes the existence of gravity: it starts at the bottommost vertex in the image and assumes that, unless there is information to the contrary, this vertex is resting on the ground plane. A ray is projected from the viewing position in the 3d scene through the vertex location on the picture plane until it intersects the ground plane of the scene. An initial, 3d point in the scene is established in this way. The program examines the remaining points in the object that are members of the same edge as the starting point and projects them onto the ground plane. After it has projected all vertices in the scene that rest on the ground, the program examines each of the edges adjoining them and, if its angle suggests that it is perpendicular to the ground, located the vertex at the opposite end of this edge. After the program has located all the top ends of edges perpendicular to the ground, it attempts to locate remaining vertices. Each face of the objects in the scene is examined. If it has at least three vertices that have been located in 3d space its plane equation is derived and rays are projected from the view point through any remaining vertices bounding the face until it intersects the plane. If there are remaining un-placed vertices, the program examines any spatial relation labels attached to objects that have been located. The program proceeds recursively in the manner described above until it has located all the vertices.

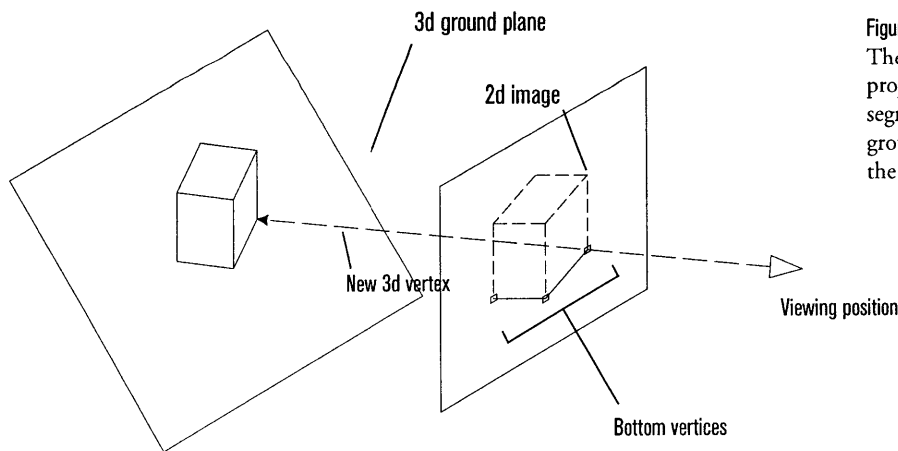


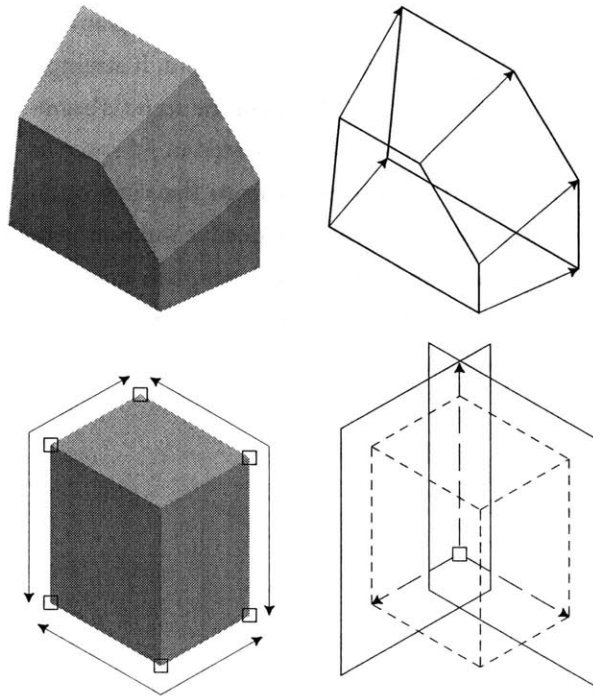
Figure 29  
The intersection point of a 3d ray, projected through a vertex on the segmented 2d image, and the ground plane in the 3d scene yields the vertex position in 3d space.

### *Rebuilding the Backfaces*

In the final stage of the scene recovery the program attempts to build the faces of the 3d objects that are turned away from the viewer. The program uses three basic assumptions to infer the backfaces. First, any object that has a non-occluded face that is connected to a backface by two or more edges is a candidate for an extrusion operation (diagram). Second, the backfaces of any object that has three connected arrow vertices on its boundary can be recovered using the plane equations of the object's boundary points (diagram). Third, all backfaces are assumed to be simple polygons. The program finds the backfaces by a combination of the ray-tracing method described in the previous section and a simple intersection of 3d planes.

**Figure 30**  
The two conditions under which backfaces of objects can be recovered.

Top: extrusion condition  
Bottom: intersection of planes



## DISPLAYING AND REDRAWING

The extracted 3d vertices are made into a connectivity graph and exported to a viewing module. The viewer uses simple wireframe or polygon rendering. If the user draws over the view of the model, the scene recovery module runs again but incorporates the existing geometric model into the spatial relations and vertex projection algorithm. The program exports a text file in .rad format to be used as a source for the Radiance radiosity rendering application.

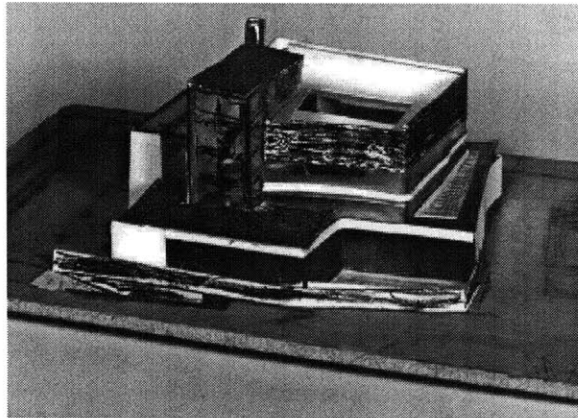


Figure 31  
Gerrit Rietveld (1963). Model for  
the Van Gogh Museum,  
Amsterdam. In Slothouber (1997).



## ASSESSMENT AND EXTENSIONS

### LIMITATIONS OF THE CHOSEN MODEL

Freehand sketch interpretation relies on the encoding of a vast set of assumptions. Inferring 3d geometry from single 2d images, while a trivial problem for humans, is a vastly difficult problem for computers. In order for the program to make useful inferences it has to be supplied with assumptions about object regularity and rigidity; so, we might ask, what is the advantage of a system for design that severely restricts the possible shapes of the objects that can be produced with it?

Another limitation of the chosen model emerges when we consider that architectural design is often a collaborative activity. The problem arises: how can we build a design tool that extends a method of working that is particular to one architect while allowing easy transactions among designers? This problem is implicit to any attempt to design a tool that acts, in a sense, as another design team member. The opposite approach is to try to design a tool based on the model of the graphite pencil; that is to say, a tool which, despite having specific internal attributes, is as transparent and neutral as possible when used for designing, and recedes into the background when communicating a design to others. This seems like a lost opportunity: computers provide a means to design a tool that can more actively contribute to the development of a design.

### POSSIBLE EXTENSIONS OF THE WORK

#### STUDY TYPES OF PROJECTION

The geometric model created by the program is a measure of the disjuncture between the designer's intention and the perspective regime. This mismatch occurs because the algorithm that projects the 2d points back into 3d space and projects the model onto the surface of the screen

makes certain assumptions about the parameters of the viewer's eye. A deeper investigation of projection algorithms and possible mutations of them could provide suggestions for extending the program's capacity for proposing novel forms.

#### **EXPAND THE NOTATIONAL SYSTEM**

In making freehand sketches, the designer employs an extensive system of notation for describing conditions that the chosen mode of representation leaves ambiguous. For example, a shadow might be added to indicate that one surface lies in front of another, or a dashed line might be used to show that an object extends behind another. If the program were to have a notational system such as designers use when making sketches, then many of the assumptions that are necessary to infer 3d objects - and, in doing so, limit the formal vocabulary - would become unnecessary.

#### **KEEP A RECORD OF SKETCHES AND MODELS**

Sketches that the user makes can be stored in a library and overlaid on the current model. A sketch library can become a concise record of design decisions and can allow for the recovery of discarded designs.

#### **SUPPORT MULTIPLE-USERS**

The system can easily be extended to support multiple users. My choice to implement it in Java came from an intention that the program be run over a network and that a group of designers could draw over the same model at the same time, much as they do using drawings in offices today.

#### **MAKE THE PROGRAM LEARN THE USER'S INTENTIONS AND GESTURES**

A significant improvement to the program would be a capacity to learn the user's graphic style and design intentions. Knowledge of the user could be gleaned from comparisons of the sketches with the changes and corrections that the user makes to the program's interpretation of them.

#### **INTRODUCE OTHER IMAGES**

The program can be easily modified to support sketching over photographs. The commonly used technique of photomontage can be extended

into digital form, allowing the user to sketch a proposal over a site photograph, for example, and quickly examine the resulting model.

#### **BUILD IN EXTERNAL CONSTRAINTS**

Many non-formal constraints can be encoded into the scene recovery algorithm to assist and confine the generation of the model. Some of these constraints might be: floor-area ratios, site boundaries, floor-to-floor heights, programmatic requirements such as spatial adjacency, and building code and zoning restrictions.

#### **PROVIDE BETTER DRAWING METHODS**

The requirement that the user use, at worst, a mouse and, at best, a stylus and tablet severely restrict the usefulness of this tool. The program supports scanned drawings, but an electronic drawing tool in which the user can see the drawn lines at the same location as the tip of the pen would be a significant improvement.

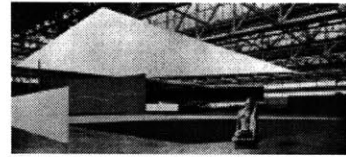


Figure 32  
Ludwig Mies van der Rohe, Collage  
of Concert Hall Project, 1942. In  
Schultz (1985).





## REFERENCES

- Alberti, Leone Battista (1955). *Ten Books On Architecture*, ed. J. Rykwert. London: A. Tiranti.
- Auer, Gerhard (1994). Editorial. *Daidalos* 51.
- Benevolo, Leonardo (1971). *History of Modern Architecture*. Cambridge, Massachusetts: MIT Press.
- Block, Ned, ed. (1981). *Imagery*. Cambridge, Massachusetts: MIT Press.
- Boesiger, W. (Ed.) (1957). *Le Corbusier: Oeuvre complète 1952-1957*. Zurich: Les Editions d'Architecture Zurich.
- Booker, Peter (1963). *A History of Engineering Drawing*. London: Chatto and Windus.
- Bryson, Norman (1981). *Word and Image: French Painting of the Ancien Regime*. Cambridge, England: Cambridge University Press.
- Crary, Jonathan (1990). *Techniques of the Observer*. Cambridge, Massachusetts: MIT Press.
- Edgerton, Samuel Y. (1980). "The Renaissance Artist as Quantifier." In Margaret A. Hagen (Ed.), *The Perception of Pictures*, vol. 1. New York: Academic Press.
- Evans, Robin (1986). "Translations From Drawing to Building." *AA Files* 12: 16.
- Evans, Robin (1995). *The Projective Cast: Architecture and its Three Geometries*. Cambridge, Massachusetts: MIT Press.
- Ferguson, Eugene S. (1993). *Engineering and the Mind's Eye*. Cambridge, Massachusetts: MIT Press.
- Ferraz, M. C. (Ed.) (1994). *Lina Bo Bardi*. Milan: Edizioni Charta
- Foley, James D. et al, (1996). *Computer Graphics: Principles and Practice*. Reading, Massachusetts: Addison-Wesley.
- Goethe, Johann Wolfgang (1840). *Theory of Colors*. Cambridge, Massachusetts: MIT Press.
- Graves, Michael (Ed.) (1981). *Le Corbusier: selected drawings*. New York : Rizzoli, 1981.
- Gross, Mark. D. and Do, Ellen Y. (1996). "Ambiguous Intentions: A Paper-like Interface for Creative Design." In Brown, Marc. and Rao, Ramana. (Eds.), *Proceedings of the Ninth Annual Symposium for User Interface Software and Technology*. Seattle.

- Guillerme, Jacques and Vérin, Hélène (1989). "The Archaeology of Section." *Perspecta* 25: 228-231.
- Guzmán, Adolfo (1968). "Computer Recognition of Three-Dimensional Objects in a Visual Scene," PhD Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts.
- Hearn, Donald and Baker, M. Pauline (1997). *Computer Graphics*. Upper Saddle River: Prentice Hall, Inc.
- Johnson, Mark (1987). *The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reasoning*. Chicago: University of Chicago Press.
- Kanade, Takeo (1981). "Recovery of a Three-Dimensional Shape of an Object from a Single View." *Artificial Intelligence* 17: 409-460.
- Koolhaas, Rem (1978). "Dali and Le Corbusier: The Paranoid-critical Method." *Architectural Design* 2-3: 152-163.
- Latour, Bruno (1980). "Drawing Things Together." In Lynch, Michael and Woolgar, Steve (Eds.), *Representation in Scientific Practice*. Cambridge, Massachusetts: MIT Press.
- Lotz, Wolfgang (1977). "The Rendering of the Interior in Architectural Drawings of the Renaissance." In *Studies in Italian Renaissance Architecture*. Cambridge, Massachusetts: MIT Press.
- Mahabala, H. N. (1969). "Preprocessor for Programs which Recognize Scenes." Massachusetts Institute of Technology Artificial Intelligence Laboratory Memo 177. Cambridge, Massachusetts.
- Merleau-Ponty, Maurice (1964). "Eye and Mind." In *The Primacy of Perception*. Chicago: Northwestern University Press.
- Mitchell, William J. and McCullough, Malcolm (1992). *Digital Design Media*. Cambridge, Massachusetts: MIT Press.
- Negroponte, Nicholas (1975). *Soft Architecture Machines*. Cambridge, Massachusetts: MIT Press.
- O’Gorman, Lawrence (1995). "Curvilinear Feature Detection from Curvature Estimation." In O’Gorman, Lawrence (Ed.), *Document Image Analysis*. Los Alamitos: IEEE Computer Society Press.
- Parker, J.R. (1997). *Algorithms for Image Processing and Computer Vision*. New York: John Wiley and Sons, Inc.

- Pérez-Gómez, Alberto (1985). *Architecture and the Crisis of Modern Science*. Cambridge, Massachusetts: MIT Press.
- Polano, Sergio (1988). *Mostrare*. Milan: Edizioni Lybra Immagine.
- Richens, Paul and Schofield, Simon (1997). *Piranesi* rendering system.  
[http://www.informatix.co.uk/pir\\_intro.htm](http://www.informatix.co.uk/pir_intro.htm)
- Rudwick, Martin J.S. (1976). "The Emergence of a Visual Language for Geological Science 1760-1840." *History of Science* 14: 149-195.
- Schulze, Franz (1985). *Mies van der Rohe: A Critical Biography*. Chicago: University of Chicago Press.
- Shomar, Wasim J. and Young, Tzay Y. (1986). "Three-Dimensional Shape Recovery from Line Drawings." In Young, Tzay Y. and Fu, King-Sun (Eds.), *Handbook of Pattern Recognition and Image Processing: Computer Vision*. Orlando : Academic Press.
- Slothouber, Erik (Ed.) (1997). *The Artschools of Gerrit Rietveld*. Amsterdam: Uitgeverij de Balie.
- Stafford, Barbara Maria (1996). *Good Looking: Essays on the Virtue of Images*. Cambridge, Massachusetts: MIT Press.
- Ullman, Shimon (1991). "Tacit Assumptions in the Computational Study of Vision." In Gorea, Andrei (Ed.), *Representations of Vision: Trends and Tacit Assumptions in Vision Research*. Cambridge, England: Cambridge University Press.
- Winston, Patrick Henry (1975). "Learning Structural Descriptions From Examples." In Winston, Patrick Henry (Ed.), *The Psychology of Computer Vision*. New York: McGraw-Hill Book Company.
- Winston, Patrick (1997). "Why I am Optimistic." Web publication, <http://www.ai.mit.edu/people/phw/optimism.html>.