DOCUMENT RESUME

ED 036 116 EF 004 026

AUTHOR MACDONALD, STEPHEN L.; WEHRLI, ROBERT

TITLE AN ARCHITECTURAL DESIGN SYSTEM BASED ON COMPUTER GRAPHICS.

PUB DATE 66

NOTE 17P.; SPEECH PRESENTED AT NATIONAL RESEARCH CONFERENCE ON ARCHITECTURAL PSYCHOLOGY (2ND, PARK CITY, UTAH, 1966)

EDRS PRICE EDRS PRICE MF-$0.25 HC-$0.95

DESCRIPTORS ARCHITECTURE, COMPUTER GRAPHICS, COMPUTERS, *DESIGN, *DISPLAY SYSTEMS, *ELECTRONIC DATA PROCESSING, GRAPHS, OPTICAL SCANNERS, *PROGRAMING, *TECHNOLOGICAL ADVANCEMENT

ABSTRACT

THE RECENT DEVELOPMENTS IN COMPUTER HARDWARE AND SOFTWARE ARE PRESENTED TO INFORM ARCHITECTS OF THIS DESIGN TOOL. TECHNICAL ADVANCEMENTS IN EQUIPMENT INCLUDE--(1) CATHODE RAY TUBE DISPLAYS, (2) LIGHT PENS, (3) PRINT-OUT AND PHOTO COPYING ATTACHMENTS, (4) CONTROLS FOR COMPARISON AND SELECTION OF IMAGES, (5) CHORDING KEYBOARDS, (6) PLOTTERS, AND (7) DIGITIZERS. COMPUTER PROGRAMING AND TOPOLOGY ARE DISCUSSED AS SOFTWARE DEVELOPMENTS IN THIS FIELD. SEVERAL PRESENT PROBLEMS AND PROJECTED FUTURE USE OF THIS DESIGN SYSTEM ARE ALSO EXPLAINED. (TC)

An Architectural Design System Based on Computer Graphics

Stephen L. MacDonald
Robert Wehrli
Department of Architecture
University of Utah

Although architects and other environmental designers followed developments in the computer field with interest, they did not become deeply involved as long as computer operations were confined largely to textual capabilities and to logical and numeric analysis. The advent of computer graphics as a potential design tool, however, is a compelling invitation for involvement. The purpose of this article is to describe some of the challenges and limitations of computer graphics, with particular emphasis upon ongoing research at the University of Utah.[1]

The Computer Science Program at the University of Utah is a division of the Department of Electrical Engineering, which in turn is part of the College of Engineering. The Computer Science Program is entrusted with education and research in computer science, as well as management of the Computer Center, which offers computer services to all non-administrative segments of the University and to the business and industrial community of Salt Lake City. Education includes general, cross-campus courses in programming and specialized coursework leading to the B.S., M.S., and Ph.D. degrees in computer science. Professor David C. Evans was named Head of the Computer Science Program in 1966.

[1] Editor's Note: Dr. Louis Schmittroth, Director of the Computer Center at the University of Utah, was a conference participant, and announced some of the developments discussed in this paper.

Supported by a large grant from Advanced Research Projects Administration (ARPA), of the Department of Defense, Evans has embarked upon a comprehensive program of research to improve the quality of computer use by improved man-machine communications. On the premise that future improvements in computer design can best be achieved when attuned to the immediate needs of ultimate users, Evans has formed four user groups to establish criteria. These user groups are: a medical diagnostics group under the direction of Dr. Homer R. Warner, of the Latter-Day Saints Hospital, Salt Lake City; a differential equations group under the direction of Dr. Louis Schmittroth; a programmed teaching group directed by Dr. Robert Barton; and an architectural group under the co-direction of Professor Stephen MacDonald and Lecturer Robert Wehrli, of the Department of Architecture.

Evans provides and directs a core research group of computer scientists to produce new equipment and generalized programs and to give consulting services to the four user groups cited above. Steven Carr, a doctoral candidate in computer science, provides liaison between the core and architectural groups, as well as consulting and programming for the architecture research group. The balance of this report will be devoted to the architectural research now in progress.

The Computer Center is served by a Univac 1108 installed in November, 1966. The graphics laboratory, located near the computer machine room, is equipped with an Industrial Displays Incorporated cathode ray tube display (CRT) and light pen, which are combined with a teletype and interfaced with a PDP8 computer used to refresh the CRT display and perform minor, routine operations with pictures displayed. An IBM card punch machine, a Univac 1004 card reader-printer, and a Univac console give the graphics laboratory direct command of the 1108 or permit operation by swap mode. Output is by print-out on hard copy, by display on the CRT, or by photographing of the display of an oscilloscope linked to the CRT. Also, a Polaroid camera is used to photograph the image on the CRT display. The display and light pen are to be used in conjunction with a flexible console which will be, in effect, an experimental laboratory, since it will permit the above mentioned peripheral devices to be plugged in in various combinations, and will for example, permit the testing of various types of tracking balls or knee and foot controls for comparison and selection. A five-fingered chording keyboard of the type invented by Englebart (reference) will be compared with a standard typewriter keyboard, and Sylvania and Rand tablets will be compared.

Other equipment available at the Computer Center is a Calcomp plotter, a Calma digitizer, and a Gerber table plotter.

In August 1967, four additional displays (Univac) will be delivered to the Computer Center. These will be assigned to the four user groups. As soon as practical, the display assigned to the Architecture group will be moved across campus to the Architecture building, to test remote operation.

The long-range objective of the architectural research is to provide a completely interactive design system such that the architectural designer can do all work for a given project from initial sketches to final working drawings and specifications.

Basic to such a design system are: a data structure sufficiently general and comprehensive that any sort of mathematical, textual, or graphic material can be stored and retrieved when needed; a programming method and compiler language that treats each subroutine as a "module" that can be altered and added or removed at any time; remote consoles that facilitate graphic manipulations of a design on-line, and with time-sharing; and a problem-oriented language suitable for designers. Although the initial emphasis is upon architecture, it is envisioned that the system will be expansible to include landscape design, interior design, furniture and fixture design, urban planning, and the like. It is hoped that the system will permit wide variations in design approaches and that it will be as simple and as natural to operate as possible.

Architectural design as traditionally taught by the project method in the studio situation is involved in great measure with what Gropius called "search," in contradistinction to the research of scientists. By search, Gropius meant an exhaustive and profound study of a proposed building in drawing and model form. By search, the designer takes vicarious experience from the drawings as a way of predicting his own responses to the constructed building, and hopefully, the responses of the building users. In this search, the designer works in cycles of rough to precise sketching (freehand to scaled drawing); general to specific; one orthographic projection to another; orthographic projection to perspective, drawing to model to mock-up; and one building system to another (architectural to mechanical to electrical to structural). While the computer will not speed the experiencing of a building scheme, it will vastly speed the preparation of the various drawings by which the scheme is experienced, and by virtue of speed, provide a great deal of information.

The potential user of an Architectural Design System will be concerned about the effect it has upon his ultimate schemes, and how this effect relates to that of traditional tools, namely tri-angle and T-square. While it is difficult to anticipate these effects with any precision, certain generalizations can be made. First, the variation in forms created with traditional tools has been great, and there is no reason to believe that the variation will be lessened by computer graphics. On the contrary, computer speed and capability to handle lines and surfaces mathematically described, as well as traditional "sketched" lines, will vastly increase the possible variation. Second, triangle and T-square tend to place on the architectural designer the limitations of linearity and flatness, so that buildings tend to become assemblages of orthographic projections, or planes, rather than truly three-dimensional sculptures. Computer capabilities should overcome this and promote more sculptural shapes, such as the hyperbolic paraboloid. Third, triangle and T-square tend to resist the free-flowing experientiality of any architectural scheme, since it is so slow and cumberson to draw all possible views of a building, whereas these are available in computer graphics by rotation of the scheme. The added capabilities, then, should enhance the designer's creativity, if only by affording him more time for creativity, since menial chores will be done quite automatically.

A picture _is_ worth a thousand words. Certainly it is worth a thousand numbers, and it is the ability to communicate in pictures, as well as in words and numbers, that makes the dramatic difference between computer graphics and the traditional numeric-logical use of the computer. For a full understanding of this, it is necessary to have a basic knowledge of the workings of a computer, with particular emphasis upon the man-machine interface.

The minimum essentials of a digital computer are: (1) an input device, (2) a "memory," (3) an arithmetic and logical unit, (4) a control mechanism, and (5) an output device (Organick, p. 3, 1966).[2]

Until recently, the most common method of input of information was by means of Hollerith cards. The cards have 80 columns and 13 rows. A single alphabetic character or number (digit) can be represented by punching an appropriate number of holes in one or more rows of a single column, allowing up to 80 alphanumeric characters per card. The input device for Hollerith cards is the card reader, which "senses" as each card is passed through it, whether a row-column location is punched or not. The information is conveyed as electric pulses to the computer memory. This memory is made up of a series of registers consisting of row upon row of magnetic cores threaded onto fine electrical wires. The pulses are stored simply as a tilting of

[2]For a well-illustrated description of computer operation, see the issue of _Life_ dated November 27, 1967.

the doughnut-shaped cores in one direction or another, depending on the direction of flow of each electric pulse in the circuitry. When tipped one way, the core represents a "one," and when tipped the opposite way, a "zero." The ones and zeros across one row, or "word" of the register represent a number in binary form. This "word," in turn, represents the alphabetic or numeric character from the Hollerith card.

The arithmetic and logical unit is capable of sensing a negative or a zero value and of addition, subtraction, multiplication, and division, and the control mechanism is able to control the sequence of events in the computer in response to coded instructions from the memory. When these have been completed for any problem, there is a conversion from the "pulse" and "no pulse" machine language to numeric and alphabetic language of the user.

It would seem that the availability of only two states of information--"pulse" or "no pulse"--would be a severe limitation. This, however, is overcome by the great number of core storage locations available, and by the great speed with which changing information can be pulsed through the computer. It should be obvious now that images in the traditional,visual sense cannot be stored in the computer, and that there must be a conversion of information from graphic to numeric to permit storage and manipulation within the computer, and then a reconversion from numeric to graphic for viewing by the user. We could conceptualize this as follows: Suppose we have a white line figure of a diamond drawn on a black square of paper. Suppose further that the sheet of paper is gridded with a mosaic consisting of a thousand squares across the format and a thousand squares from top to bottom, or a million squares for the sheet. We could then scan the sheet row after row from left to right and from top to bottom, and describe the background and the diamond on the background simply by telling for any square in the mosaic whether it is black or white. Assuming the diamond is inset from the edges of the sheet, the top few rows would be reported as "black" for every grid. When we come down as far as the top point of the diamond, we would report each square as black until the center of the sheet was reached, then report one white square for the tip of the diamond. The rest of the row would be black. By continuing to scan and report each square in the grid as black or white, we would eventually describe the diamond.

To carry our simplified description one step farther, we could let "pulse" stand for white and "no pulse" stand for black. By converting the information, then, from black or white to pulse or no pulse, the description of the white diamond on the black field can be input to the computer.

Now let us consider briefly the nature of the cathode ray tube display to see how the electronic information can be reconverted for visual use by the designer. The cathode ray tube used for display is much like a home television set in appearance. It is somewhat pear-shaped with a flat face at the large, viewing end of the "pear." The backside of this flat face is coated with phosphors which "light up" when activated by an electron gun located at the neck of the tube. The stream of electrons from this gun is aimed by a vertical and a horizontal pair of yokes which are interconnected by circuitry to the computer. Appropriate programming in the computer causes the yokes to direct the stream of electrons across the backface of the tube in a prescribed pattern.

For a very complicated picture, similar to a newspaper half-tone, we might wish to reproduce the picture on the display in much the same way as we describe our diamond. For such halftones we might wish to add to our "black" and "white" a number of degrees of light intensity from white to black. By directing the electron stream from left to right for each row, and from top to bottom of the screen for all rows, and by adjusting the intensity of the electron stream from point to point, we could reproduce a halftone-like picture. For a line drawing diagram, like our diamond, however, we can program a set of instructions in the computer which simply directs the electron stream from one point of the diamond to another, tracing it in four lines. Since the phosphors decay rapidly, it is necessary to refresh the picture in continuous cycles of electron stream tracery.

While the use of such illustrations as the above are helpful to reveal the nature of computer graphics, it is obvious that such methods as the point-by-point description of the diamond is quite laborious, and therefore unworkable for the designer. This applies also to input by Hollerith cards, even though it must be acknowledged that a great deal of productive work has been accomplished by this means. At the two extremes of the man-machine interface, then, are the man, whose natural language is the vernacular, and the machine which compensates for its simplistic language of pulse and no pulse by manipulating the pulses with phenomenal speed. What is needed is a conversational mode of interaction between man and machine, which can be accomplished by appropriate hardware and programming. And this conversation must include, now, the manipulation of graphic images as well as text and numbers. A number of concepts useful in developing such a conversational mode for architects and other designers have emerged. These concepts are of topology, notions, instances, copies, constraints, algorithms, perspective building blocks, overlays, and systems design. These concepts are related not only to the on-line display, but to the data structure programmed in the computer.[1]

Topology

The concepts as listed above reflect the nature of the digital computer and its capabilities which, as described, are quite different from the nature of the traditional architectural tools, namely T-square and triangle together with the drawings they produce. Whereas topology, the study of relationships, has not

---

[1]The concepts of notions, instances, constraints, and topology were first presented in Ivan Southerland's Sketch Pad II. Dietz presented his version of Architectural System Design at the 1966 research conference at the American Institute of Architects. The following discussion of topology was taken almost wholly from a 1967 lecture of David C. Evans.

emerged as a useful concept in making drawings on paper, it is extremely helpful in the computer. By topology the computer "knows" the relationships which maintain, for example, between the surfaces of an object. It "knows" of a box, for example, that all sides are connected pairwise to each other, that each side is connected to the top and bottom, and that these relationships do not change when the box is moved about. This is just the opposite from present architectural practice where such orthographic projections as elevations, sections, and plans of a building are laid out on the various sheets, but the drawings do not "know" what the relationships are between the various orthographic projections. It is, in fact, only by a very subtle talent that the architect or builder can make mental connections between the orthographic projections and imagine the completed building as assembled and expanded from these projections.

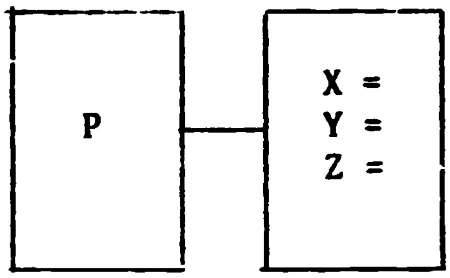
In order to understand fully the concept of topology, we must recall from the previous discussion that the computer is not a good device for storing images of objects. When an image such as a photograph or drawing is to be stored in the computer, it must be converted to a digital form in which bits of information from the photograph or drawing are input to the computer by means of punched holes on a Hollerith card, impressions on a magnetic tape, or holes in a punched tape.

The inability to store images directly turns out to have advantages which favorably offset the disadvantages. To consider this, let us begin making a simple description of objects which can at one level be stored in the computer, and at another presented on the CRT display.

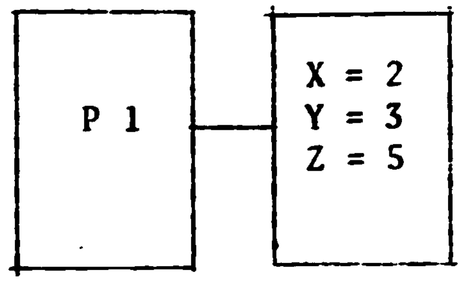
Consider first the <u>notion of a point</u>. This is the most abstract definition in which the point has no attributes save the potential attribute of an <u>x</u>, <u>y</u>, and <u>z</u> location in space. The <u>notion of a point</u> has no dimension and occupies neither one dimension (any line) nor two dimensions (any plane) nor three dimensions (any solid).

A somewhat more concrete concept is the <u>instance of a point</u>. This is a point for which <u>x</u>, <u>y</u>, and <u>z</u> values have been assigned and represented in the computer. The instance of a point, again, occupies neither area nor volume, but it contains the information to direct the electron gun at a specific location on the CRT and to light up a minute patch of phosphors. We may imagine a block, or format, for representing the notion or instance of a point. This format is simply a consistent way of listing the potential or assigned coordinates of a point.

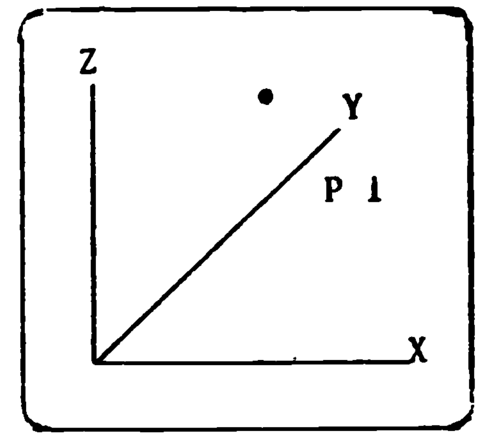
A still more concrete concept is the copy of a point. It is the copy of a point which appears on the graphics display. A copy produced from an instance of a point may be displayed in relation to the <u>x</u>, <u>y</u>, and <u>z</u> axes as indicated by its <u>x</u>, <u>y</u>, and <u>z</u> coordinates. The copy may be reproduced again and again without destruction of the instance stored in the computer as if the latter were a rubber stamp that endlessly stamps impressions without reinking.



Topological Representation of a Notion of a Point



Topological Representation of an Instance of a Point



CRT Display of a copy of a point

To continue to describe a simple object, let us now proceed to the <u>notion of a line</u>.[2] The notion of a line has only the potential attributes of connection between two notions of points. To sture any <u>instance of a line</u>, one has only to assign values to the <u>x</u>, <u>y</u>, and <u>z</u> coordinates of the two points which define the line, and to constrain these points to be linked by a line. The format for the line is such that its computer description is always stored in the same, conventional order. A line segment has the attribute of length as stated in the generalized form of the Pythagorean Theorem:

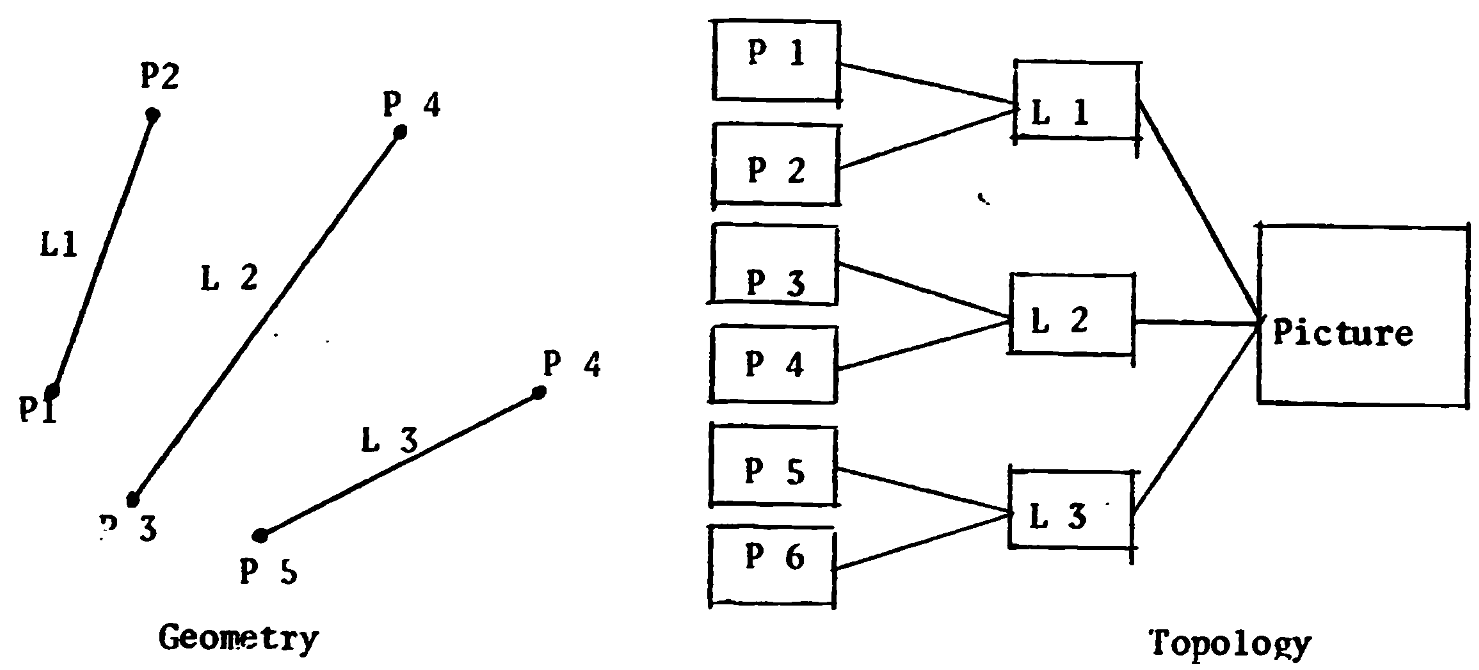
$$\text{Length} \quad L = \sqrt{(X_1 - X_2)^2 + (Y_1 + Y_2)^2 + (Z_1 - Z_2)^2}$$

A <u>copy of a line</u> is the image presented on the display.

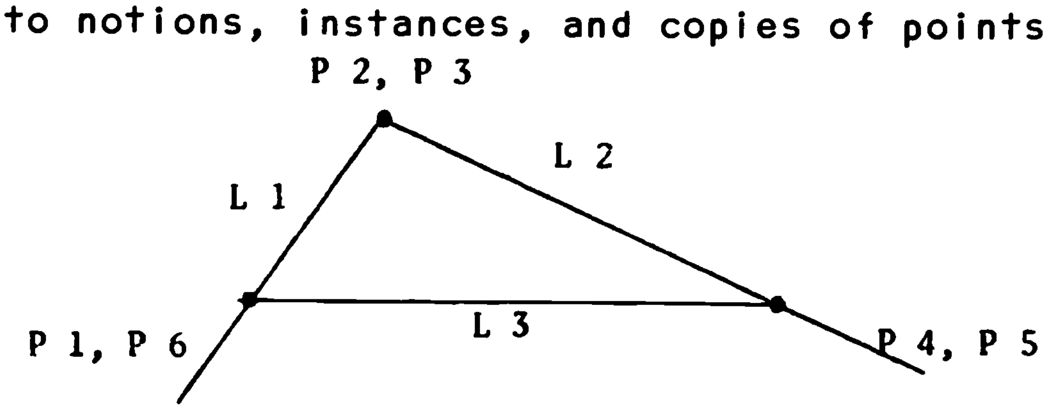
---

[2]This is precisely defined as a straight line segment, since straight lines by Euclidean axiom extend infinitely in both directions. In this context, however, "line" means "straight line segment."

A picture consists of points and lines. For example, the three lines L_1, L_2, and L_3 make a picture. If all z coordinates have the same value, the picture below would be coplanar, and the picture two-dimensional in the plane of the page. But if z coordinates have different values, the picture is three-dimensional in space. The diagram below is descriptive of the three lines. Each pair of points is linked to the line which the points define, and each line is linked to the picture. In this way, the diagram represents the relationships between points and lines, or topology, rather than their geometry.



To produce a picture of a triangle, a <u>constraint</u> is applied to the three lines such that points P_1 and P_6, P_2 and P_3, and P_4 and P_5 are pairwise coincident. The diagram below has been adjusted from the diagram above to reflect the constraint. Additional constraints could be applied to secure an equilateral triangle ($L_1 = L_2 = L_3$), or a right triangle ($L_1 = L_2$), etc. Notions, instances, and copies of pictures are parallel, or analogous to notions, instances, and copies of points and lines.



A more practical picture than the triangle for architects is the rectangle. The notion of this four-sided figure would be constrained so that opposite sides are of equal length, and adjacent sides are at right angles, and the whole figure is planar.

In addition to the advantages in the use of topologically described pictures, there is still another feature of computer graphics which makes sketching, in the sense of pen or pencil sketching, inappropriate--this is the awkwardness of the light pen, the most common tool used as a drafting tool. The top end of the light pen is connected to the CRT console with a length of coaxial cable, which inhibits free and easy movement of the pen. In addition, it is necessary to depress a lever on the pen to complete an instruction to the machine, and there is a relatively long "response" time for the machine to start and stop any line. Further, the user cannot be sure that any drawn line is exactly horizontal or vertical since T-square and triangles cannot readily be handled.

Alternatives to the light pen are the so-called tablets, such as the Rand Tablet or the Sylvania Tablet, but they also have disadvantages. The tablet is a small drafting "board" mounted horizontally in front of the CRT. Beneath the drafting surface, there are two dense sets of wires, one of which extends from top to bottom of the board and the other from side to side. The ends of the sets of wires are connected to the computer and electron gun circuitry in such a way that an exact spot on the tablet can be identified as the intersection of one of the horizontal and one of the vertical wires.

The remoteness of the tablet from the viewing surface (CRT) has been reported by users as not distracting. Because of the continuous visual feedback, it is relatively easy to draw on the tablet images that are viewed on the display.

It should be remembered, however, that both the light pen as a sketching tool and the Rand Tablet are essentially two- rather than three-dimensional devices. To overcome this one can simply use topologically constrained geometric shapes as perspective building blocks. This should be welcomed by the designer who will now be able to design for the three-dimensional world with truly three-dimensional tools.

It is anticipated that the system will work as follows: Seated at the console ready to begin a project, the designer would command an array of instances of basic geometric shapes to appear on the display. The instances would include such geometric solids as a cuboid, an hexagonal bar, an equilateral triangle bar, a cylinder, and a sphere.

Other geometric solids frequently used by a designer could be added by him to the array. The designer would select from the array those geometric shapes needed for the project at hand by touching them with the light pen (or tablet stylus). Upon a suitable typed command, unwanted geometric shapes would then vanish, and the selected subset would appear in a column at the right of the screen. A French curve, to be described later, would always appear at the bottom of this column of geometric shapes. The column constitutes a key to symbols in use as well as source for generating building blocks.

To begin a scheme for a project, the designer, by pointing the light pen at an instance of a shape, "picks up" a copy and leads it to a central position on the display. Let's say the shape is a cuboid. The instance would have no fixed dimensions. Rather, its sides would have proportions of 1:1:1. The designer could proceed in either the proportional or dimensional mode. In the proportional mode, the designer attends only to the relative dimensions of his shapes. The dimensioned mode could be entered at any time by simply assigning a value to any edge of any shape, whereupon relative lengths of all other edges are automatically assigned.

Any shapes whatever, could be constructed as a derivative of the basic shapes either by subtraction or addition. Addition takes two forms--assemblage or expansion, and subtraction by cutting or contraction. Buildings, then, become assemblages of these simple geometric shapes.