(NASA-CR-168744) THE USE OF INTERACTIVE RASTER GRAPHICS IN THE DISPLAY AND MANIPULATION OF MULTIDIMENSIONAL DATA Final Report, 1 Jan. 1979 - 31 Dec. 1980 (Purdue Univ.) 135 p HC A07/MF A01 CSCL 09B G3/61
N82-21920
Unclas
09534

Final Report

NASA - PURDUE UNIVERSITY JOINT COOPERATIVE PROGRAM IN COMPUTER GRAPHICS

(Grant No. NSG - 2192)

Grant Period: 1/1/79 - 12/31/80

Submitted to:

Mr. Thomas Gregory, Branch Chief
Aircraft Aerodynamics Branch
NASA Ames Research Center
Moffett Field, California

by

D. C. Anderson
Associate Professor of Mechanical Engineering
Purdue University

Purdue University CADLAB

By mutual agreement, the final report for NASA grant NSG-2192, "NASA - PURDUE UNIVERSITY JOINT COOPERATIVE PROGRAM IN COMPUTER GRAPHICS," consisted of the Ph.D. thesis of Dr. Sheldon Applegate, which was presented to the Aircraft Aerodynamics Branch.

This is a copy of that report, re-submitted to NASA at the request of Ms. Barbara Hastings, grants specialist.

David C Anderson

THE USE OF INTERACTIVE RASTER GRAPHICS IN THE DISPLAY AND MANIPULATION OF MULTIDIMENSIONAL DATA

A Thesis

Submitted to the Faculty

of

Purdue University

by

Sheldon Lee Applegate

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 1981

for Siblings, Inc.

ACKNOWLEDGMENTS

The author wishes to express his gratitude to a number of people, the foremost of whom is Dr. David C. Anderson. Dr. Anderson's assistance and encouragement in the course of this research were most helpful. The financial support of the National Aeronautics and Space Administration and the Purdue University School of Mechanical Engineering is greatly appreciated. Gratitude is extended to Dr. Alan T. McDonald, Dr. Raymond J. Cipra, and Dr. Henry T. Y. Yang for serving as members of his graduate committee, and to James Macqueene and James Cozzolongo for supplying data. A note of thanks also goes to Dr. Michael J. Bailey, with and from whom a great deal of computer graphics was learned. A special note of thanks goes to Dr. Richard E. Garrett of the Purdue Lacrosse Club, who first directed the author toward the field of computer aided design.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

Appendix Figure

ABSTRACT

Applegate, Sheldon Lee. Ph.D., Purdue University, August, 1981. The Use of Interactive Raster Graphics in the Display and Manipulation of Multidimensional Data. Major Professor: David C. Anderson.

Techniques for the review, display, and manipulation of multidimensional data are developed and described. Multidimensional data is meant in this context to describe scalar data associated with a three dimensional geometry or otherwise too complex to be well represented by traditional graphs. Raster graphics techniques are used to display a shaded image of a three dimensional geometry. The use of color to represent scalar data associated with the geometries in shaded images is explored. Distinct hues are associated with discrete data ranges, thus emulating the traditional representation of data with isarithms, or lines of constant numerical value. Data ranges are alternatively associated with a continuous spectrum of hues to show subtler data trends. The application of raster graphics techniques to the display of bivariate functions is explored.

An experimental data display and review tool is described, and examples of displays of several types of engineering data are shown. The ability to access raw numerical information associated with a complex data display

is demonstrated as an integral part of the data display package. The utility of interactive, analyst controlled alterations of erroneous or questionable data in the creation of meaningful displays is also demonstrated.

INTRODUCTION

The advancement of science in recent decades has changed not only mankind's view of his universe, it has changed the techniques of scientific study. A few centuries ago the "Renaissance" man contrived, often with fair success, to be well acquainted with the whole of science and the fine arts. Scientific discoveries were made by individuals performing their own experiments with hand made equipment. Much of the work was qualitative, and a great deal of effort was spent on "Will it work?" experiments testing basic principles.

The ever increasing depth of scientific study has eliminated the global expert, and the increasing cost and complexity of highly technical experimentation severely restricts those attempting individual research. The application of research teams to scientific studies has allowed more extensive experimentation, and consequently more raw information becomes quickly available. The historical scientist used his insight into physical phenomena, before or after his experiments, to reach conclusions regarding his research. Voluminous experimental results can overwhelm those wishing to interpret them, but if large quantities of information can be readily assimilated, it can give a greater understanding of the problem and enhance the opportunity for insight.

Computers and Graphics

The advent of the digital computer has given tremendous impetus to the expansion of numerical results, both in experimental and theoretical fields. Where an experimenter formerly read individual meters, digitally controlled test facilities can take over 100,000 samples per second while storing data from hundreds of instruments. Copernicus spent years making calculations to justify his theories on the organization of our relatively simple solar system; current cosmologists use a few hours of computer time just to verify when some of Copernicus' observations took place. The same computers can perform many times more extensive calculations in order to examine complex and detailed theories on the evolution of the universe, a task Copernicus could not have approached.

The automation of test control and the vast increase in computational power have allowed experimenters and researchers to study much more detailed models, but the resulting profusion of numerical information is a mixed blessing. Even if data verification and the elimination of computational errors are assumed, the task of manually interpreting results may loom as large as the former task of obtaining them. Copernicus had the advantage of knowing when his calculations placed the planets where his telescope showed them, but the voluminous, detailed results produced by contemporary theoreticians and experimentalists may give no concise indication of their overall significance. The digital computer can be used to run extensive analyses reducing results to a yes or no answer, but a pure yes or no from a complex program tells a decision maker almost nothing.

The impracticality of manually reviewing numerous data sheets has long led men to condense the available data by graphical means. Rene Descartes' use of orthogonal axes for plotting was a major step toward readily comprehendable informational display. A graphical presentation of information allows the use of man's well developed visual abilities to detect correlations. These correlations might often be overlooked if a purely mental analysis of the raw numbers were performed. Graphical displays can give an overview of a great deal of information in a concise form.

The advent of the digital computer has given an additional, direct impulse to the expanding use of graphical data presentation. The tedious chores of scaling, measuring, and plotting data can make a tabular listing of results acceptable to the person charged with making the graph, but computer automated plots eliminate the mechanical chores of plotting. The digital computer may have worsened the problem of voluminous data by automating data acquisition, but it has helped to alleviate the problem by automating graphical data presentation.

The preponderant application of computer graphics has been the presentation of simple Cartesian graphs. These graphs are familiar to those analyzing the data, but computer graphics is a powerful tool whose potential is not realized by such mundane applications. It is hardly an indictment to state that computer graphics is not always used to its potential, for the radio, the airfoil, the laser, and other inventions were used for straightforward applications before their potentials were widely explored.

The newness of computer graphics as a field of study has allowed people from a wide variety of disciplines to stumble into it and become leaders in its development. The heterogeneous backgrounds of those involved has given computer graphics research a haphazard appearance, but it has also encouraged a wide spectrum of applications without the strictures of uniform experience. Major efforts in computer graphics are being conducted by people in the fields of mechanical engineering (Purdue University), architecture (Cornell University), art (Ohio State University), civil engineering (Brigham Young University), communications (Bell Laboratories), motion pictures (Lucasfilm), computer science (University of Utah), and visualisation and simulation (Jet Propulsion Laboratories).

The visually exotic results of current research in computer graphics techniques are self-justifying to those interested in the research. Many engineers would consider them pretty, but of little practical value. The computational power used to create much of the extensive analytical data has largely been ignored in the creation of straightforward graphs, when it could have been used to create more sophisticated and more meaningful displays showing more information. The use of advanced computer graphics for informational display is becoming more a necessity than a luxury as the amount of experimental, analytical, and statistical data continues to expand. The application of more sophisticated graphics will become ever more important as the information to be analyzed becomes ever more complex. If computer graphics is to be a useful tool for the practicing engineer, this computational power must be used to create more meaningful informational displays.

A more flexible and usable means of information display may be developed using current interactive capabilities. Having the reviewer communicate with the display and the controlling computer program during execution allows the fine tuning of display options without the time and expense of repeated runs. A still more powerful tool for the review of data can be created by allowing the analyst not only to interact with the parameters controlling the display, but to operate on the data itself. An analyst reviewing data from whatever source will likely wish to know both the general trends and certain specific values of his data. Interactively controlled computer displays of data can show the trends of the data. Interactive access to numerical information allows the review of the raw numbers with which most contemporary analysts are familiar.

Interactive computer graphics can be a powerful tool in data review if it is properly applied. The hardware and software must be combined to create a data display package with three important characteristics: visual clarity, usability, and versatility. The representations of information should be lucid and, as possible, self documenting. An individual display should ideally require no external explanations of its content. The analyst must be comfortable with the display package and his interface to it so he is not substituting a tedious ritual of artificial inputs for the drudgery of plotting and interpreting pages of numerical output. The commands available to the user should stress functionality rather than formality and powerfulness. A reasonably short sequence of simple, understandable commands would for most users be preferable to a single, complex command that works by magic. The versatility necessary in a data display package would depend on the variety of its probable applications. A general purpose package intended for use as a group resource would need to be much more versatile than a package used as a subsystem in a testing facility, but both packages should be flexible in their handling of information. If the interaction can be made "comfortable" and the graphics can be made comprehensive and comprehendable, then the computer can be an invaluable tool in the understanding of available information.

Research Objectives

The primary goal of this research is the development of a tool whereby an engineer can more efficiently review large quantities of numerical information. The effectiveness of color, raster graphics in the display of complex data is examined. The user interface, the engineer's control of his display, is a primary consideration and includes both the control of the data displays produced and access to the raw numerical information used to create the displays.

CLASSES OF DATA AND THEIR DISPLAY

The irregular development of computer graphics research in general is paralleled by the irregular development of computer graphics equipment and applications. Computer generated displays of information have usually been created by persons whose primary responsibilities were in the applications and not in computer graphics. The developmental goals have generally been short range, to wit, the presentation of specific information in more or less traditional format using whatever output devices happened to be readily available. The emphasis has generally been on short development times and the fast generation of acceptable displays rather than on the creation of quality displays. Commercially available graphics software reflects this; it is typically tailored to the needs of inexpert users desiring simple, business oriented graphs and charts.

This research is primarily concerned with the display of technical data, either empirical or theoretical, and many other types of data are tacitly ignored or lightly brushed aside. This is not to imply that they are unimportant, but they are not relevant to this research. Many types of technical data are also inappropriate for the display techniques developed and are consequently ignored.

The format of a data display is dependent on the type of data being presented. Informational displays typically show the correspondence between a quantity of interest and parameters affecting it. The style of display should reflect the characteristics of the quantity of interest and the parameters of which it is a function.

Functions of a Single Variable

Functions, discrete or continuous, of a single variable have long been the primary subject of data presentation graphics. Exotic displays of piping systems or space shuttles are useful and dramatic, but the day to day utility of traditional graphs makes the Cartesian plot the most widely used form of graphical data display.

Discrete functions of a single variable, such as populations of states or births in given weeks, can be shown as Cartesian graphs, but they are often more useful if shown in some other form. Histograms are used when relative comparisons are more useful than actual numbers. The cost of living in various cities is one example of information that can be clearly presented with a histogram. Pie charts are often used for financial or budgetary displays, as they clearly show each "slice" in relation to the whole "pie." Histograms, pie charts, and other pictorial representations of discrete functions are quite popular in newspapers and textbooks. Pictorial figures give the reader a break from the text and help to keep him entertained. The maintenance of a reader's interest is a valid reason for using graphical data presentation, but it is not the only reason.

Functions of Two Variables

When the correlation between a function and more than one parameter influencing it is to be shown, simple cartesian graphs are inadequate. Kaplan[1] and Benson and Ktous[2] have demonstrated the use of computer graphics in the display of discrete, tabular data. Both used tabular forms but replaced the normal numerical entries with symbols or shading. The size or pattern of the symbols give qualitative information, and strong trends showed up as definite patterns. This technique is useful, but it relies on the tabular format of its data and is consequently restricted in its applicability.

Geographical data is a function of a two dimensional area rather than of two distinct parameters, but the general format for graphical display of geographical information is often the same as used for tabular data. Brassel[3] used the aforementioned symbols and shading to represent crime statistics on political maps of the Buffalo, New York area.

Color raster graphics has been used to display topological and topographical information. Tanimoto[4] and others have used color to verify the topology of circuit designs. McCleary[5] used color to represent oceanographic data. All parts of the Atlantic Ocean with a depth between 1000 and 2000 feet might be colored red, with other colors representing other ranges of depth. Dalton, et al.[6] have developed the Domestic Information Display System for use by the President and Congress. This system shows domestic information as a function of political boundaries within the United States. The user of this interactive system has control over how colors are used to represent the information of interest to him. He might choose to show counties with low unemployment in blue and counties with high unemployment in red.

Many functions of two variables are functions over an area or are well represented as such. For years researchers have striven to efficiently draw isarithms, or lines of constant numerical value, on sketches of two dimensional surfaces. Bengtsson and Nordbeck[7] had computerized the construction of isarithms by 1964. McLain[8] and Gold, Charters, and Ramsden[9] developed techniques creating isarithms from less regular input than Bengtsson and Nordbeck required. Dutton[10] and Fowler and Little[11] also explored alternate means of creating or displaying contours or isarithms.

Another popular means of displaying continuous functions of two variables is the creation of a topography, or a three dimensional surface showing by its altitude the value of a function at a point on its two dimensional base.[1] Fowler and Little[11], Kubert, Szabo, and Guilierie[12], and others have drawn wire frame terrain models of functions. Siddell, et al.[13] used topographies to show biomedical information. This style of display is also available in commercially marketed software graphics packages such as the DISSPLA system developed by the Integrated Software System Corporation.

Many functions of two parameters are not continuous functions of both parameters, and the parameters may not be continuously variable. The energy required to drive a pump is a function of the flow rate and the size of the pump; the latter is not normally considered to be a continuous function. Functions of two variables are sometimes shown as a family of separately labeled curves on a standard Cartesian graph. If the function of interest is f(r,s) the curves might show f(r) for several separate, constant values of s. This technique is widely used in engineering texts and reference books.

---

[1] The author implemented this technique at Lockheed Missiles and Space Company. The data displayed was from vibration tests.

Functions of Three or More Variables

The graphical techniques previously discussed are widely applicable and handle many types of data, but more complex information either presses the limits of these simple techniques or renders them all but useless. Robots used in manufacturing or other processes use five or six degrees of freedom to go through well defined movements in three dimensions. Control systems and manufacturing processes can have numerous input parameters affecting the output and product cost. Since functions of numerous variables cannot often be well represented by graphical techniques, purely numerical or analytical procedures are commonly used for their investigation. Numerical analyses can find optimal combinations of input variables or give the local effects of individual parameters, but associated graphical output typically depicts a few degenerate cases.

In certain special cases functions of more than two variables can be depicted with graphical techniques. Functions of two or three positional variables and time are often shown by videotapes or movies created on a frame-by-frame basis, with each frame corresponding to a particular instant in time. Human vision is so acutely attuned to the detection of motion and the perception of motion as a function of time that movies or other time dependent output forms are the best means of conveying complex, time dependent information.[2] Software developed with an eye toward movies is common in computer graphics. SynthaVision[14] and MOVIE.BYU[15,16,17] are two examples of software packages with movie making facilities.

Functions of three dimensional space are common but difficult to display clearly. Stress, velocity, temperature, and many other quantities occur as functions of three dimensional position, but graphical depictions of these quantities must often depend on simple cases or displays of only part of the available information. These functions are generally shown by cross sectioning the volume and using previously mentioned techniques.

One simple function of volume that has received a great deal of attention is the solid object. The function in this case has only two possible values, for either the solid object occupies a location or it does not. The representation of a solid object is of considerable import during geometric modelling, or the computer aided design of three dimensional solid objects. When the definition of an object is held only in the computer's memory, a means of accurately representing the object to the designer is necessary.

The problems in the display of a three dimensional object may be divided into two parts, the detection of the surface and the depiction of the surface. The detection of the surface, or the deciding of what part of the object goes

---

2. The author implemented time dependent displays of vibration data at Lockheed Missiles and Space Company.

where on the display, depends on the orientation of the object with respect to the viewer as well as the definition of the object. Many algorithms for the display of an object and the elimination of hidden surfaces demand the surfaces be composed of flat polygons. Newell, Newell, and Sancha[18], Watkins[19], and Warnock[20] developed three basic, if disparate, approaches to the display of surfaces composed of flat, polygonal elements. Blinn, et al.[21] further developed the scan line procedure used by Watkins to display parametrically defined surfaces.

The depiction of surfaces has been developed under the term "lighting model." The computer representation of reflectance and other surface characteristics of objects has been researched by Phong[22], Blinn[23], and Whitted[24] among others. Whitted accounted for multiple light sources, shadows, reflections, and transparency. The lighting models of Phong, Blinn, and Whitted are relatively complex. Simplified lighting models requiring less computational time are often used when representational accuracy is not a prime consideration.

The removal of hidden surfaces and the shading of surfaces can produce excellent images of solid objects, but they are not always the best representation of a geometry. The verification of three dimensional geometry definitions for structural mechanics analyses is an excellent application of computer graphics that does not need a refined lighting model. A wire frame representation of a geometry can show all parts of a three dimensional object, which hidden surface representations cannot. Feeser and Ewald[25], Siffle[26], and Potts[27] have shown various applications of computer graphics to structural mechanics. The previously mentioned MOVIE.BYU system was first developed as a structural analysis post processor.

Representations of solids by wire frame geometries can show three dimensionality, but they cannot show a fully three dimensional field of data. Rodrigues and Simon[28] have displayed three dimensional displacements at specific points in structural mechanics models by arrows. Structural displacements are also shown as exaggerated deformations of wire frames or other displays, especially when the shape of a vibrational mode is to be represented.[3]

Flow fields are another complex form of data whose representation has received considerable attention. Two dimensional and axisymetric flows have often been depicted by velocity profiles, vector fields, and two dimensional streamlines.[4] Bruch[29] demonstrated the use of interactive computer graphics in the conformal mapping area, showing streamlines of fluid flow before and after conformal

---

[3] The author used this technique at Lockheed Missiles and Space Company to display vibrational mode shapes.

[4] The author implemented a vector field display of calculated velocities in a nozzle at Lockheed Missiles and Space Company.

mappings. Three dimensional streamlines, successive cross sections of trailing vortices, and other techniques were used by Cozzolongo[30] and Tinoco et al.[31] to show flow patterns. The number of streamlines that can be meaningfully displayed is limited, for the viewer must be able to follow the streamlines and recognize their paths in relation to obstructions and other streamlines. Baden Fuller and dos Santos[32] and Nassif and Silvester[33] demonstrated techniques for representing cross sections of three dimensional flow fields. The former pair used three dimensional arrows whose length and orientation indicated the magnitude and direction of the flow at the tail of the arrow. The latter pair displayed flow with cones and cylinders instead of the arrows used by Baden Fuller and dos Santos.

Belie and Rapagnani[34] used color to show both cross sectional and fully three dimensional properties of flow fields. Two dimensional cross sections were colored according to data ranges, with specific colors associated with specific data ranges. Belie and Rapagnani also used color to provide additional magnitude information in three dimensional arrow displays of flow fields. The use of color instead of size for magnitude representation allowed the presentation of fully three dimensional fields with reasonable clarity.

The representation of three dimensional fields of data has not been thoroughly researched. Wright and

Humbrecht[35] developed techniques for displaying isovalued surfaces in three dimensions, but their representation does not allow the display of more than one such isosurface. Future research may use the transparency and highlighting facets of Whitted's lighting models in the display of multiple isosurfaces. Three dimensional, semitransparent isosurfaces may provide the three dimensional analogue to the widely used two dimensional isarithm.

Restricted Functions of Three Variables

Numerous types of engineering data, both empirical and analytical, are too complex or too voluminous to be well represented on graph paper, but are simple enough for computer generated displays to be effective. Wind tunnel test data, which may be distributed over a three dimensional surface, are but one example. Temperatures on the surface of nozzles or engine parts are another example. Finite element analyses can produce sets of data, e.g. stresses, across and through fairly complex objects. Just as in the display of three dimensional flow fields, orthogonal or cross sectioned views of geometries are sometimes used to simplify the presentation of data associated with complex shapes. An analyst could feasibly draw isarithms across a perspective sketch of an object as Chang[36] did for air pressure data, but color raster graphics provides a more powerful display medium.

Forrest[37] showed some techniques of surface rendering with color graphics, but MOVIE.BYU and the GRAPE extension of MOVIE.BYU developed by Brown[38] show the highest quality results to be found in current literature. MOVIE.BYU and GRAPE produce shaded images of three dimensional objects with stress fringes superimposed with user defined colors.

MOVIE.BYU and GRAPE demonstrate the use of color to represent engineering information across the surface of a shaded, computer generated display of an object, but they are restricted in both scope and interactivity. MOVIE.BYU was originally developed to display the results of structural mechanics analyses and is accordingly biased. The interactivity of MOVIE.BYU and GRAPE is geared toward the production of a few selected views or the definition of a sequence of views for the creation of movies. The displays produced in the course of this research are in some cases similar to those produced by CRAPE, but the crux of this research is not the creation of high quality data displays but rather the development of techniques for the manipulation of and interaction with displays of engineering data.

The techniques mentioned and references cited cover many disparate approaches to the computer aided, graphical display of data. Given the newness of computer graphics and the wide variety of data forms it is not surprising that currently developed techniques do not comprise a thorough research of the area. The possible uses of interaction and color graphics as additional tools in the display of information have barely been explored. The potential power of interactive computer graphics shows it to be an obvious solution to some of the problems of informational display, and much research will be done in the future.

HARDWARE AND SOFTWARE ENVIRONMENT

The hardware used in this research consisted of a Digital Equipment Corporation PDP 11/40 minicomputer with 90,112 (88K) words of sixteen bit memory and a Ramtek 9351 color raster cathode ray tube (CRT), with direct memory access to the PDP 11/40. The Ramtek has 262,144 (512x512) addressable picture elements (pixels). Each pixel may be any of 4096 colors, and up to 2048 different colors can be shown on the screen at any one time. This might be analogous to an artist having 4096 tubes of paint, but for any given canvas using a palette with room for 2048 colors of paint. The user must define in a video lookup table (VLT) all colors that are to be used. The video lookup table has 2048 words of 13 bit memory. The Ramtek has three four-bit digital to analog converters controlling the red, green, and blue color guns used to refresh the Ramtek screen. Each video lookup table word has four bits allocated for the control of each of the three color guns, so each VLT word defines a color that may be used on the Ramtek screen. The thirteenth bits in the video lookup table words are used to control blinking. Each pixel on the Ramtek screen is associated with eleven bits of memory that index the video lookup table. In each refresh cycle the Ramtek checks each pixel's video lookup table index, accesses the video lookup table memory, and sends the appropriate analog voltages to the red, green, and blue guns as they intensify the pixel.

UNIX[5], C, and GRAFIC

The PDP 11/40 in this research runs the UNIX operating system, and all research and support software was written in the C programming language. Although the power of the C programming language and the friendly, user-oriented environment provided by UNIX were great assets in the development of a working model for the techniques investigated, the PDP 11/40 is a small and (by today's standards) relatively weak minicomputer. The slowness of the hardware performing floating point calculations caused long delays in the processing of large data sets. These long delays would be unacceptable in a production environment, but they were unavoidable during the development of a working model.

The 11/40's sixteen-bit words allow the addressing of only 32,768 words of memory, severely limiting the size of programs that run on the 11/40. UNIX and C do not support the overlaying of programs. Since applying the techniques researched required programs that were several times larger than the addressable memory of the 11/40, UNIX system calls were used to initiate the execution of five independent main programs. The five main programs communicated via temporary

---

5. UNIX is a trademark of Bell Telephone Laboratories.

files which held the parameters necessary for whatever actions were requested. The delays caused by the rolling into memory of these five main programs and the reading of communication files would also be unacceptable in a production environment.

The graphics support software used was a subset of the GRAFIC multidevice computer graphics package developed in the Purdue University School of Mechanical Engineering Computer Aided Design and Graphics Laboratory[39]. Only those routines accessing the Ramtek 9351 were used. These routines give their user the power to draw lines or strings of characters, to define the colors in the video lookup table, to access the Ramtek's keyboard and joystick, and to fill polygonal areas of the Ramtek screen.

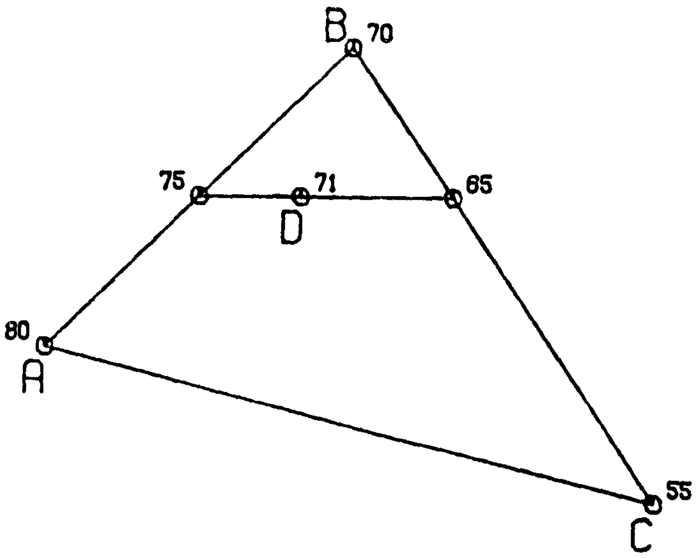


Figure 1

Polygon fill interpolations

The GRAFIC polygon fill routine "paints" the area encompassed by a sequentially ordered set of vertices. Each vertex has a screen location and an associated video lookup table index. Figure 1 shows how the GRAFIC polygon fill routine uses linear interpolations within the VLT to give an appropriate VLT index for each pixel in a polygon. Each scan line or raster is checked for intersections with the sides of the polygon. If an intersection is found, the VLT indices of the side's end points are used to interpolate a VLT index for the pixel of intersection. In Figure 1 these VLT indices are 75 and 65. A closed polygon will, with only degenerate exceptions, have an even number of intersections with any given raster, and the polygon fill routine uses a linear interpolation to calculate the VLT index for each pixel between pairs of intersections. In Figure 1 the VLT index at D was calculated to be 71. The linear interpolations can yield anomalous results for concave polygons, but the restriction of polygons to convexity was not a serious constraint in the cases considered. The standard GRAFIC polygon fill routine performs linear interpolations among the VLT indices of the vertices, and it therefore depends on the one dimensional organization of the colors defined in the VLT. GRAFIC would normally be filling a polygon in shades of, for example, blue, and the user would have a linear sequence of blues defined in the VLT. GRAFIC would interpolate within the linear sequence of blues to obtain a shade of blue for each pixel in the polygon.

Color Spaces

The Ramtek 9351 uses four-bit digital to analog converters controlling red, green and blue intensities for each pixel, but it is not necessary to define each color used in terms of red, green, and blue components. GRAFIC software allows the user to define colors in terms of red, green, and blue components; cyan, magenta, and yellow components; or hue, saturation, and intensity[40]. All three means of color definition, or color spaces, consider color to be a continuous function of three variables and capable of depiction as a volume. In GRAFIC, as elsewhere, each color component is constrained to range from zero to one.

The red, green, and blue (rgb) and cyan, magenta, and yellow (cmy) color spaces offer linear correspondences to the color guns of the Ramtek, but the hue, saturation, and intensity (hsi) color space was used because of its conceptual clarity and ease of manipulation. The hsi color space is conceptually a pair of cones joined at their bases. The top and bottom apexes of the joined cones are respectively white and black, and the lightness of a color is a function of its vertical position. A traversal of the circumference passes through all hues, so if the traversal started at red, it would pass through yellow, green, cyan, blue, and magenta before returning to red. The saturation or vivacity of the color is dependent on the radial distance from the central axis, which is a continuous spectrum of grays. A radial traversal from the central axis to the surface would begin with a neutral gray and, with constant hue and lightness, show increasing vivacity until a fully brilliant color is obtained. The brilliance is dependent on the display device, but a color on the surface of the hsi color space would be as brilliant as possible. The colors used in the course of this research were either pure grays or of maximum saturation. Since hue was used as an important cue to the user, full saturation was employed to make the distinctions among colors as clear as possible.

INTERACTIVE DATA DISPLAY

Many of the information display techniques described in the previous chapter have been used to produce elegant and dramatic displays, but attaining the highest possible output quality is not always the objective of those displaying information. When the essential task is the rapid conveyance of information, superfluous refinements of the display that interfere with utility should be eliminated. In a batch environment the additional computational time required for display refinements may have no noticeable effect, but in an interactive environment the execution time is a major concern.

A fast display algorithm is beneficial in that it reduces the necessary computer time, however in an interactive situation this is not the primary motive for rapid display presentation. If the user is to maintain his concentration and interest, there can be no inordinately long delays. Delays during interactive data display can dull the impression of previous displays and make mental comparisons more difficult. Excessive delays can also make the use of an interactive data display tool tedious or in extreme cases bore the user into applying techniques with far less potential. The tradeoff between speed of display and quality of display must be closely examined, and in many cases the speed of display must be given priority. A shoddy presentation can be useful if it is fast, but in an interactive environment the highest quality display is virtually useless for data review if it is too slow.

The display of information associated with a geometry is dependent on the display of the geometry itself. People can readily adjust to widely varied forms of data display, but their demands on representations of geometries are more stringent. Techniques for the display of geometries were therefore the first consideration in this research. The representation of associated information, while hardly an afterthought, is treated as an addition to the display of geometries.

Coordinate Transformations

The first step in displaying a geometry on a CRT is the definition of a procedure whereby a point in three dimensional space can be associated with a point on the two dimensional CRT. The relationship between the three dimensional location of a point on an object and the two dimensional location of that point on the CRT depends on the three dimensional locations of the viewer, the viewer's focal point, and the point to be represented. The vector from the viewer's position to his focal point is referred to as the viewing vector. Scaling factors and the coordinate system used to address the CRT screen must also be taken into account when deriving this relationship.

The standard technique for transforming a three dimensional location into the appropriate screen coordinates involves treating the point's position as a vector and describing any necessary rotations, translations, and scalings as matrices by which the position vector is multiplied[41,42]. The transformations used in this research consist of two translations, three rotations, and one scaling. In order to perform coordinate transformations with matrix operations the three dimensional coordinates of a point in space are formed into a vector having four components. The fourth element is set to unity. The transformation matrices are square and of order four.

Multiplying the position vector by an identity transformation matrix does not change the vector. If, however, elements of the transformation matrix are altered, the effect on the transformed vector is changed in a predictable manner. The physical significance of the change depends on which elements of the matrix are altered. Equation 1 shows that the first three elements of the fourth row control translations parallel to the X, Y, and Z axes.

$$[x\ y\ z\ 1]\begin{vmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ tx & ty & tz & 1\end{vmatrix} = [x+tx\ \ y+ty\ \ z+tz\ \ 1] \qquad (1)$$

Changes in the first three elements of the matrix's diagonal affect the vector's scaling as shown in Equation 2.

$$[x\ y\ z\ 1]\begin{vmatrix}sx & 0 & 0 & 0\\ 0 & sy & 0 & 0\\ 0 & 0 & sz & 0\\ 0 & 0 & 0 & 1\end{vmatrix} = [(x)sx\ \ (y)sy\ \ (z)sz\ \ 1] \qquad (2)$$

The matrices that produce rotations of angle theta about the X, Y, and Z axes of a right-handed coordinate system are given by Equations 3, 4, and 5 respectively.

$$[R_x] = \begin{vmatrix}1 & 0 & 0 & 0\\ 0 & \cos(\theta) & \sin(\theta) & 0\\ 0 & -\sin(\theta) & \cos(\theta) & 0\\ 0 & 0 & 0 & 1\end{vmatrix} \qquad (3)$$

$$[R_y] = \begin{vmatrix}\cos(\theta) & 0 & -\sin(\theta) & 0\\ 0 & 1 & 0 & 0\\ \sin(\theta) & 0 & \cos(\theta) & 0\\ 0 & 0 & 0 & 1\end{vmatrix} \qquad (4)$$

$$[R_z] = \begin{vmatrix}\cos(\theta) & \sin(\theta) & 0 & 0\\ -\sin(\theta) & \cos(\theta) & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{vmatrix} \qquad (5)$$

The popularity of this notation stems from both the clarity of representation and the ease with which numerous transformations may be combined. Matrix multiplication is associative, so consecutive transformation matrices may be multiplied to give a single, combined transformation matrix. Multiplying a position vector by the combined transformation matrix yields the same vector that would be produced by sequentially multiplying the position vector by each individual transformation matrix. If a complex transformation is to be applied to a set of position vectors, most of the calculations can be eliminated by multiplicatively combining the component transformation matrices.

The scaling transformation in Equation 2 scales about the origin, and the rotations defined in Equations 3, 4, and 5 are rotations about the principal coordinate axes. Rotations about arbitrary axes and scalings about arbitrary points require translations before and after the essential transformation. In order to scale an object about some point A, the object must be translated so the point on the object corresponding to A is moved to the origin. The scaling may then be performed as per Equation 2. The object must then be translated back to its original position. Rotations about arbitrary axes can be more involved, but translations are still a necessary part of the transformation.

The viewing transformation used in this research consists of: a translation of the geometry so the viewer's focal point coincides with the origin, rotations about the X, Y, and Z axes to align the viewing vector with the display device's screen coordinate system, a scaling to convert the geometry's coordinates into screen coordinates, and a translation to center the image of the geometry in the allocated portion of the Ramtek screen. The six elements of the viewing transformation used are not combined into a single transformation. Breaks on both sides of the rotation transforms are maintained, partly to allow the optional inclusion of a perspective transformation.

Raster Considerations

A raster display device such as the Ramtek 9351 gives the user the power to draw complete images in addition to wire frame outlines. For many applications, especially in business data display, the use of a few solid colors to enhance simple displays is adequate, and raster displays showing perhaps eight colors are common. The Ramtek 9351's 2048 color palette gives the user a much better selection of colors, which is necessary for sophisticated displays. The four-bit digital to analog converters controlling the Ramtek's color guns allow thirty-one intensities of any particular hue, and these intensity spectra are used to create shaded images.

The realistic depiction of scenes requires an elaborate lighting model such as those developed by Phong[22], Blinn[23], and Whitted[24]. Cruder lighting models can produce depictions of three dimensional geometries, but highlighting or reflectance anomalies are noticeable. The display of a geometry and some associated data requires only that the geometry be recognizable, so a simple lighting model can be used. The calculations required to depict a scene with multiple light sources, shadows, variable surface characteristics, reflections, transparency, and other enhancements can be bypassed in the interests of time.

The lighting model used in this research accounts for a single point light source and a certain amount of ambient light. There is a limit on the reflectance of the surfaces displayed and no shadowing is performed. The lightnesses and hues are calculated at the vertices of polygonal screen areas, and the hues and lightnesses for the interior of the polygon are determined by linear interpolations. This is similar to the technique used by MOVIE.BYU.

The lightness, or intensity, is calculated to fall into the integer range from one to thirty-one and thus match the hardware characteristics of the Ramtek 9351. The ambient light level and reflectance limitations are compile-time constants and further limited the range of allowable lightnesses. A typical lightness range is six to twenty-four. Experience with the Ramtek hardware showed that different hues were indistinguishable if both had very low or very high intensities. The lightness limits, six and twenty-four, have significance only in relation to the thirty-one levels of intensity available on the Ramtek 9351. They do suggest, however, that only the central sixty percent of a device's intensity range is apt to be useful in the color cuing of information.

The lightness value for a polygonal vertex is calculated using Equation 6, where L is the lightness level being determined, A is the ambient light level, R is the maximum allowable lightness level, and alpha is the angle in three dimensional space between the vector from the vertex in question to the light source and a vector normal to the polygon being displayed.

$$L = A + (R-A)\cos^2(\alpha) \qquad (6)$$

The power to which the cosine function is raised affects the highlighting of the display. The typical range for this exponent is from one to ten, and two and three are the most common values. An exponent of two was chosen primarily because the square of the cosine of alpha is easy to calculate using vector algebra and does not require the extraction of any square roots. The lighting model produces acceptable results and reasonable highlights. Although the Ramtek hardware made a lightness range of six to twenty-four convenient, Equation 6 or similar equations can be used for any desired range of lightnesses.

Hidden Surface Algorithms

The display of geometries must generally account for the obscuring of one surface by another, and there are numerous approaches to the problem of hidden surface elimination[43]. Warnock[20] proposed an area by area search of the screen. Watkins[19] and others have developed raster or scan line techniques that examine the geometry to find the parts that are visible on each of the display device's rasters. Whitted[24] and SynthaVision[14] mathematically fire rays from the viewer's position toward each pixel's back-transformed position in three space and calculate which portion of the geometry is closest to the viewer along each ray.

This research uses perhaps the simplest effective hidden surface removal algorithm, that developed by Newell, Newell, and Sancha[18]. The Newell, Newell, and Sancha technique is based on depth prioritization of the elements of the geometry. By displaying the elements in an order determined by their distance from the viewer, the elements further from the viewer will be obscured or painted over by those elements of the geometry closer to the viewer. This technique requires a complete sort of the elements and maintenance of the sorted list, which for moderately sized geometries is not an inordinate amount of overhead. A simplified form of the Franta-Maly[43] bilevel data structure was used to lessen the time required to insert an element into the sorted list. Although there are cases when this algorithm can be fooled into an incorrect display ordering, these cases seldom occur in simple geometries, and the effect of their occurrences is minor. The data display package developed as part of this research takes approximately three minutes to display a geometry with 100 triangular elements.

Description of Geometries

The described transformation of a geometry from traditional three dimensional space to screen coordinates is based on the successive transformations of individual points. Most definitions of solid geometries used for computer graphics consist of points in three dimensional space and various connecting surfaces. When additional information is to be displayed with the geometry, the points of the geometry definition may readily be used as hooks on which to hang data values. In this research the points of the geometry definition were assigned data values according to a data file being represented with the geometry, and the terms geometry point and data point are virtually synonymous. The definition of the geometry and the set of data to be displayed with the geometry are stored separately, so several sets of data can be viewed without rereading the geometry definition.

Many means of describing surfaces between points in space are currently used. Flat plates, bicubic parametric patches, Bezier patches, B-spline patches, Overhauser patches, and others are used in various applications[41]. When geometries with complex patches are displayed on raster devices, calculations are often performed for every pixel on the screen. The alternative is to divide and subdivide each patch until the remaining portion of the patch is nearly flat or until the limit of the device's resolution is reached. The quality of the resultant display is thus dependent on device limits or on tolerance limits that are set by the user. The research described here uses flat, triangular plates as surface elements, thus gaining speed and ease of display, but losing the ability to represent curvature.

The geometry file format used in this research is intended to be simple and easy to use. It has four lines of titling information and a group of five counters at the beginning of the file. All points used in the geometry are then defined in a list of floating point triples representing locations in three dimensional space. The plate elements making up the surface of the geometry are defined by sets of indices to the points at their vertices. For example, the fourth plate might be described as the triangular area defined by points one, twelve, and eleven. Quadrilateral plates may be a part of the geometry description, but when the geometry file is read, they are broken into pairs of triangular elements. The points describing each plate are ordered (counterclockwise) to give a definite spatial orientation to the plate.

The research implementation of the Newell, Newell, and Sancha hidden surface algorithm sorts the elements into a display sequence. As each element's turn to be displayed comes, the vertices are transformed from three dimensional coordinates to screen locations. The GRAFIC support software is then called on to fill the polygonal area defined by the screen locations of the vertices. If it is assumed that the geometry described is a solid object, the spatial orientation of the plates can be used to speed the display of the geometry. If the geometry to be displayed is solid, then any plate facing away from the viewer will be on the reverse side and will be obscured by the obverse plates. The display time can be cut, perhaps in half, by not displaying plates with reverse orientation.

Displaying Data with a Geometry

The lighting model used in the research is independent of the colors used, so data associated with portions of the displayed geometry can be mapped into colors without prejudicing the shading of the geometry. Two principal means of associating color with data were examined, continuous tone and isarithmic.

The continuous tone technique, which is used by MOVIE.BYU[15,16,17] and GRAPE[38], associates the data range of interest with a continuous spectrum of colors, perhaps from blue through green to red. The viewer can see fairly subtle data trends by noticing the patterns of the hues. The data range represented by a spectrum is determined solely by the data levels corresponding to the two extremes of the spectrum, so if a restricted data range is of particular interest, it can easily be accentuated with a continuous tone display. Figures 2 and 3 show theoretical temperatures in metal during a welding process.

The spectrum in Figure 2 covers the full range of temperatures encountered. Figure 3 shows the same data with the continuous tone spectrum limited to the temperature range where melting is occurring, from 1500°K to 1800°K. (Figures 2 and 3 show no evidence of shading because they are orthogonal views of a flat surface with a distant light source.)

The isarithmic color-to-data correspondence scheme emulates the graphical presentation of isarithms, or lines of constant numerical value, by assigning a specific hue to the data range between two isarithms. Although this display technique does not bring out subtle data trends as well as the continuous tone technique, it does give the analyst definite bounds on the datum at any particular point. About eight easily distinguishable colors can be extracted from a normal range of hues, thus nominally limiting the isarithms that can be displayed to seven. This restriction can be circumvented by displaying more than one isarithmic band in the same color. Although a particular display may be limited to eight or so colors, some fast and easy manipulation of the VLT allows the accentuation of many more isarithms.

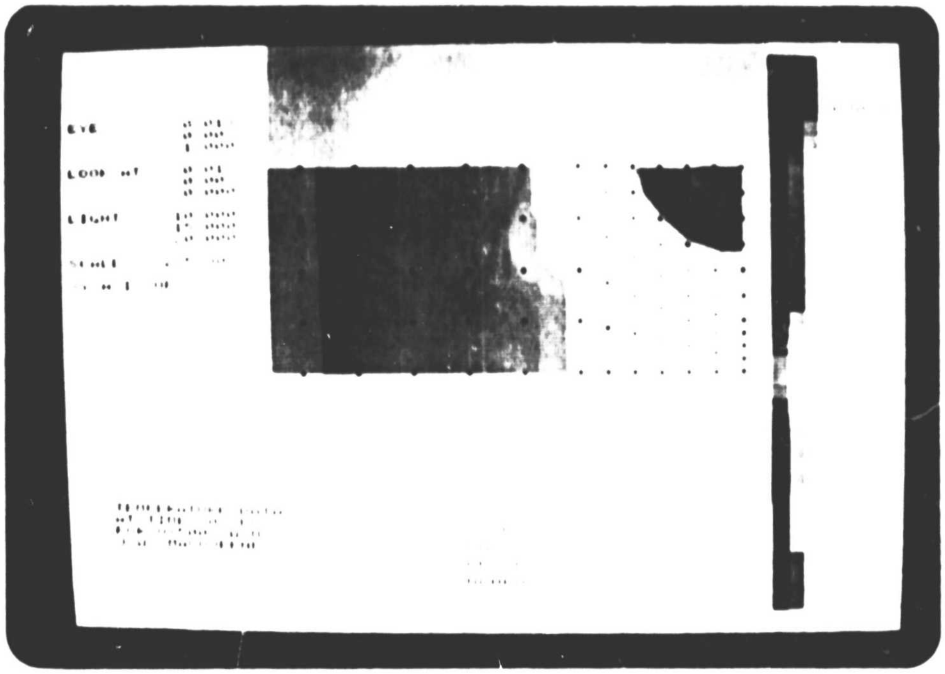


Figure 2

Full Range Continuous Tone Display

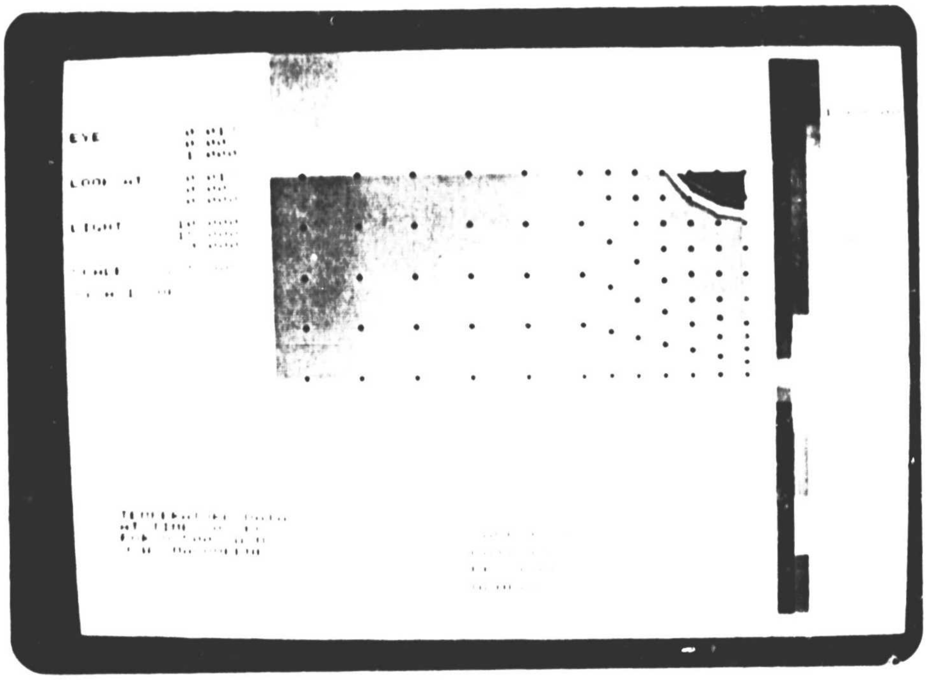


Figure 3

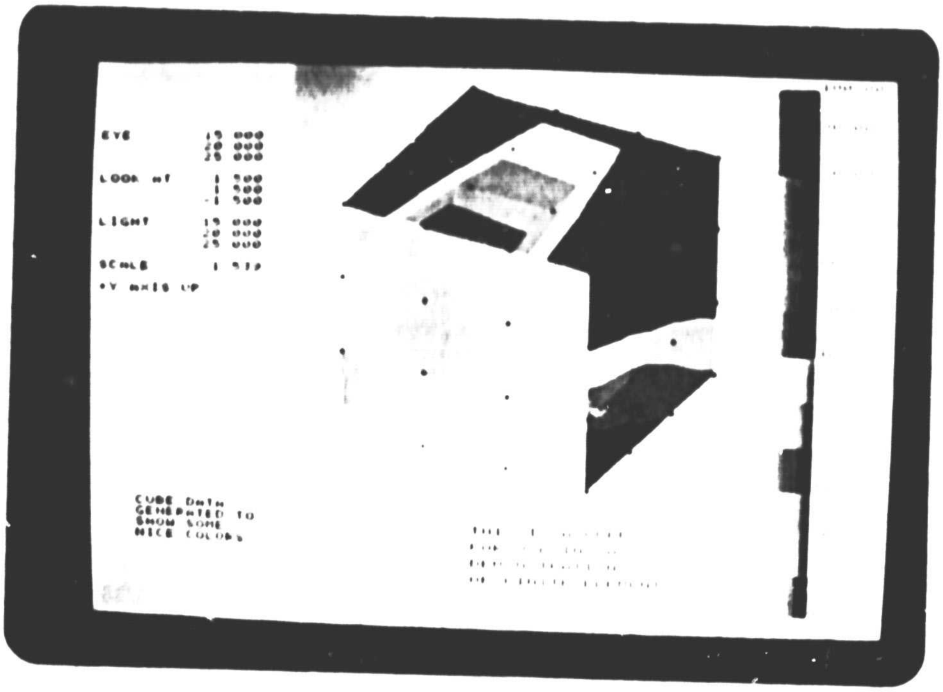
Restricted Range Continuous Tone Display

Figures 4 and 5 show the same data with different color-to-data range correspondences. The change from Figure 4 to Figure 5 required only alteration of the VLT and not a full redrawing of the geometry. The interactive alteration of the VLT takes a few seconds as opposed to the minutes required for a full redraw of a geometry.

Color-to-data level correspondences for both the isarithmic and continuous tone displays are user controlled, although automatically determined correspondences are available. The user control extends to the definition of all colors in an isarithmic display and the specification of the hues at both ends of the continuous tone spectrum. The hues representing data above and below the continuous tone spectrum limits are also user controlled.

The continuous tone and isarithmic displays use differently organized sets of colors in the VLT. The colors in a VLT set up for a continuous tone display are organized as a series of hue spectra, each with a distinct intensity. The isarithmic display VLT has a series of monochromatic intensity spectra. If the lighting model is slightly degraded, an isarithmic display could be created using the continuous tone VLT organization, but the converse procedure would be difficult.

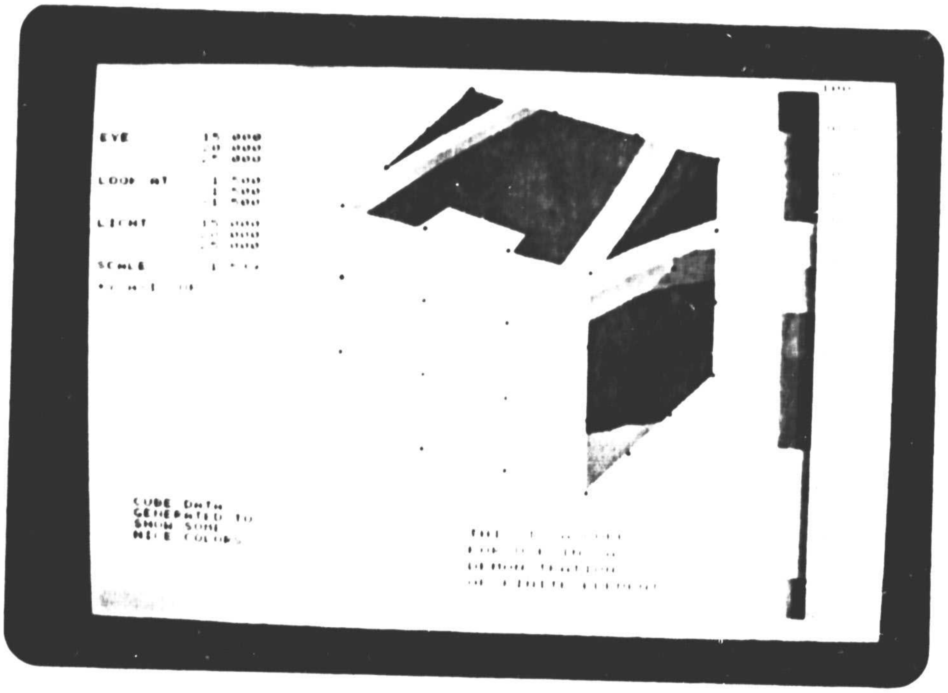
Both the continuous tone and isarithmic displays, as used in this research, rely on linear interpolations across



```
EYE        15.000
           20.000
           25.000
LOOK AT     1.500
            1.500
           -1.500
LIGHT      15.000
           20.000
           25.000
SCALE       1.519
+Y AXIS UP

  CUBE DATA
  GENERATED TO
  SHOW SOME
  NICE COLORS
```

Figure 4

Low Emphasis Isarithmic Display



```
EYE        15.000
           20.000
           25.000
LOOK AT     1.500
            1.500
            1.500
LIGHT      15.000
           20.000
           25.000
SCALE

  CUBE DATA
  GENERATED TO
  SHOW SOME
  NICE COLORS
```

Figure 5

High Emphasis Isarithmic Display

the area of each displayed element. To display an element the continuous tone algorithm calculates the intensity based on the lighting model. The calculated intensity determines which constant-intensity spectrum in the VLT is used to depict the element. Within the determined spectrum the VLT locations (and thus the hues) corresponding to the data values at each vertex of the element are then calculated. The two dimensional, linear interpolation described in the previous chapter is then used to determine a hue for each pixel in the area covered by the element.

The isarithmic display algorithm must do more calculations before calling on an area filling routine. Each element must be divided into the areas covered by each isarithmic band. This determines the hue for the screen area corresponding to each portion of the element. The subelemental area can be filled with a constant intensity, thus saving some time, or the intensities at each corner of the subelemental area can be calculated and the support software called on to do monochromatic shading. The use of constant intensities across all elements would, for flat elements, correspond to an infinitely distant light source. There is little degradation of display quality when using a constant intensity across each element, although this is to a certain extent a function of the geometry displayed.

Interactive Control of a Geometry Display

The real power of interactive data display lies in the immediate feedback it can give the user. To make full use of interactivity, immediacy of response should be maintained while flexible control of display options is granted the user. A lenient attitude toward erroneous input should be inherent in a data display package as all users, especially those unfamiliar with the package, will occasionally make an incorrect entry. Default values for various options should be available, thus allowing the casual user to obtain an acceptable display without fully learning the software or using a set of "magic" inputs.

The refinement of the viewing parameters should not be unnecessarily slowed by the presentation of a fully developed display. The use of wire frames or other rapidly generated views for the refinement of viewing parameters is a practical necessity. This necessity may be obviated by the introduction of interpolative hardware area fill or other capabilities on forthcoming raster devices. Contemporary area filling hardware uses a single color or pattern for whatever area is being filled. The option of aborting an inappropriate display is also desirable, especially when the geometry is complex and takes considerable time to display. Another alternative explored during this research is the switching of display modes during the display of a geometry. Figure 6 shows the result of changing from a display of the geometry with the associated data to a wire frame display and shortly thereafter aborting the display process.

The simplest means of determining viewing parameters, allowing the user to input the three dimensional coordinates of a viewpoint or eye position and a focal point, is effective and was used in this research, but it is not the ideal means of view control. On vector devices it is possible to draw a geometry from an arbitrary position and allow the user to interactively rotate the geometry until a desirable view is obtained, but this is not now practical on full color raster devices. A raster display may consist of over one million pixels, and the sheer number of pixel colors to

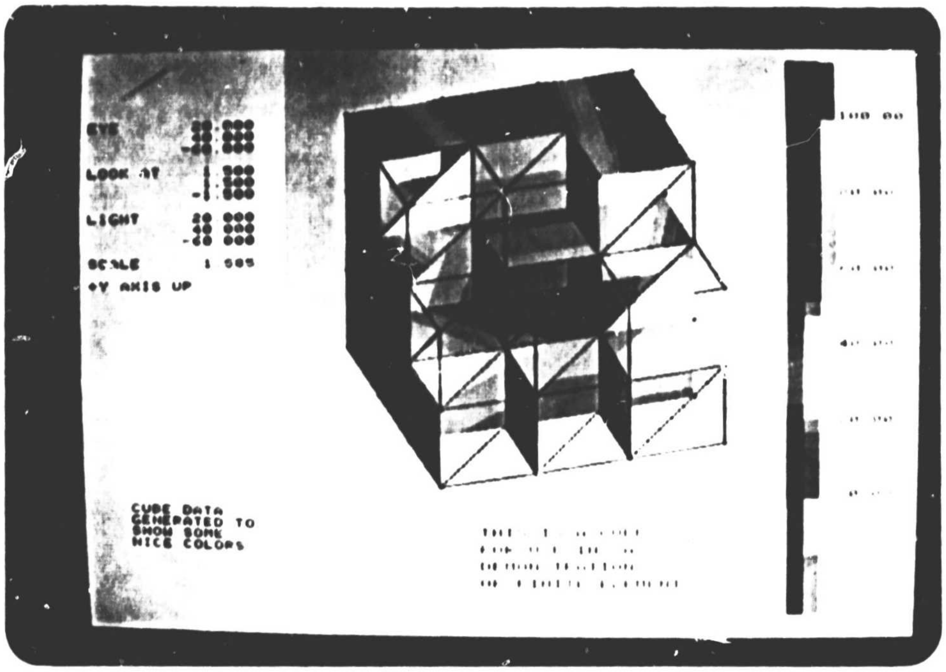


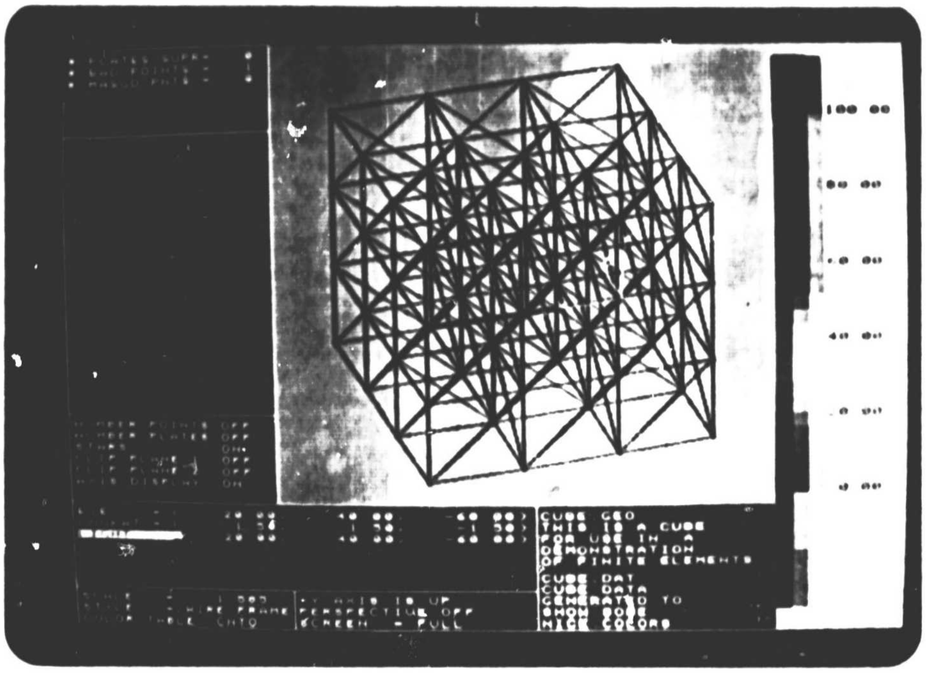
Figure 6

An Aborted Display

be determined makes user-controlled, continuous rotations impractical. This constraint on rotations will probably be eliminated by the advent of reasonably inexpensive hardware for the elimination of hidden surfaces. Contemporary hidden surface hardware is quite costly, and raster displays of complex objects will remain essentially static until the cost of hidden surface hardware comes down. The best available situation is probably the tandem application of vector and raster devices. A vector device with hardware rotation, translation, and scaling would be used to determine the viewing parameters, which would then be used in the creation of a raster display.

The positional control of light sources should also be under user control. Subtleties in the geometry may require a specific lighting direction for the user to easily recognize features. Figure 7 shows the input of viewing and lighting information.

The user should also have control over several subsidiary display parameters. The scaling of the geometry in its transformation to screen coordinates should be user controlled, although the ability to automatically calculate an appropriate scaling factor for the current view is an excellent extension of the scaling control. The wing in Figure 8 was automatically scaled so that its width, being greater than its height, crossed 90% of the viewing area. The use of perspective is an option that can be helpful if the user



AXIS DISPLAY ON

CUBE GEO
THIS IS A CUBE
FOR USE IN A
DEMONSTRATION
OF FINITE ELEMENTS

STYLE = WIRE FRAME

+Y AXIS IS UP
PERSPECTIVE OFF
SCREEN = FULL

CUBE DAT
CUBE DATA
GENERATED TO
SHOW GOOD
NICE COLORS

Figure 7

Setting the Lighting and Viewing Parameters

is accustomed to perspective representations. The choice of which axis is to be represented as vertical is also given to the user. In the aerospace industry it is common the have the positive Z axis running from the nose of a missile toward the exhaust nozzles, but in other applications the positive Y or positive Z axes rise from ground origins. The user specification of the vertical axis is a practical necessity for a widely used data display package. In Figure 8 the negative X axis was chosen as vertical.

Focusing the Display on a Section of Interest

Numerous means can be given the user for focusing the display on geometry sections of particular interest. The

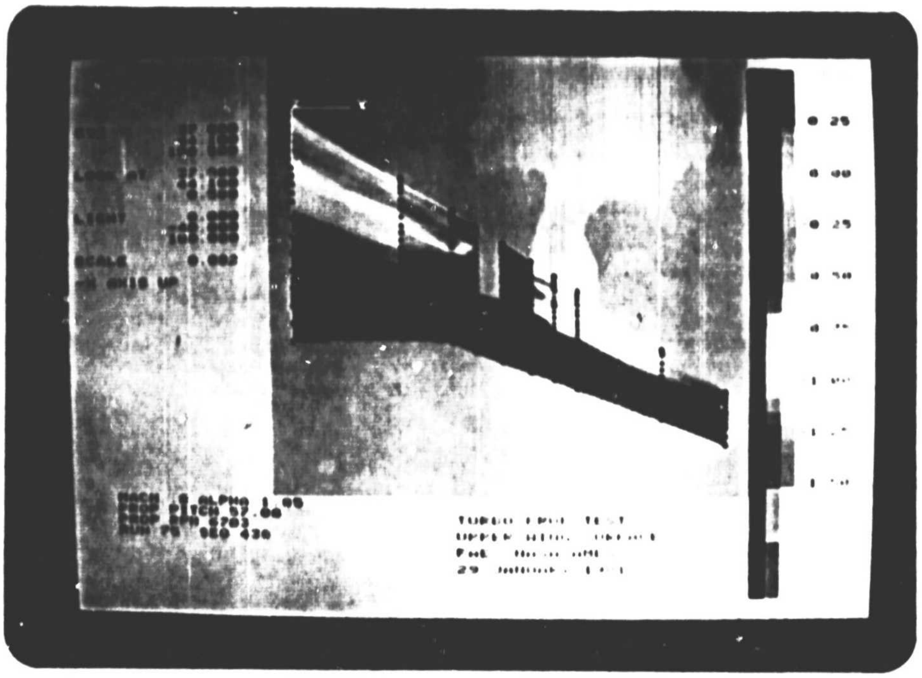


Figure 8

Automatically Scaled Display

previously mentioned choice of a focal point defines which part of the geometry is to be centered in the display area. The scaling factor can be used to control how much of the geometry is displayed on the screen. The user could manipulate the focal point and scaling factor to concentrate on a particular section, but it is hard to obtain an intuitive feel for how far to shift the focal point in space and how much to change the scale factor. Ten percent zooming functions were installed as a means of controlling the scale factor, but the display time, being a matter of minutes rather than seconds, made this an intolerably slow means of scale control. One straightforward approach would have the user pick an area of a previously presented display and request that the chosen area be enlarged to fill the entire display area.

The redisplay of a detail of an existent display is, unfortunately, much easier to describe functionally than mathematically. A new scale factor can easily be calculated from the previous scale factor and the size of the area of interest on the previous display, but the determination of a new focal point requires the definition of a three dimensional position from two dimensional information. The inversion of the viewing transformation is required to determine a point in the geometry coordinates from a screen location.

The reversal of the scaling and translation transformations performed on the geometry in the display process is easily accomplished, but reversing the rotations is more involved. The reversal of the rotations applied to the geometry would nominally require either the re-creation of the rotation transformation matrices or the inversion of the combined rotation transformation matrix, but if the translation and scaling transformation matrices are kept separately from the rotation transformation matrix, these processes can be avoided. The rotation matrices are orthogonal, i.e. their inverses are equal to their transposes. This property holds for multiplicative combinations of these matrices. The reversed rotations can be applied to a point by multiplying the point's position vector by the transpose of the combined rotation transformation matrix. The translation and rotation transformation matrices are not orthogonal, and if the described rotation reversal procedure is to be used, the rotation transformations must be kept separately.

The extraction of three dimensional information from a two dimensional source requires the specification of an additional constraint. The reversing, by matrix multiplication, of the display transformation requires a 1x4 vector in the screen coordinate system. The first two entries are the screen (X and Y) coordinates of the new focal point. The fourth entry is unity by notational convention. It is convenient to set the third element, the Z position in the screen coordinate system, to zero. The resultant focal point in the geometry coordinate system will be in a plane passing through the old focal point and perpendicular to the viewing vector. When changing the focal point in this manner, it is appropriate to translate the eye position by the same amount that the focal point was translated. The new view of the geometry is from the same direction, but from a different point in space. A detail of Figure 8 is shown in Figure 9.

A new display of a geometry with a new focal point and new eye position takes as long to produce as the original display, but faster, if cruder, means of examining a portion of a display are available. It is possible to "blow up" a portion of a display by pixel replication, or the pixel by pixel expansion of the area of interest. Pixel replication gives a coarser display with increased "jaggies," or

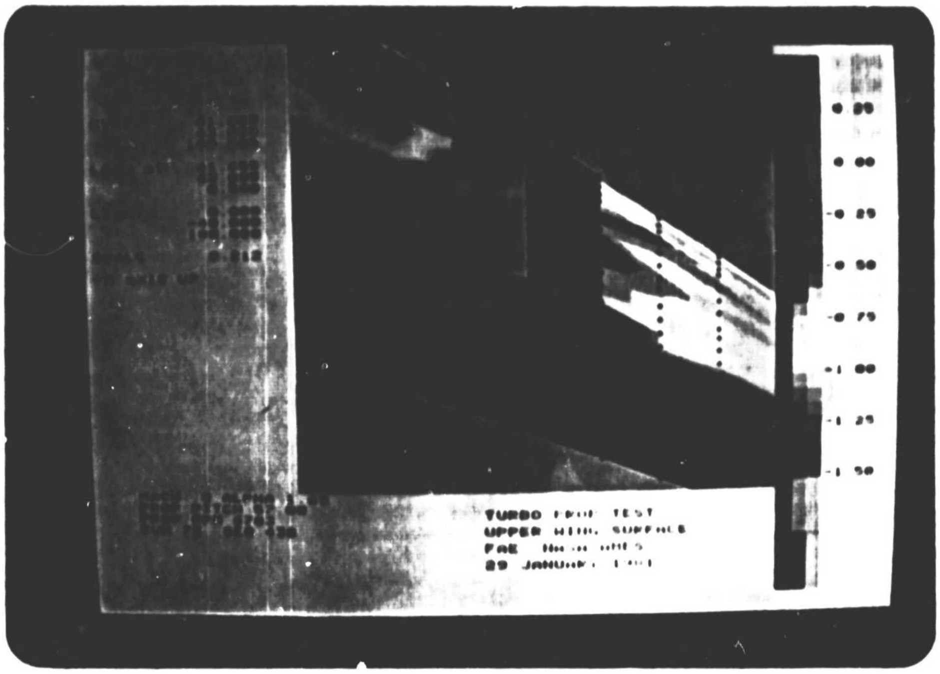


Figure 9

A Detail of Figure 8

stairstepped edges. The severity of the stairstepping is linearly dependent on the extent of the replication. It should be noted that the Ramtek 9351's 512x512 resolution is mediocre by current standards, and the severity of the jaggies caused by pixel replication is accordingly more severe on the 9351 than on raster devices with higher resolution. A combination of hardware and software difficulties kept pixel replication from being a useful tool during this research. Newer raster display devices have pixel replication as a hardware feature, so efforts at software emulation of this hardware feature were curtailed.

Clipping Planes

The display of complex geometries may require the manipulation of many more elements or plates than are of immediate interest, and interior portions of a solid geometry are in general obscured by the exterior. In both cases an instrument for the removal of unwanted portions of the geometry could increase the speed and utility of the data display package. The use of clipping planes, defined in the geometry's coordinate system, can be a powerful tool. Clipping planes could be described as the path of a knife as it cuts off portions of a geometry. By eliminating unwanted plates from consideration early in the display process, much time can be saved. Clipping planes can be used to give cross sections or other interior views of solid geometries.

One implementation strategy is hither-yon or Z clipping. This would entail the specification of two distances in the geometry coordinate system, Z_h or Z_{hither}, and Z_y or Z_{yon}. The clipping planes so specified are perpendicular to the viewing vector and at distances Z_h and Z_y from the viewing position. Any point in front of the Z_h clipping plane or beyond the Z_y clipping plane is eliminated. Most users can readily comprehend the concept of "anything too close to the viewer or too far from him is clipped."

A point's position relative to the viewer's position can be construed as a vector with components parallel to the viewing vector and perpendicular to the viewing vector. If d_i is the magnitude of the component parallel to the viewing vector of the vector from the viewpoint to point P_i, then P_i would be clipped if $d_i < Z_h$ or $d_i > Z_y$. In Figure 10 the viewing vector is in the plane of the paper. P_1 would be clipped as too close to the viewing position and P_2 would be clipped as too far from the viewing position. The calculations are simple if performed after the first translation and the rotation transformations but before the scaling and second translation in the viewing transformation.

A more flexible implementation of clipping planes would allow arbitrarily located and oriented clipping planes. The clipping plane definition used in this research consists of

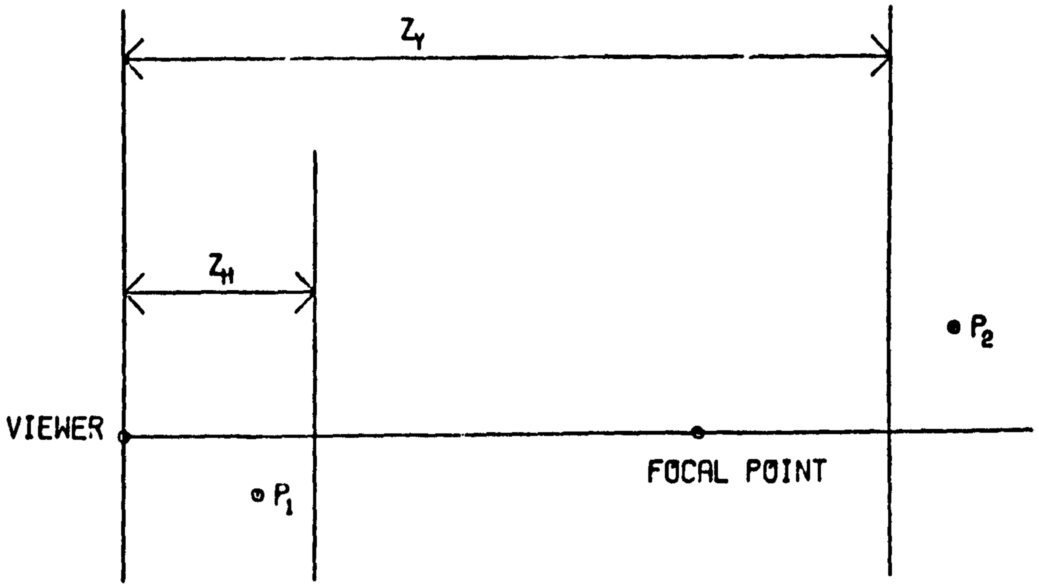


Figure 10

Hither and Yon Clipping

a point through which the clipping plane passes and a direction vector. The clipping plane is oriented perpendicular to the direction vector, and any point on the side of the clipping plane indicated by the direction vector is clipped. In practice the point is clipped if the dot product of the clipping plane's direction vector and the vector from the plane location point to the point in question is greater than zero. If this dot product is greater than zero, the angle between the normal vector and the vector from the plane definition point to the point in question is less than $\pi/2$ radians, indicating that the point is on the same side of the clipping plane as the plane's normal vector.

It is practical to use more than one clipping plane. Interior slices of solid geometries can then be displayed in minimal time. Quarter sections also become available. Figure 11 shows a slice of the geometry shown in Figure 6.

The discussion of clipping plane usage has been limited to points of the geometry and has ignored the clipping of elements. It is mathematically possible to clip any plate exactly where the clipping plane passes through it. It is also possible to create new portions of the geometry lying in the clipping plane and thus to display the geometry exactly as clipped. If data values are to be displayed with the geometry, much interpolation would be involved in generating values on the newly clipped face. In an interactive environment the calculations required for exact clipping are

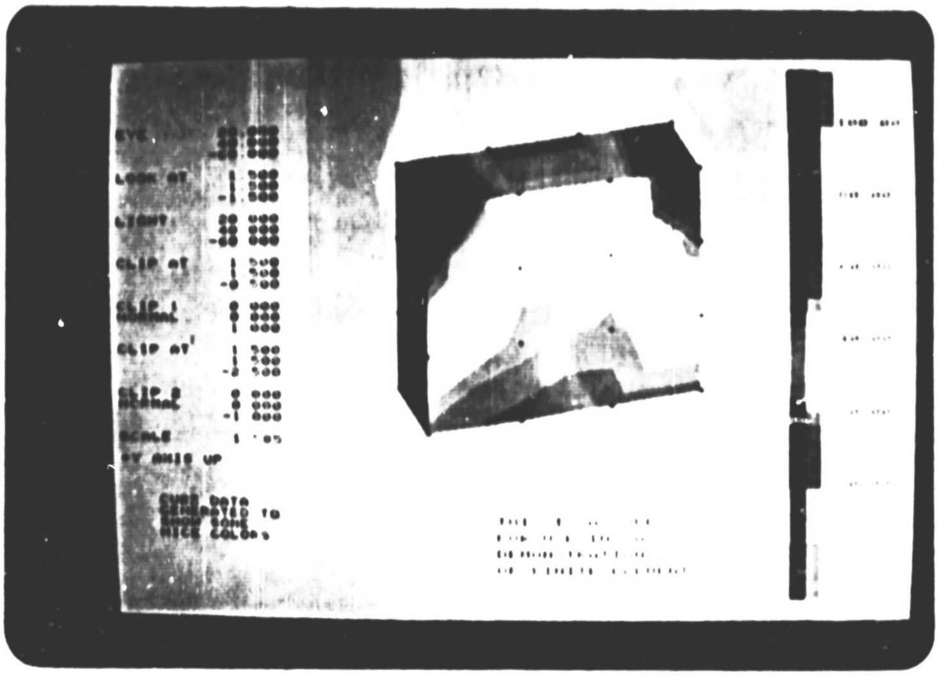


Figure 11

Clipping Planes

apt to cause inordinate delays in display presentation. Eliminating all of any plates that even partially cross a clipping plane is faster and, for viewing interior nodal data, more easily used.

User Orientation

A global view of a distinctive geometry leaves no doubt in the viewer's mind about spatial orientation, but a detail or a close up view of portions of the same geometry can give the same user no clues as to what he is seeing. Viewer orientation aids are desirable, especially in details or other views of unclear orientation. One aid would be the display of three dimensional coordinate axes, either at the viewer's focal point or at the origin of the geometry's coordinate system.

The marking of the geometry points on the display and the numbering of the elements or points of the geometry can also be helpful. An analyst reviewing data should normally have access to hard copies of the geometry definition. The on-screen display of node numbers, as in Figure 12, can be correlated with the geometry definition to find exactly which part of the geometry is being shown. Point or element numbering on a global view of a geometry is less apt to be useful. If many points are displayed, the labeling can become confused and less useful than when fewer points are labeled.

Marking the screen location of the geometry points can be helpful both in orienting the viewer and in

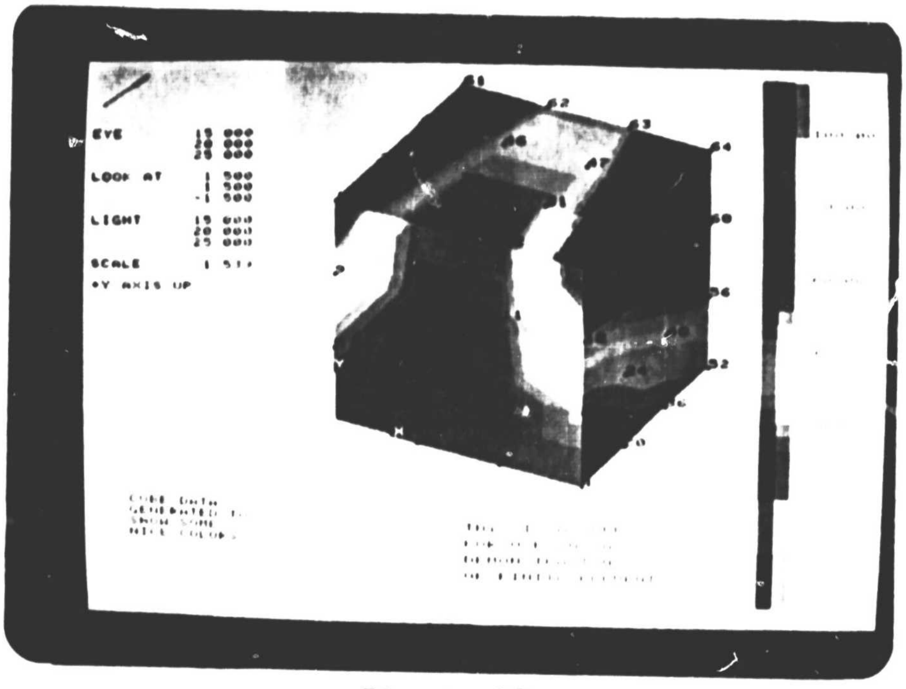


Figure 12

Node Numbers and Axes

distinguishing data trends from data anomalies. If used with node numbers, stars or other marks at geometry points can give the viewer solid reference points. If several data points show similar results, it is apt to be a local trend, but if a single datum is markedly different from the surrounding data, the single datum might be questioned. The marking of the data points can show whether a local trend is caused by one or several data values. Further discussion of anomalous data and the examination of local effects follows in the next chapter.

Screen Accoutrements

If an engineer were asked how large a display he wants, his initial response might easily be "As large as possible." There is, however, much pertinent information that should be presented with the geometry and data display. Figure 12 and other figures show some of the information that might be deemed necessary accoutrements to a display. The self documentation of a display is of practical importance when reviewing data, but it a virtual necessity when hard copies are made for later review. Photographic reproductions in general seem to have a self-shuffling capability that makes accompanying sheets of paper inadequate as documentation.

In Figure 12 less than half of the screen is devoted to the geometry and data display. The header information from the geometry and data files is displayed at the bottom of the screen. These might have information about the geometry configuration, relevant dates, and test parameters. The left side of the screen is devoted to viewing parameter information. The viewpoint, focal point, and scale are especially important if details or otherwise ambiguous views are used. If clipping planes are used, their definition should also be noted on the screen. The position of the light source and the vertical axis are also listed to help orient the viewer.

The right side of the screen is used to show the color-to-data correspondence by a color bar. The color bars for isarithmic and continuous tone displays are similar in both form and function. A color bar, as in Figure 12, shows all hues currently used in the data display. The hues are ordered by the data ranges that they represent, and the limits of the data ranges are labeled beside the corresponding hues. The higher data ranges are near the top of the screen with the lower data ranges ordered below. The intensity, or lightness, of the color bar changes across the width of the bar, but the hue is constant. Since the intensity is used for shading the display and not the representation of data, a variety of intensities are given for each hue to assure the user that all intensities of a hue do indeed represent the same data level. In summary, the vertical position on the color bar defines a hue and is associated with data range labels; the breadth of the color bar is used to show some of the intensities used in shading.

of the extensive calculations required. An interactive tool for the review of data should not emphasize those functions better done elsewhere. It should instead emphasize the freedom to manipulate and experiment with parameters that it gives the analyst. Using an interactive data display tool an analyst can, with a minimal investment of time, make a variety of seemingly minor alterations in the viewing parameters and possibly uncover previously obscured information. The manipulation of viewing parameters and individual elements or data values could be a tedious and time consuming affair if done in a batch mode, but an analyst interactively manipulating the same few quantities could see the results in a few minutes. Fast and easy access to information and a relatively short display generation time are the best features that an interactive data display tool can offer.

THE DISPLAY OF BIVARIATE DATA

A function over a three dimensional surface might be considered three-and-one-half dimensional information in that it gives additional complexity to the three dimensional surface, but it is less complex than an unrestricted function of three variables, which would have values throughout a volume. Paired bivariate functions are another type of information that could be classified as three-and-one-half dimensional. Pairs of bivariate functions, or functions of two independent variables, can be represented with techniques developed for other types of information.

The Domestic Information Display System of Dalton et al.[6] can show bivariate data but the information is discretized by the political divisions basic to the purpose of the system, and the information displayed can in no way be construed as pairs of continuous functions. Chung[45] and Friedman[46] have used interactive computer graphics for the review of multivariate statistical information.

The shading techniques used to give positional cues can be used to convey other information, so pairs of bivariate functions can be displayed using virtually the same raster techniques applied to the display of functions across a three dimensional surface. Color conveyance of data range information, the highlighting of data points, the massaging of bad data, and the saving and restoring of display parameters can all be used in the display of bivariate data. The geometry definition used in this research for bivariate function display was two dimensional and was defined in a Cartesian X-Y plane. It was composed of points and elements just as the previously displayed three dimensional geometries was.

The Topographical Display of Bivariate Data

Single functions over an area have often been displayed as surfaces. Kubert, Szabo, and Guilierie[12] are just one example. The addition of color to such topographical displays allows the simultaneous display of a second bivariate function over the same area. The display of a second function via the addition of color to a three dimensional surface is closely related to the display of a single function over a three dimensional geometry. The lighting model, the use of flat elements to define the surface, and the isarithmic and continuous tone color-data correspondences used to display surface data can be applied to bivariate data.

The display of surfaces over a two dimensional area does have features that distinguish it from the display of a three dimensional geometry, and changes are necessary in the handling of the viewing parameters. Three dimensional clipping planes could be used with area-based displays, but their complexity is not compensated for by the power they give their user. The two dimensional analogue, an arbitrarily oriented clipping line, could be implemented, but this is also more involved than is necessary. A better application of the underlying principle might be the introduction of user specified X and Y coordinate limits. X and Y limits are simpler, easier to use, and little less flexible than arbitrary clipping lines.

The Z axis coordinates need special consideration since their use in the topographical display of bivariate data is dissimilar to their use in the display of a geometry. The definition of the base area in terms of traditional Cartesian X and Y coordinates dictates that the Z axis be displayed as vertical, and in virtually all cases the positive Z axis will be going up. The information represented by the Z coordinate of a point on the displayed surface can, however, differ from the X and Y coordinates of the point by orders of magnitude; independent scaling of the Z axis is a practical necessity. The scaling of the Z axis should be under user control and should be listed on the display with the other viewing parameters, but these concessions to the user are insufficient aids in the extraction of Z axis information from the display. In Figure 22 a grid corresponding to the geometry was drawn in the X-Y plane corresponding to Z = 0. Figure 22 also shows piping, or vertical lines from the grid at Z = 0 to each datum point on the surface. Tic marks along the pipes give the user an indication of the value represented by a point's Z coordinate. The change in value represented by each piping tic is shown with the other viewing parameters, and the familiar color bar is retained from geometry surface data displays.

The viewing parameters controlling a bivariate data display can be manipulated much as they are in the display of three dimensional geometries. The Z coordinate of the light source, eye position, and focal point should be given in Z data units instead of geometry coordinates. The user's dependence on automatic scaling will likely increase as the transformation from geometry coordinates to screen coordinates is further complicated by the requisite Z axis distortion.

Both the isarithmic and the continuous tone color representations of data are practicable in a topographical bivariate data display, as are the highlighting techniques previously discussed. Figures 22 and 23 show the same data, namely the maximum and minimum principal stresses in a quarter section of a plate under tension. In Figure 22 the minimum principal stresses are shown topographically and the maximum principal stresses are shown by the color of the surface. In Figure 23 the roles are reversed.

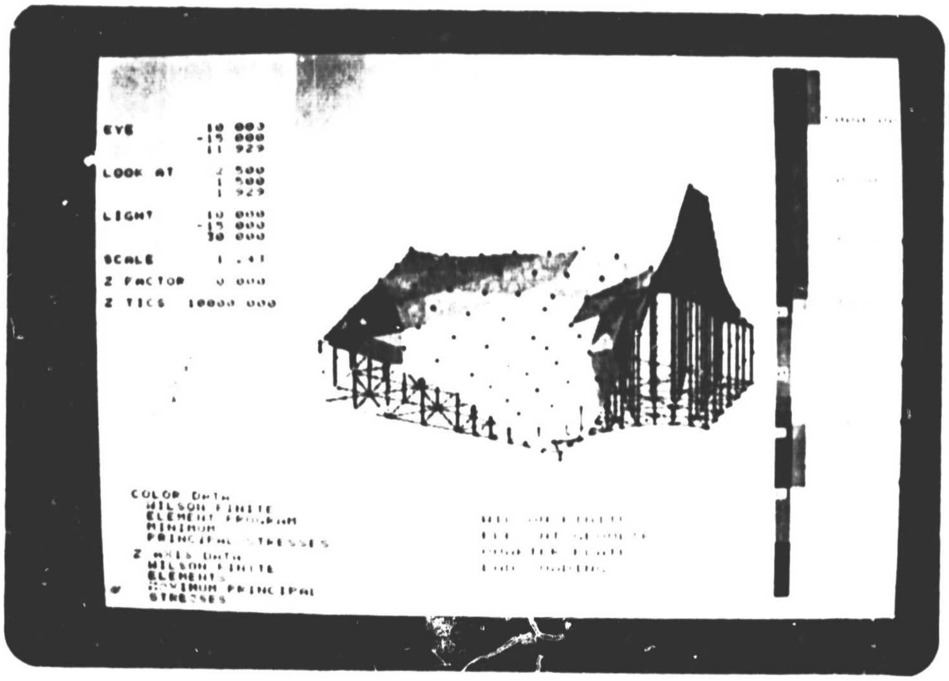


Figure 22

A Topographical Display of Bivariate Functions

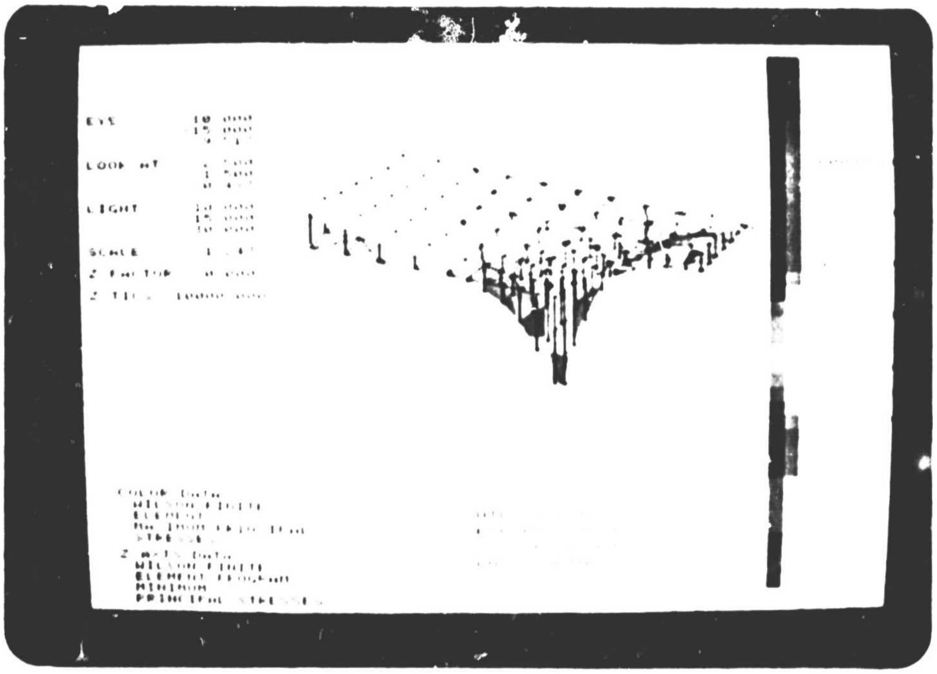


Figure 23

The Functions of Figure 22 Redisplayed

The visual extraction of information from bivariate function displays is harder than might be hoped. Topographical displays are typically intended to convey qualitative information. Attempts at topographical, quantitative displays are hindered by the average person's inexpertise at judging heights. Visual judgements of altitude-related information are further hindered by the uncertainties of an unusual viewing angle and the change from a "real world" context to a computer-controlled cathode ray tube. Although topographical representations of information can be useful whether used independently or in conjunction with a color representation of a second function, both the viewer and the creator of the topographical display should be aware of the limits of topographical displays when quantitative information is to be extracted.

Hue and Intensity Displays of Bivariate Data

The hue, saturation, and intensity color space described in the chapter on Hardware and Software Environment and in various references, e.g. Joblove and Greenberg [40], has three distinct components, but the use of the full range of all three components is both impractical and undesirable. The colors with high or low intensity are nearly indistinguishable, as are those with low saturation. (As the saying goes, "All cats are gray in the dark.") The colors on the surface of the double cone and between the extremes of intensity form a coherent, two dimensional surface in hue and intensity. The hue and intensity are independent coordinates on this surface, and it is possible to use these independent coordinates to represent a pair of bivariate functions. One of the bivariate functions can be represented by changes in surface hues, and the other can be independently represented by changes in intensity or lightness.

Figures 24 and 25 show the same data represented in Figures 22 and 23. Figures 24 and 25 use an orthogonal view of the same quarter plate, thus eliminating any consideration of Z coordinates, whether from the geometry or from the data. In Figure 24 the maximum principal stresses are represented by changes in the hue of the plate, and the minimum principal stresses are represented by changes in the intensity of the plate. The roles are reversed in Figure 25.

A bivariate hue-intensity display requires two simultaneous linear interpolations during the filling of a polygonal area and a two dimensional arrangement of colors in the VLT. The GRAFIC polygon fill routine was changed to base its interpolations on two sets of vertex data instead of one set of vertex VLT indices. The modified polygon filling subroutine uses linear interpolations among the data at the vertices to obtain appropriate values of both functions at the locations represented by each pixel. These function values were then used to calculate a particular VLT

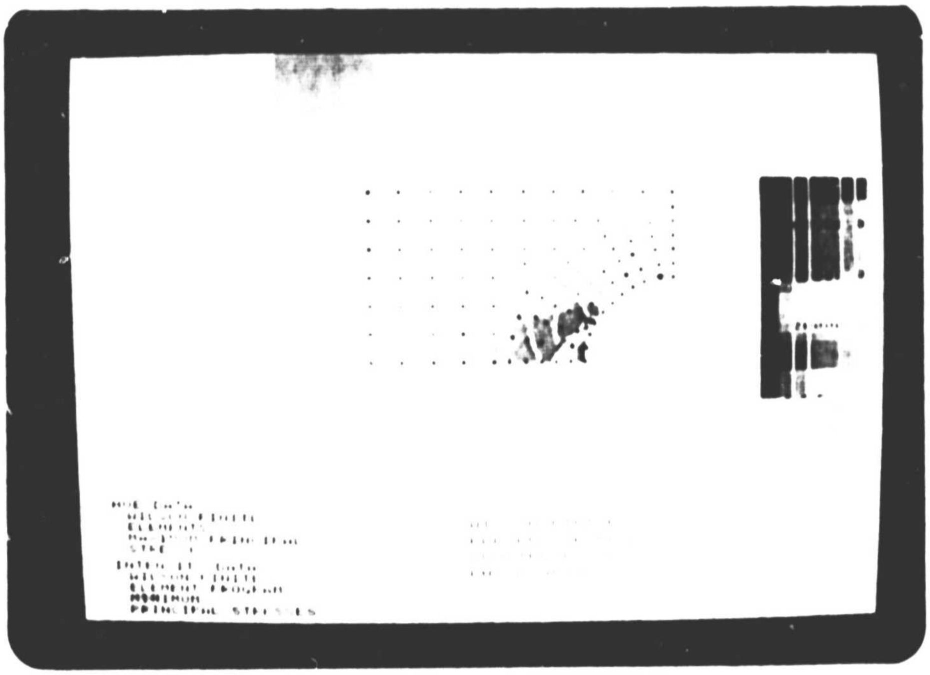


Figure 24

Hue and Intensity Display with Isarithms

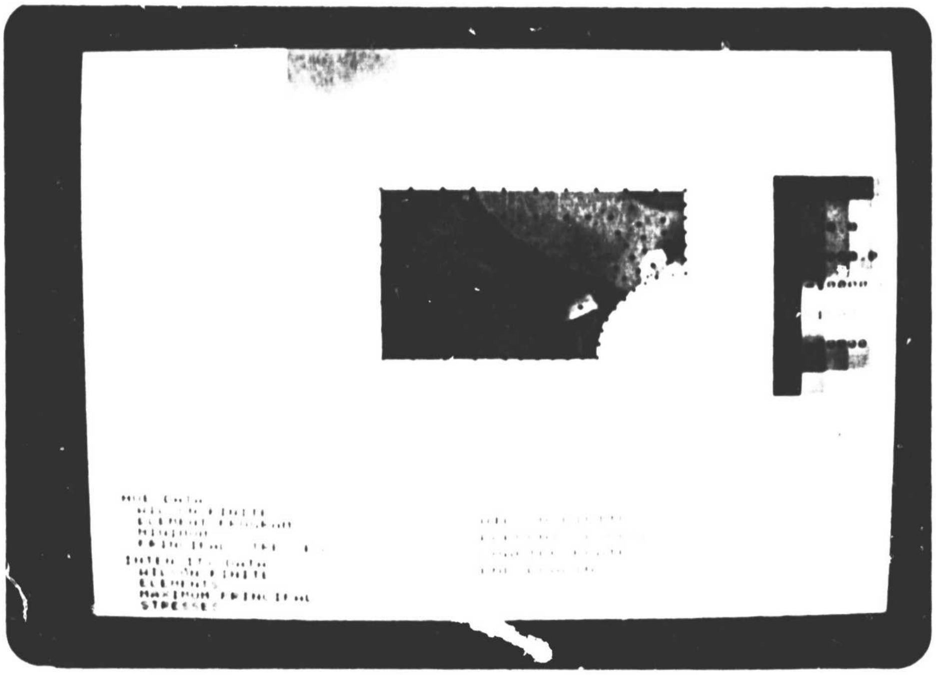


Figure 25

Hue and Intensity Display of Bivariate Functions index for each pixel and therefore the color of the pixel in question. The determination of a color from two function values would be analogous to finding an entry in a table with row and column indices. The revised polygon fill subroutine obviously requires a special sequencing of color definitions in the VLT. The sequence used was a series of hue spectra, with a higher intensity level at each succeeding spectrum.

The modified color bars in Figures 24 and 25 show the ranges of hue and intensity. There are top and bottom data range limits which, for both hue and intensity data, function as the data range limits of continuous tone displays. Figure 24 was created with twenty hues and one hundred intensity levels, while Figure 25 has ninety hues and six intensity levels.

Figure 24 was defined in terms of one hundred intensities in spite of the hardware limitations of the Ramtek, which allow only thirty-one distinct levels of intensity. The large number of intensity levels allows for the fine resolution of data range boundaries by the polygonal area filling subroutine. The surfeit of calculated intensity levels is used to gain a traditional end, i.e. isostress curves. Definitions of the color white are placed in the VLT locations whose intensity index correspond to the labeled stress values. As the intensity interpolation crossed each labeled value, the use of the VLT entry that was defined as white left a distinct line across the surface of the plate.

The hue and intensity display of bivariate data is unfortunately a poor means of conveying information about the bivariate functions. The human eye is insufficiently sensitive to changes in intensity or brightness to allow a viewer to determine the data level represented by the intensity of a pixel. This is especially evident when the hue is in the range between cyan and yellow. In many cases the falsely perceived intensity differences between hues overshadow any actual intensity differences. This domination of the intensity information by the hue variations is shown in Figure 24. What seem to be the obvious intensity variations in the blue portions of the display are virtually perpendicular to the white lines indicating constant intensity.

In Figure 25 the number of distinct intensities was decreased to six in an attempt to increase the distinctions among intensity levels. Although reducing the number of intensity levels helps the viewer to distinguish the intensity levels, the clarity of the data representation is still poor. Even when the viewer can distinguish between adjacent intensity levels, the accurate matching of a specific color in the midst of a display with its twin in the midst of a continuous spectrum of colors is extremely difficult.

The addition of approximate isarithms is helpful in the interpretation of the data shown, but it shows the weakness of the hue and intensity display of bivariate functions. The computer can be made to create hue and intensity displays, but the human eye cannot easily extract the information that was to have been made clear. If it becomes necessary to add isarithms to hue and intensity displays, then the analyst is better off using a faster and easier continuous tone or isarithmic display for one function and adding isarithms representing the second function as a second step.

ORGANIZATIONAL OVERVIEW OF INTERACTION

The physical devices used in a human-computer interface dictate the substance of the interface, but the form is determined by the structure or organization of the interface. Identically organized interfaces could use different devices, just as identically designed houses could be made of straw, of sticks, or of bricks. It is the structure of an interface that dictates when a particular action is allowed and what procedure or sequence of procedures must be executed to initiate the desired action.

Human-computer interfaces should generally be structured to give the human the power to command the available resources and the flexibility to approach his problem in a manner natural to him. The experience of the potential user must, however, be taken into account. Computer aided drafting systems are used by experts for long hours at a stretch, and the interfaces attempt to be as flexible as possible. In some systems the user is even allowed to create his own abbreviated commands that request the performance of an often repeated instruction. Computer aided instruction programs are often seen but once by a user, and the interface must lead the user carefully with what would, to an experienced user, be unbearably sluggish interaction.

Computer aided review of data would probably be an occasional duty of people whose primary responsibilities lie in the acquisition or analysis of the data. Such users would be neither experts nor novices, and an interface designed for them should be structured accordingly. Minimal instructions should suffice, but indications of status and positive responses to inputs should be included.

Methods of Interaction

There are two basic methods of interaction available for user interface, keyboard commands and menu selection. There are a variety of supplemental inputs available to enhance each technique, but most of the user's control must generally be channeled through one technique or the other. Most computer users are familiar with keyboard control of a computer, so a keyboard driven user interface provides a familiar environment for the user. Even when most of the interface is menu based, keyboards are often used to input numbers, file names, and other appropriate quantities. It is possible to have all interaction performed at a standard terminal, but using the keyboard associated with the display device eliminates continual switching of attention between devices. The Ramtek, like many other graphical output devices, has a set of function keys in addition to the standard terminal keys. A template can be used to label the action

Saving Viewing Parameters

The experienced user of a data display package could no doubt obtain a good set of viewing parameters quickly, but if the same geometry is to be viewed with several data files it is ridiculous for the user to reset the viewing parameters for each data display. A means of saving and restoring the display parameters as a unit should be available. The data analysis package developed as part of this research allows the user to store the current display parameters in a temporary file. After viewing other sides or the interior of the geometry, a single command can reset the viewpoint, focal point, light source, clipping planes, and other viewing parameters. Included with what might be termed the "pure" viewing parameters in this temporary file is the definition of the color bar with its hues and their corresponding data levels. The parameters defining the color bar affect any data display and therefore belong with the other viewing parameters.

It is also appropriate to save viewing information for use at later sessions. The three obvious places to store viewing parameters are with the geometry file, with the data file, and in a separate file. The last possibility could lead to excessive viewing information files, which can be cumbersome to store and to document. Storing parameters with the geometry files would be appropriate for pure viewing parameters, but any color bar information associated with the geometry would have at best limited applicability. Different data sets associated with the same geometry might have completely different levels of data or even different types of data, as pressure and temperature. The color bar definition could be saved separately from the the pure viewing parameters, but this would present the same problems associated with separately saved display parameter files. An alternative might be the storing of pure viewing parameters with the geometry file and the color bar definition with the data file.

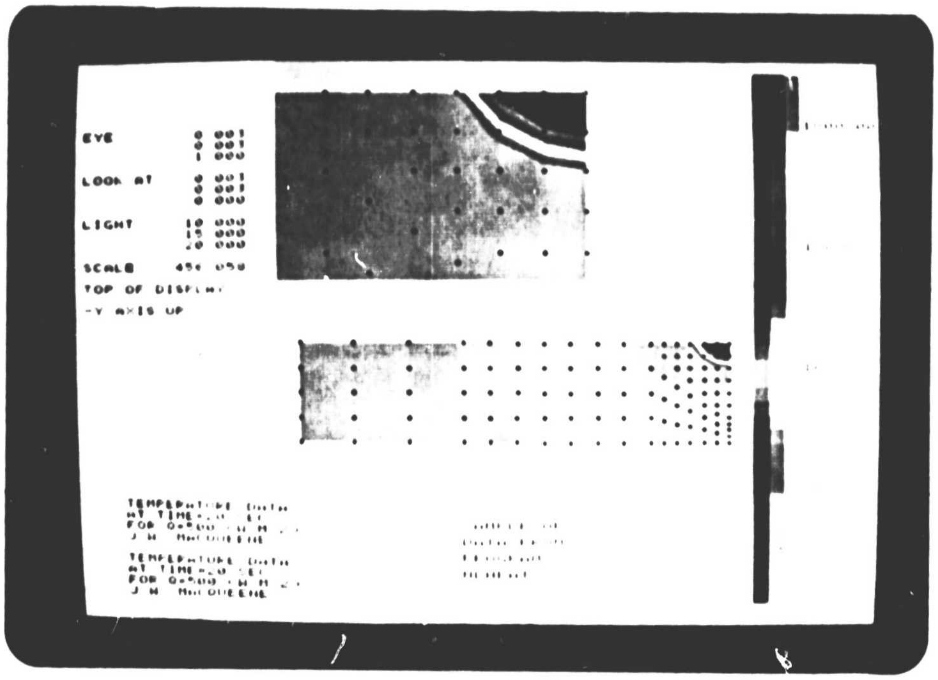
The approach used in this research is the storage of all display parameters with the data file. At any time during a data display session the user is allowed to store the current display parameters with the current data file. This is in addition to the temporary storage of a current set of display parameters. Whenever the user requests that a new data file be used, he can either keep the current display parameters or access those parameters stored with the new data file. This allows the viewing of several data files with identical display parameters and facilitates the storing of reasonable display parameters with previously unaccessed data files.

Split Screen Viewing

In most cases a data analyst would want a large view of the geometry and data under investigation, but comparisons of displays suffer badly when performed sequentially or subjected to lengthy delays. Splitting the display screen allows the simultaneous viewing of separate data sets or the viewing of a geometry from two viewpoints. The latter might be particularly useful in examining a local phenomenon and its global significance, as seen in Figure 13.

Figure 14 shows a comparison of two mach 0.79 wind tunnel tests on a wing with a turboprop engine. The top view shows the coefficients of pressure over the upper surface of the wing with the engine rotating at 6703 RPM. The lower view shows the same surface with the engine rotating at 8472 RPM.

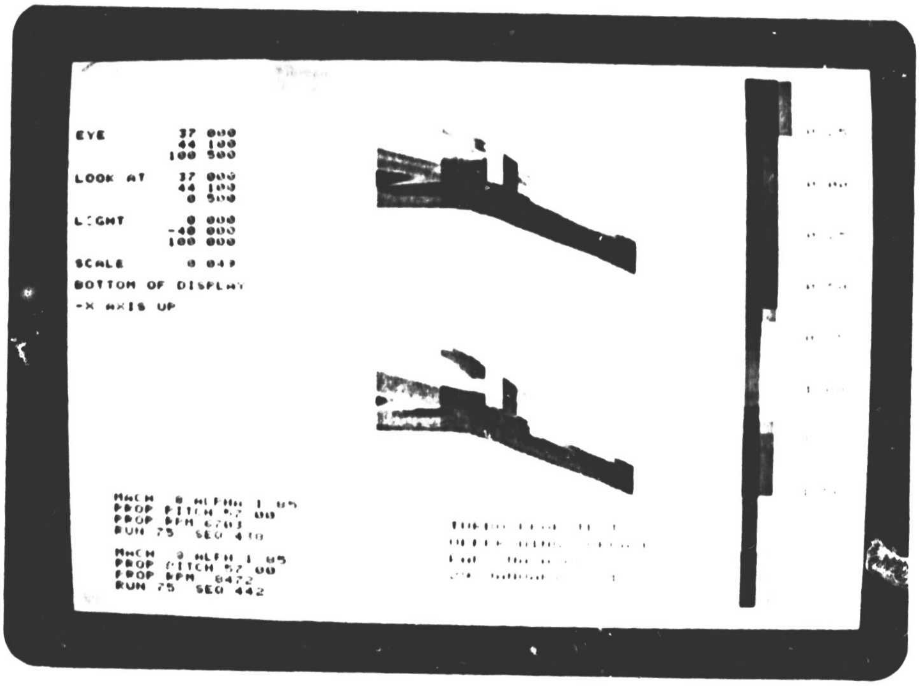
An attempt was made to run pixel-by-pixel comparisons of split screen views. If the two simultaneous views are of the same geometry and use the same viewing parameters, there should be a simple relationship between the screen locations of the two images. In Figure 14 one image is 200 pixels below the other. A comparison of the two images can reveal if the VLT locations used to color the geometry indicate higher, lower, or nearly equal data values for the area represented by individual pixels of the upper or lower display. Equality would be determinable to within the data



```
EYE          0 001
             0 001
             1 000

LOOK AT      0 001
             0 001
             0 000

LIGHT       10 000
            15 000
            20 000

SCALE      45[illegible] 050

TOP OF DISPLAY

-Y AXIS UP

TEMPERATURE DATA
AT TIME=[illegible]
FOR Q=500 [illegible]
J H MACQUEENE

TEMPERATURE DATA
AT TIME=20 SEC
FOR Q=500 [illegible]
J H MACQUEENE
```

Figure 13

Split Screen Detailing



```
EYE         37 000
            44 100
           100 500

LOOK AT     37 000
            44 100
             0 500

LIGHT        0 000
           -40 000
           100 000

SCALE        0 04[illegible]

BOTTOM OF DISPLAY

-X AXIS UP

MACH  0 ALFHA 1 05
PROP PITCH 57 00
PROP RPM 6703
RUN 75   SEQ 410

MACH  0 ALFH 1 05
PROP PITCH 57 00
PROP RPM  8472
RUN 75   SEQ 442
```

Figure 14

Split Screen Comparison

range represented by a single VLT index. The precision of the equality determination is thus a function of the style of display and the definition of the color-to-data correspondence, both of which are user controlled. This pixel-by-pixel comparison was performed for isarithmic displays, but the results were at best difficult to interpret. More direct comparisons of data sets, be they visual or numerical, would produce more easily comprehensible results.

Handling Large Geometries

Programs for engineering applications, especially those employing iterative techniques, often analyze very large models. It is therefore important for a data display package to handle reasonably large models. The size and speed limitations of the PDP 11/40 made the handling of truly large geometries infeasible in the research model, but geometries with approximately 900 total points and elements were displayed with associated data.

The code needed for generating a display on the Ramtek leaves sufficient memory for descriptions of approximately 400 total points and elements if a geometry with 1000 total elements and points is to be displayed. A division or series of divisions of the geometry information is obviously necessary. Arrays of flags describing the status of all elements and points are kept, and a common set of data structure nodes is used to describe both elements and points. The allocation of these common nodes is done during execution to maximize the number of nodes available to the user.

The Newell, Newell, and Sancha algorithm creates a strict ordering of all elements, which provides a useful means of separating the geometry into manageable sections. Since the elements furthest from the viewer are to be displayed first in the Newell, Newell, and Sancha algorithm, a split of the geometry would save the further elements and place the nearer elements back into the pool of unexamined elements. The point nodes are examined for usage at this time, since in certain cases more nodal space can be recovered by reclamation of unused points than by reclamation of eliminated or unexamined elements. When all elements have been examined and the largest practical list of the furthest elements has been created, the list is displayed. The displayed elements are appropriately flagged, and the display procedure is restarted with a smaller set of elements to be displayed. The geometry in Figure 15 has 768 elements and 125 points and took fifteen minutes to display. It took five minutes to create Figure 14, a geometry with 318 elements and 184 points. The latter geometry fit into the available memory, and no list splitting or resorting of elements was required. The sorting, splitting, and re-sorting of the elements shown in figure 12 took roughly five minutes of the fifteen minute display time.

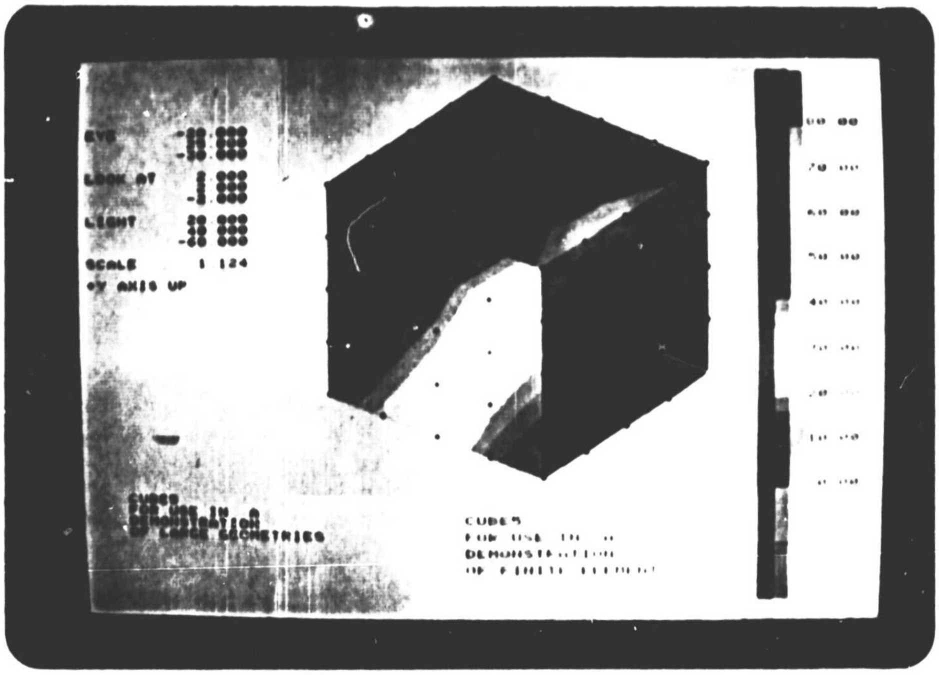


Figure 15

Display of a Large Geometry

INTERACTIVE DATA MANIPULATION

The power to interactively review color displays of complex data is certainly useful, but still more powerful tools can be made available to the engineer. The previously described interaction allows the analyst to control the representation of data, but it does not allow him to access the raw numerical data being displayed. This restriction is like the placing of a glass cover over a work of art: it is appropriate when the piece is out for public viewing, but it tends to hinder the artist when he is trying to finish his masterpiece. A data analyst should be able to extract numerical information from the data being displayed.

The Ramtek 9351 has an overlay mode that allows writing and drawing in white without corrupting any information on the Ramtek screen. The overlaid drawing and writing, or highlighting, can be separately erased to leave the original display unchanged. This hardware feature provides an appropriate means of accessing information. The following is an example of how and why an analyst might interact with his data to obtain a better understanding of its significance.

Figure 16 shows wind tunnel test results displayed using an automatic color bar generation routine. One point in the center of the wing has an extremely high data value, and this has distorted the automatically generated color bar to the point that little useful information is conveyed to the analyst. In Figure 17 the user has requested information on node 20, the node with the high datum. The analyst may access specific nodal information by entering the appropriate node number, or he may also step sequentially through the nodes and thus peruse information on several consecutive nodes. The overlay mode of the Ramtek is used to verify the location of the point in question and to show the node number, the datum associated with the node, and a flag indicating the current status of the datum. The datum displayed in Figure 17 is flagged as valid or "good," since the analyst has not declared it otherwise.

The analyst can at this point request that the node being examined be added to a list of bad data points. Any further display of the geometry will show all elements defined with node 20 in shades of gray to indicate that the data for that portion of the geometry is uncertain. Requests for highlighted information on node 20 will yield the same data value, but a flag value of "bad" instead of "good" will accompany the the printed value.

The analyst can alternately choose to massage or specifically alter the datum at any particular node. When

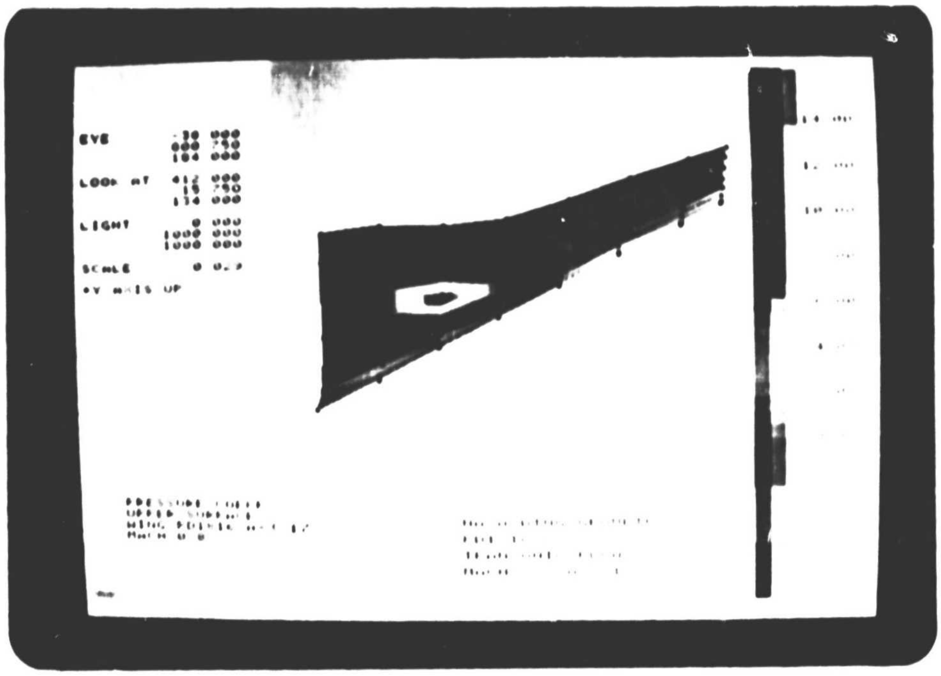


Figure 16

The Effect of a Bad Data Point

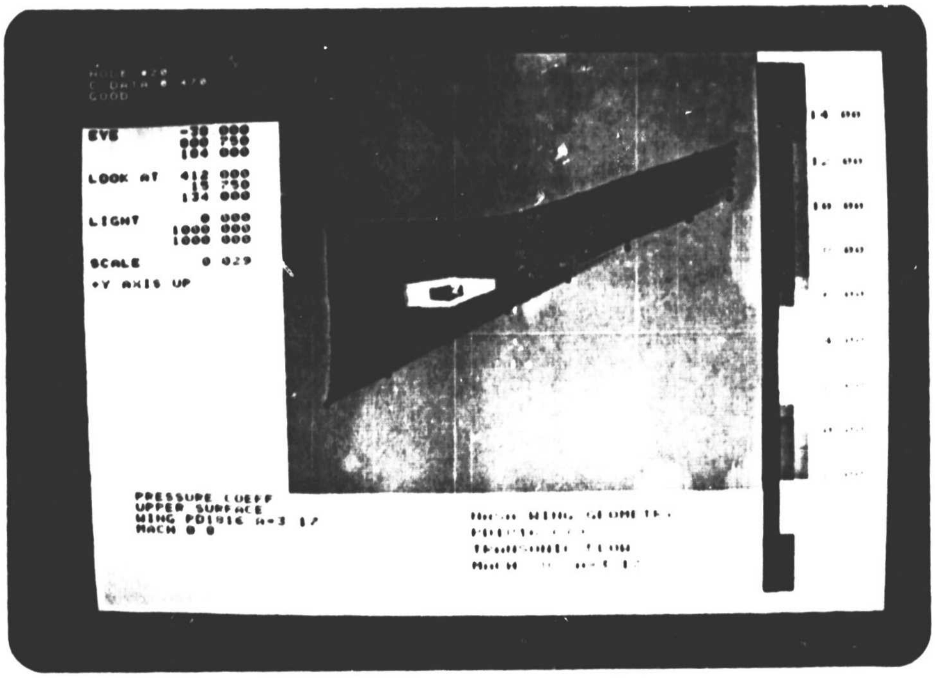


Figure 17

An Overlaid Display of Information

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

this request is made, the data display package queries him regarding the data value that he wishes to assign to the node in question. This step is shown in Figure 18. Subsequent displays will use the analyst's input value as the datum in all references to a massaged node. Highlighted information on the node will show the massaged data value, and the flag will indicate that the data has been altered. Figure 19 shows the same data as Figures 16 through 18, but it uses a different color bar. The upper view in Figure 19 shows the result of declaring node 20 to have a bad data value, and the lower view shows the massaged data display.

Massaging a data point or declaring it to have a bad data value should not destroy the original value. The above example of data manipulation used separate lists of bad and massaged nodes. These lists were stored with the data files as trailing additions. Any manipulation of data points was easily reversible, and the original data values were automatically put back into full use.

It should be stressed that the lists of bad and massaged data points were generated by the analyst and not by the data display package. Computer evaluation of the viability of individual data points within a data set would require the addition of task-specific software and is beyond the scope of the data display package described.

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

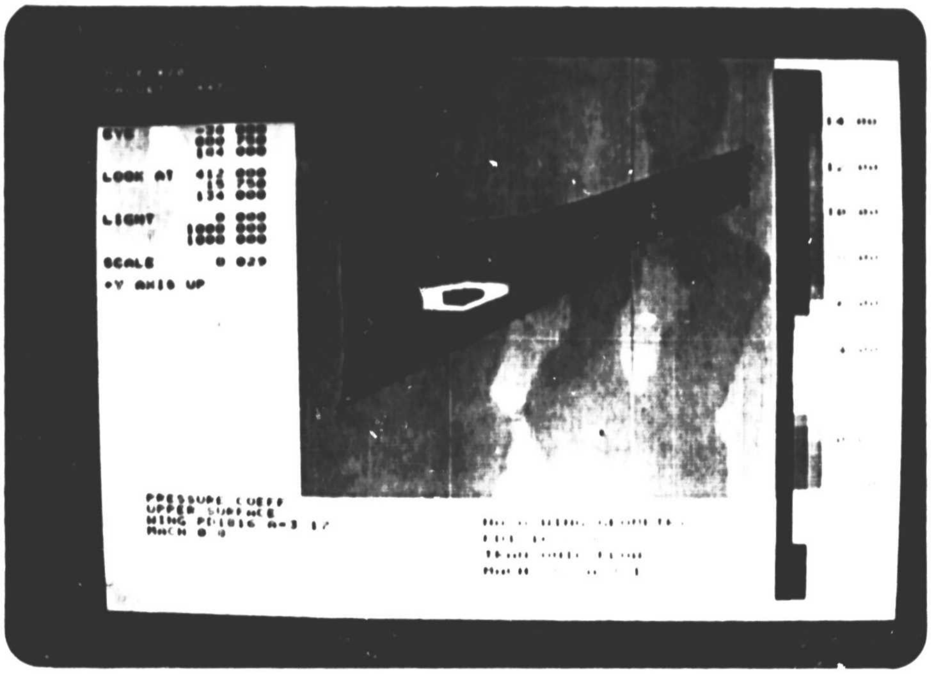


Figure 18

Massaging a Data Point

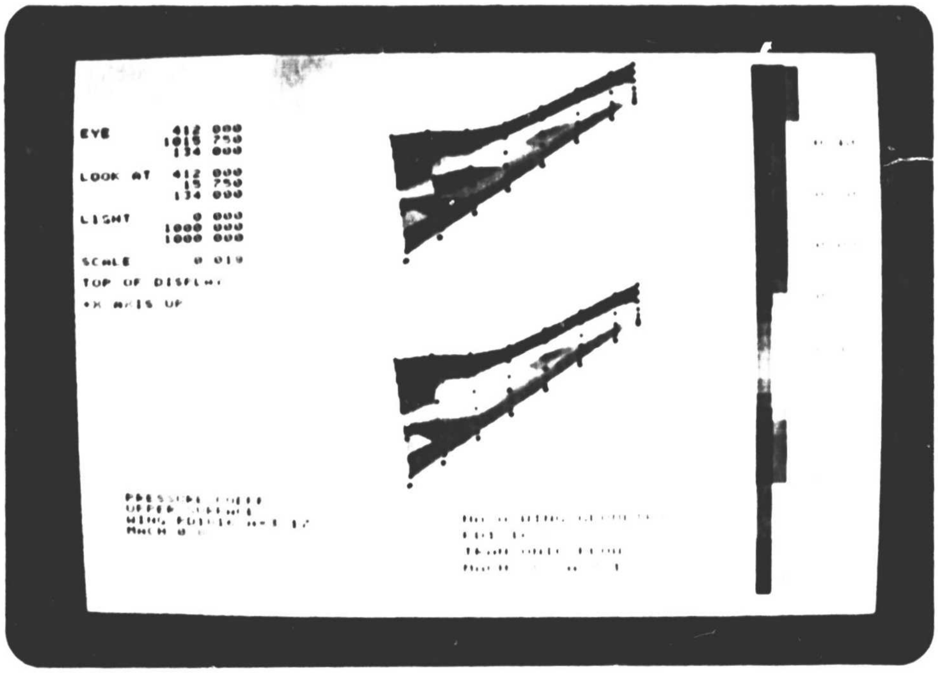


Figure 19

The Effects of Bad and Massaged Data Points

Highlighting Geometry Elements

The Ramtek's overlay mode can also be used to highlight individual elements of a displayed geometry. Figure 20 shows a highlighted element outlined and labeled. The nodes at the corners of the element and their associated data values are listed in the upper right corner of the Ramtek screen. The analyst can, as with nodes, step through sequences of elements or request the highlighting of specific elements.

Some manipulation of the elements of the geometry was implemented in the data display package developed. While highlighting elements the analyst can add the element to a list of suppressed plates. If the geometry is subsequently displayed, all elements on the suppression list are purged from the display list. The interior view in Figure 21 could not be obtained with clipping planes alone. Information concerning an element is not affected by the element's suppression, and subsequent highlighting is possible, as shown in Figure 21. It is, of course, possible to remove an element from the suppression list.

The power of interaction lies in the flexibility it gives its user. Few of the geometry or data display techniques described in the previous chapter cannot be done in a batch environment, and some things, such as the creation of high quality output, are better done in a batch mode because

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

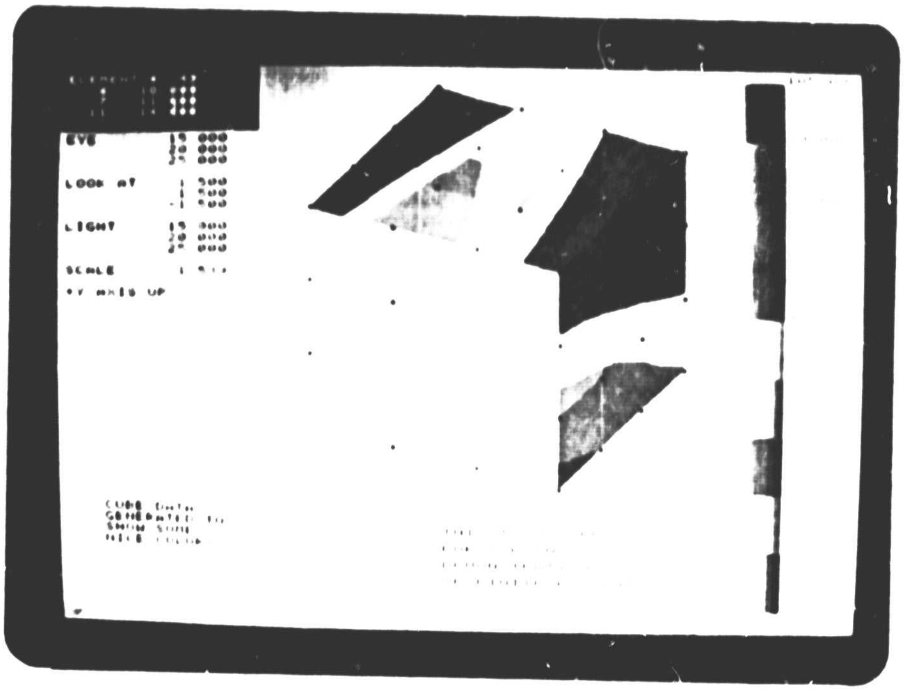


Figure 20

Highlighting Elements

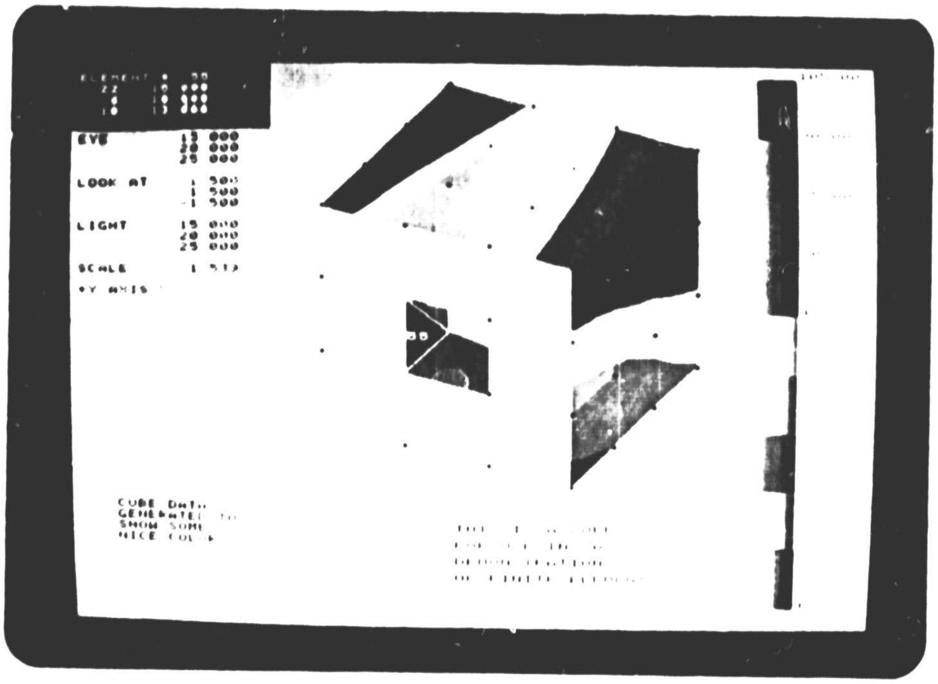


Figure 21

Suppression of Elements

associated with each function key, thus avoiding both extensive mnemonics for commands and subsidiary instruction sheets listing what keyboard entries will initiate what actions.

A menu-driven interface generally lists on the screen the set of commands available to the user and allows the locational selection of actions to be performed. Menu-driven interfaces can be quite powerful and are widely used, but menu-driven interaction is dependent on the device used to control the locational input. The Ramtek's joystick is velocity controlled, and the cursor moves slowly and is often difficult to control. It is a poor device for selecting small and widely separated items on the screen. This makes menu-driven interfaces using the Ramtek joystick slow and unwieldy. The addition of a substantial menu area would further cramp the already crowded Ramtek screen. The menu could be alternately displayed and erased as commands are entered, but this would not affect the inadequacy of the joystick-controlled cursor. The joystick can be an excellent device for interaction, but the particular joystick available was neither fast enough nor accurate enough to be effective at menu selection.

An increasingly popular device is the tablet. Tablets are flat surfaces with the ability to determine the location of a stylus or pen. Tablets are normally rectangular and give the cursor location in Cartesian coordinates. In

c-2

general tablets are square and vary from eleven to thirty-six inches on a side. Numerous means of determining the stylus location are used in tablets. Grids of wires with distinct signals and partially conductive tablet surfaces giving voltage gradients have been used. Strip microphones along the sides of a tablet are used to infer position from acoustic delay timing. Other tablets determine stylus coordinates from the time lag of electrically induced strain waves propagating through magnetorestrictive materials. Since the user is moving a pen or stylus around on a flat surface, the tablet is natural to use. The effectiveness of a tablet is not lessened by placing a piece of paper over its surface, so an electric stylus or pen can double as a ball point pen for sketching purposes. Drawings may be digitized with a tablet, or a menu of available commands may be placed upon the face of the tablet.

A tablet-based, menu-driven interface could quite effectively service analysts with intermediate knowledge of a data display package. A permanent menu could and should be placed on the surface of the tablet, but a corresponding on-screen menu would be helpful for less experienced users. Figure 26 shows a menu as it might be displayed. The white plus sign to right of center is the Ramtek cursor, which would nominally be controlled by the tablet stylus. The analyst could select commands from either the screen menu or the tablet menu; the position of the stylus on the tablet

ORIGINAL PAGE
BLACK AND WHITE PHOTOGRAPH

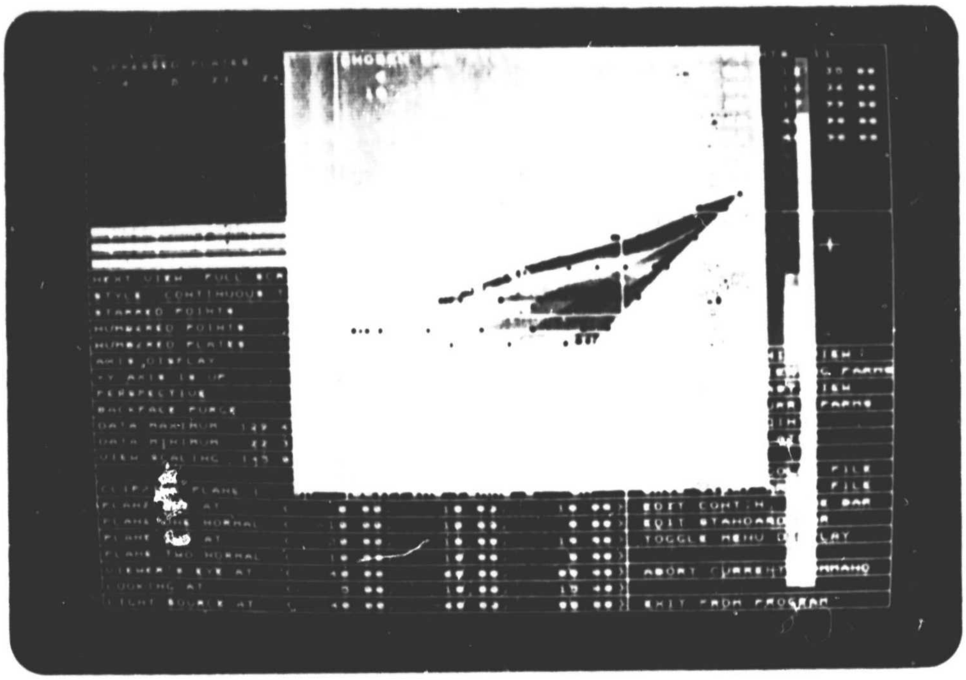


Figure 26

A Menu Overlaying a Display

would correspond exactly to the position of the screen cursor, and the locations of commands would be identical for both menus.

An on-screen menu as in Figure 26 would probably make use of overlaying or nondestructiva writeover capabilities. Extensive menus and other information can be shown to the analyst and separately erased when viewing of displayed geometries and data is desired. An interactively controlled on-screen menu can give the user more information than the list of commands on a tablet menu. The status of display options, geometry and data file information, lists of edited points and elements, and other useful information are shown in Figure 26. The menu in Figure 26 also has a command controlling the display of the menu and its associated information. Unfortunately, no tablet was available for this research, and the Ramtek keyboard was used as the primary means of interaction.

The interface implemented as part of this research is described below. The limitations imposed by hardware, software, and time left their marks on the interface, and to characterize the resulting interface as ideal would be to make a virtue of necessity. Most commands were entered via the Ramtek's function keys, although the standard keyboard was often used. The Ramtek joystick was used to control the on-screen cursor in two special cases, and lights on the Ramtek keyboard were used to indicate the current operational mode.

The implemented interface treats most of the available commands as bit players used to set the stage for the stars of the show, the display creation commands. The user can set and reset whatever display parameters he wishes; the entrances and exits of these bit players are quick and quiet. When the user is satisfied with his display parameters, he can initiate the relatively lengthy display process.

Because the limited size of the PDP 11/40 memory necessitated the division of the developed software into several distinct programs, separate command states were implemented.

The main state or primary mode has the user access to file management and the initialization of display processes. The saving and restoration of viewing parameters is controlled from the primary mode. The primary mode also gives the user access to the other command modes. The parameter modification mode allows the user to modify the viewing parameters, including the viewing vector, display styles and options, color bars, the clipping planes, and other parameters. The lists of massaged nodes, bad data points, and suppressed elements can also be edited from the parameter modification mode. The file and program input and output required for the entry into parameter editing mode and the return to primary mode took approximately thirty seconds. This delay is excessive if only one or two trivial changes are necessary, so certain often-used commands involving the screen areas and display styles are also available in the primary mode.

The highlighting mode is the other major command mode, and it controls the user interaction with the data being displayed. If elements are being highlighted, the current element is outlined and numbered on the most recent display of the geometry. The points at the vertices of the element are listed with their associated data in the upper left corner of the screen. If points are being highlighted, the geometry point is numbered and marked on the latest display, and its associated datum is given in the upper left corner of the screen.

The analyst can choose which point or element is highlighted by entering a specific number at the Ramtek keyboard, or he may step through a sequence of points or elements. Ramtek function keys with vertical arrows are used to determine whether the highlighting goes up or down through the numbered points or elements. The carriage return is used to initiate a step in a highlighting sequence.

Elements may be added to the list of suppressed elements by a keyboard request while the element is highlighted. Elements may be similarly deleted from the suppression list. Points may have their associated value declared invalid or massaged while the point is being highlighted.

The interaction described is divided into several modes by necessity and not by choice. The separation of the highlighting features is reasonable, but the manipulation of the viewing parameters should not be a major operation. Any user, and especially an occasional user, will periodically forget to make a minor adjustment or want to see a slightly altered display. A small change in a viewing parameter should not require two additional commands for mode changes. A description of the commands available in each command mode is given in Appendix A.

PRACTICAL CONSIDERATIONS

The techniques and examples of data display, review, and manipulation developed in this work are intended to be of general utility and not restricted to any particular type of data. A production application of interactive graphical data review would likely have numerous changes in both the display and the characteristics of its particular application. The display devices available, the computer used to control the interaction, any links to other computers, the type of data to be displayed, and the intended use of the displays should all be considered when designing a production data display package.

The most probable environment for a data display package would be a dedicated computer interfaced to a color raster display device. A means of transferring information to the dedicated computer must be available. If the output of large theoretical analysis programs is to be displayed, a direct line to the mainframe computer performing the analysis would be in order. Remote or mobile testing facilities may require magnetic tape transfers. Regardless of the source of the data and the geometries to be displayed, the computer controlling the interaction must be large enough to handle fairly large programs and geometries. The PDP 11/40 used in this research is too small and weak to be considered practical as an implementation computer.

The problems encountered on the PDP 11/40 did manifest those areas deserving attention. The definition of a geometry and the associated data values requires large areas of memory, so the compact and efficient handling of data and geometry definitions should be a prime consideration in the design of a data display package. The rolling of programs into memory and the reading and rereading of geometry and data files slow the execution of a program much more than any programming inefficiencies in interactive portions of the code. Delays caused by slight inefficiencies in the interactive sections of a program are generally short, perhaps one fourth of a second. For a computer to read and cursorily examine a geometry definition takes considerably longer, perhaps thirty seconds on the PDP 11/40. Even if the computer has features such as extended memory or special provisions for swapping information in and out of disk storage, the inefficient handling of information will cause unnecessary delays in execution which waste the analyst's time and tax his patience. If the computer driving the data display has several users, the effects of unnecessary delays are magnified.

The Ramtek 9351 is an adequate output device, but newer hardware capabilities would be welcome. The hardware zooming capabilities of newer raster display devices are sometimes coupled with panning capabilities, which would be quite useful. The effect of panning around an enlarged display would be like moving a magnifying glass over a picture; the user would choose which portion of a display to see at the enlarged scale. Hardware zooming and panning would thus replace the time consuming redisplay necessary in the implemented detailing feature. The 512 by 512 pixel resolution of the Ramtek 9351 was acceptable, and unless the display time is significantly lowered, a fourfold increase in display times would be an exorbitant price to pay for doubled resolution. Most of the display time for simple geometries is spent interpolating VLT indices across the interior of polygons, so interpolative polygon-filling hardware could allow higher resolution display devices to be much better than the 9351.

Possible Extensions of a Data Display Package

Many possible features of data display packages have not been discussed in previous chapters. Indeed, for certain purposes some of the described features or enhancements may be superfluous or undesirable. The removal from the display list of elements facing away from the viewing position appreciably reduces the time required to produce a display, but in some cases the inclusion of rear facing elements may be desirable.

If presentation quality output is desired, as it might be for photographic reproductions of particular displays, an additional display style could be added. Many shortcuts taken for the sake of interaction can be eliminated to produce a finer display. The lighting model described used no shadowing and only crude reflectivity calculations. Greater realism can be attained with a well-developed lighting model accounting for diffuse and specular reflection. Different reflectance characteristics on different parts of the displayed geometry could be used to enhance distinctions between parts. The weaknesses in the Newell, Newell, and Sancha hidden surface removal algorithm may also corrupt certain portions of a display, but a more sophisticated hidden surface removal algorithm, such as the Watkins algorithm, could be used to eliminate these display anomalies. The Watkins algorithm can also be used to show cross sections precisely at a clipping plane rather than at the nodes just inside the clipping plane. A higher resolution display device with finer control over hues and intensities would also be desirable for presentation quality output.

A more sophisticated treatment of the surfaces defined in the geometry is also possible. Complex shapes, such as the skin of an airplane, are often described mathematically in terms of patches or curved surfaces among points in three dimensional space. For the interactive review of surface data these patches may be crudely reduced to a few large, flat, triangular plates, but for quality displays using polygonal elements each patch must be divided into many small, flat plates that, taken in toto, closely approximate the three dimensional patch. An airplane so represented might consist of 30,000 plates, which makes such displays inappropriate for normal interactive data review.

A more sophisticated approach to the filling of polygons on the display screen can also be developed. The linear interpolations discussed in the Interactive Data Display chapter give relatively crude isarithms. Contouring algorithms for two dimensional data exist and could be extended to a three dimensional surface. A traditional two dimensional contouring algorithm could use the screen coordinates of transformed data points to give smoother isarithms, but this could not easily be synchronized with the described viewing algorithms. Complex or partially obscured parts of a three dimensional geometry would cause further problems for a screen coordinate-based contouring scheme.

An analyst might for a number of reasons wish to see a particular sequence of displays and have the ability to address a series of displays as a unit. The sequence might be ordered by an angle of rotation, by the acquisition chronology, or by some other criterion. The designer of a computer controlled presentation of sequential displays must avoid both the Scylla of lengthy display times and the Charybdis of excessive storage requirements. The attempt at sequenced displays in this research emulated Odysseus and fell prey to both Scylla and Charybdis. Although the size limitations of the PDP 11/40 memory forced the elimination of the developed sequencing code as nonessential, sequencing code used in conjunction with some form of hardcopy device could be quite effective. A video film recorder would be an appropriate hardcopy device for saving sequential displays.

Several minor extensions or small changes of technique for production systems are worthy of mention. The described massaging of specific data values depends on user-input replacement values. An alternative means of ascribing a data value to a geometry point would be an averaging of certain other data. The user-specified points would likely be nearby points on the same surface. The highlighting of individual points on the geometry and the display of the associated data value was controlled by user requests of specific node numbers. If the uncertainties caused by points on hidden parts of the geometry could be circumvented, choosing points to be highlighted by a on-screen cursor would be better than requesting specific point numbers. The highlighting of entire subassemblies could be used to help the analyst maintain his orientation in magnified views of complex structures. Highlighting could also be used to indicate which data are in a specific range, e.g. the analyst might request that all data greater than some specified level be highlighted.

A production implementation of a data display and review package would be heavily influenced by the available hardware. If system specifications for a system to be developed by perhaps 1990 were listed, they would include hardware for most of the operations that waste the user's time. Virtual memory would allow fairly easy manipulation of large geometry and data files, although the implementation software should still condence, as possible, the information from these auxiliary files. The desired view of the geometry should be chosen by manipulating a wire frame or other simple representation with hardware scaling, translation, and rotation. The longer, full color raster display would be created a minimum number of times. Hardware for the interpolative fill of polygons would be of considerable benefit, but the hardware implementation of an inexpensive, robust hidden surface algorithm would be better. Although general purpose hidden surface hardware is not currently available, it will probably be developed in the next few years. The hardware for hidden surface removal may handle only flat, polygonal elements, but if a large number of flat elements could be handled this would not be a major handicap in the presentation of most engineering data.

Many other features of data display systems can be listed. Application specific enhancements and the possibilities of new hardware capabilities make an attempt at a complete list absurd. The features implemented and suggested do, however, show some of the directions in which interactive data review and manipulation may proceed. Many more extensions and interactive features could be described, but more important than a list of features is the recognition of their existence, their variety, and their utility.

CONCLUSIONS

The interactive techniques developed in this research give an engineer the power to quickly examine complex data for trends, anomalies, and other significant facets. Shaded displays of three dimensional geometries with color representations of associated data have been shown as a viable means of portraying complex information, even when the display is degraded for the sake of interactivity. Detail enlargement, node and element numbering, clipping planes, and element suppression have proven to be promising techniques for the isolation of particulars and the investigation of data interior to a geometry. Split screen viewing for comparisons either of different views or of distinct data sets was also developed. Various features that might be found in a production system were also implemented, such as perspective, user choice of the geometry's spatial orientation, and the efficient handling of relatively large geometries.

Both the developed isarithmic color-to-data correspondence and the applied continuous tone correspondence were shown to be applicable to the interactive representation of complex information. The continuous tone technique is better for the illumination of trends and produces a more visually pleasing display, but the isarithmic technique allows more positive identification of data levels. The user's control of the color-to-data correspondences in either display style allows the user to interactively emphasize those portions of the displayed data's range that are of particular interest.

Interactive data review is a powerful tool, but it does not fully utilize the capabilities of contemporary or future interactive computer graphics. This research has demonstrated the power of interactive access to individual data which are part of a complex display. The significance of individual values can be determined, and the deleterious effects of errant data can be eliminated by the highlighting and data massaging techniques developed. Advances in hardware design will speed computer displays, and many functions may be performed in hardware, but access to displayed numerical information will still be a powerful tool for the interactive review of information. The highlighting of individual data and geometry elements will have continued applicability and can be developed for use with intricate geometries and complex data forms. Future work in data display should make full use of hardware capabilities, but interactivity will still be the key to effective use of human and computer resources.

The attempted representations of bivariate functions were unsuccessful in that the techniques developed produced inadequate representations of the functions. Topographical displays of bivariate functions had many areas of uncertain value caused by the hiding of height cues and the obscuring of rear areas of the surface. Color, topographical displays of pairs of bivariate functions could be useful if the viewing were dynamic, but the static displays in this research could not fully represent two bivariate functions. The research on bivariate function display did, however, point out both strengths and limitations of interactive computer graphics in data review. Highlighting of specific nodal and elemental information was used to extract specific data in spite of the weaknesses of the display.

The hue and intensity displays inadequately represent pairs of bivariate functions, but they demonstrate one limitation on computer displays of information. Computer driven displays can depict subtle information by variations in intensity, but human vision is limited in its ability to extract information thus displayed. Intensity variations are better used to show the three dimensionality of a geometry than to show independent functions.

Interactive computer graphics must provide the engineer with lucid representations of a variety of functions or types of information without completely removing him from familiar data handling techniques. Interactive computer graphics is better applied as an extension of an engineer's data reviewing capabilities than as a complete replacement of procedures familiar to the engineer. If the engineer is not comfortable with a display format and cognizant of its implications, he is apt to set too much or too little store by the graphical output. In either case the primary function of graphical display, the accurate representation of information, is not performed.

LIST OF REFERENCES

LIST OF REFERENCES

[1] Kaplan, R. Computer Generated Footprints for Displaying Results of Data Analysis. The HRSI Review, vol. 11, #2-3, September-December 1980.

[2] Benson, W., and Ktous, B. Interactive Analysis and Display of Tabular Data. Computer Graphics, vol. 11 #2, summer 1977.

[3] Brassel, K. , Utano, J., and Hanson, P. The Buffalo Crime Mapping System: A Design Strategy for the Display of Spatially Referenced Crime Data. Computer Graphics, vol. 11 #2, summer 1977.

[4] Tanimoto, S. Color Mapping Techniques for Computer-Aided Design of VLSI Systems. Computers & Graphics, vol. 5 #2. 1980.

[5] McCleary, L. Techniques for the Display of Ocean Data on a Raster Driven CRT. Computer Graphics, vol. 11 #2, summer 1977.

[6] Dalton, J., Billingsley, J., Quann, J., and Bracken, P. Interactive Color Map Displays of Domestic Information. Computer Graphics vol. 13 #2, August 1979.

[7] Bengtsson, B., and Nordbeck, S. Construction of Isarithms and Isarithmic Maps by Computers. BIT, vol. 4 #2, 1964.

[8] McLain, D. Drawing Contours from Arbitrary Data Points. Computer Journal, vol. 17, 1974.

[9] Gold, C., Charters, T., and Ramsden, J. Automated Contour Mapping Using Triangular Element Data Structures and an Interpolant Over Each Irregular Triangular Domain. Computer Graphics, vol. 11 #2, summer 1977.

[10] Dutton, G. An Extensible Approach to Imagery of Gridded Data. Computer Graphics, vol. 11, #2, summer 1977.

[11] Fowler, R. and Little, J. Automatic Extraction of Irregular Network Digital Terrain Models. Computer Graphics, vol. 13 #2, August 1979.

[12] Kubert, J. C., Szabo, J., and Guilierie, S. "The Perspective Representation of Functions of Two Variables." Journal of the ACM, vol. 15 #7, April, 1968.

[13] Siddell, P., Anderson, U., Knopp, J., and Bassingwaighte, J. Computer Graphics in Simulation of Cardiovascular Transport Phenomena. Computers & Graphics, vol. 1 #2-3, 1975.

[14] SynthaVision Version 1.1 Reference Manual, Control Data Corporation, St. Paul, Minnesota, 1981.

[15] Christiansen, H. N. and Stephenson, M. MOVIE.BYU - A General Purpose Computer Graphics Display System, Proceedings of the Symposium on Applications of Computer Methods in Engineering, University of Southern California, vol. 2, Los Angeles, August, 1977.

[16] Christiansen, H. N. and Stephenson, M. Graphics Utah Style '77, Workshop Notes. Snowbird Resort, Utah, 1977.

[17] Christiansen, H. N. Applications of Continuous Tone Computer Generated Images in Structural Mechanics. <u>Structural Mechanics Computer Programs</u> Pilkey, W. D., editor. University Press of Virginia, Charlottesville, Virginia, 1974.

[18] Newell, M. E., Newell, R. G., and Sancha, T. L. A New Approach to the Shaded Picture Problem. Proceedings of the ACM National Conference, 1972.

[19] Watkins, G. S., "A Real-Time Visible Surface Algorithm," Computer Science Department, University of Utah, UTECH-CSC-70-101, June, 1970.

[20] Warnock, J. E., "A Hidden-Surface Algorithm for Computer-Generated Halftone Pictures,", Commputer Science Department, University of Utah, TR 4-15, June, 1969.

[21] Blinn, J., Carpenter, L., Lane, J., and Whitted, T. Scan Line Methods for Displaying Parametrically Defined Surfaces, Communications of the ACM, vol. 23 #1, January, 1980.

[22] Phong, B. T. Illumination for Computer Generated Images. Communications of the ACM. vol. 18 #6, June, 1975.

[23] Blinn, J. Models of Light Reflection for Computer Synthesized Pictures. Computer Graphics, vol. 11 #2, summer, 1977.

[24] Whitted, T. An Improved Illumination Model for Shaded Display. Communications of the ACM, vol. 23 #6, June, 1980.

[25] Feeser, L. J. and Ewald, R. H. Dissemination of Graphics Oriented Software. *Structural Mechanics Computer Programs* Pilkey, W. D., editor. University Press of Virginia, Charlottesville, Virginia, 1974.

[26] Biffle, J. H. Interactive Graphics Aided Structural Analysis, Sandia Laboratories document SAND-74-5570, Sandia Laboratories, Albuquerque, New Mexico, 1974.

[27] Potts, J. For the Computer Gourmet - Graphics. *Structural Mechanics Computer Programs* Pilkey, W. D., editor. University Press of Virginia, Charlottesville, Virginia, 1974.

[28] Rodrigues, J. S. N., and Simon, N. Finite Element Program for Solution of Geotechnical Problems. Computer Aided Design, vol. 9 #3, July 1977.

[29] Bruch, J. C., Jr. The Use of Interactive Computer Graphics in the Conformal Mapping Area. Computers & Graphics, vol. 1 #4, 1975.

[30] Cozzolongo, J. V. Visualization of Three Dimensional Flow. Master's Thesis, University of Connecticut, 1979.

[31] Tinoco, E. N., McLean, J. D., Johnson, F., and Luckring, J. Free Flow Analysis using Panel Methods. Symposium on Computers in Aerodynamics, Polytechnic Institute of New York, June 1979.

[32] Baden Fuller, A. T. and dos Santos, M. L. X. Computer Generated Display of Three Dimensional Vector Fields. Computer Aided Design, vol. 12 #2, March 1980.

[33] Nessif, N. and Silvester, P. Graphic Representation of Three Component Vector Fields Computer Aided Design, vol. 12 #6, November, 1980.

[34] Belie, Capt. R. G., and Rapagnani, N. L. Color Computer Graphics--A Design and Analysis Potential Scarcely Tapped, Astronautics & Aeronautics, vol. 19 #6, June 1981.

[35] Wright, T. and Humbrecht, J. An Algorithm for the Plotting of Functions of Three Variables. Computer Graphics, vol. 13 #2, August 1979.

[36] Chang, C. Tornado Effects on Buildings and Structures. Proceedings of the Third International Conference of Wind Effects on Buildings and Structures. Tokyo, 1971.

[37] Forrest, A. On the Rendering of Surfaces. Computer Graphics, vol. 13 #2, August 1979.

[38] Brown, B. Computer Graphics for Large Scale Two- and Three-Dimensional Analysis of Complex Geometries. Computer Graphics, vol. 13 #2, August 1979.

[39] GRAFIC Ramtek User's Manual, Purdue University, School of Mechanical Engineering Compurer-Aided Design and Graphics Laboratory, January 1981.

[40] Joblove, G. and Greenberg, D. Color Spaces for Computer Graphics. Computer Graphics. vol. 12 #2, August 1978.

[41] Newman, W. and Sproull, R. Principles of Interactive Computer Graphics, second edition. McGraw-Hill, New York, 1979.

[42] Rogers, D. and Adams, J. Mathematical Elements for Computer Graphics McGraw-Hill, New York, 1972.

[43] Booth, K. Tutorial: Computer Graphics. The Institute of Electrical and Electronics Engineers, New York, 1979.

[44] Franta, W., and Maly, K. An Efficient Data Structure for the Simulation Event Set. Communications of the ACM, vol. 20 #8, August 1977.

[45] Chung, C. A System of Interactive Graphical Computer Programs for Multivariate Statistical Analysis for Geological Data. Proceedings of the 12th Symposium on the Interface of Computer Science and Statistics. 1979.

[46] Friedman, H. The Use of Graphics Software in Concert with Multivariate Statistical Tools for Interactive Data Analysis. Proceedings of the 12th Symposium on the Interface of Computer Science and Statistics, 1979.

GENERAL REFERENCES

[47] Carlson, H. W. Pressure Distributions at Mach 2.05 on a Series of Highly Swept Arrow Wings Employing Various Degrees of Twist and Camber. NASA Technical Note D-1264, May 1962.

[48] Giloi, W. Interactive Computer Graphics, Data Structures, Algorithms, Languages Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[49] James, G. and James, R., ed. Mathematics Dictionary D. Van Nostrand Company, Inc., Princeton, New Jersey, 1959.

[50] Kernigan, B. W., and Ritchie, D. M. The C Programming Language Prentice-Hall, Englewood Cliffs, New Jersey. 1978.

[51] Pope, A. Wind Tunnel Testing, 2nd ed. John Wiley and Sons, New York, 1954.

[52] Smith, A. R. "Color Gamut Transform Pairs," Computer Graphics, vol. 12 #3, August, 1978.

[53] Sutherland, I. E., Sproull, R. F., and Schumacker, R. A. "A Characterization of Ten Hidden-Surface Algorithms," Computer Surveys, vol. 6 #1, March, 1974.

[54] Thompson, K., and Ritchie, D. M. UNIX Programmer's Manual, Sixth Edition. Bell Telephone Laboratories, May, 1975.

[55] "Form 16 for 2000 Series User's Manual, ID Data Tablet/Digitizer," Summagraphics Corporation. Fairfield, Connecticut, 1977.

APPENDICES

Appendix A: Command Descriptions

Commands are titled by acronyms, which are used to label the Ramtek function keys. The same key may have different meanings in different command modes, as may be seen in Figure A27.

The following commands can be executed from the primary mode. Ramtek keyboard light 0 is lit while the user is in the primary command mode.

SCRC	Screen cycle Step through the cycle of available screen options. An on-screen message tells the user which screen area will be used for subsequent displays. The cycle is: Full screen : Top half screen : Bottom half screen
STYC	Style cycle Step through the cycle of available screen options. A short message tells the user what style is currently in force. Regardless of whether bivariate or three dimensional geometry displays are being created the cycle is: Wire frame : Isarithmic : Continuous tone
OPTS	Options cycle Step through the cycle of available screen options. The cycle is: Stars and element numbers : Element numbers : Stars and point numbers : Point numbers : Stars : No stars or numbering
PRSP	Perspective Toggle the use of perspective in subsequent displays.

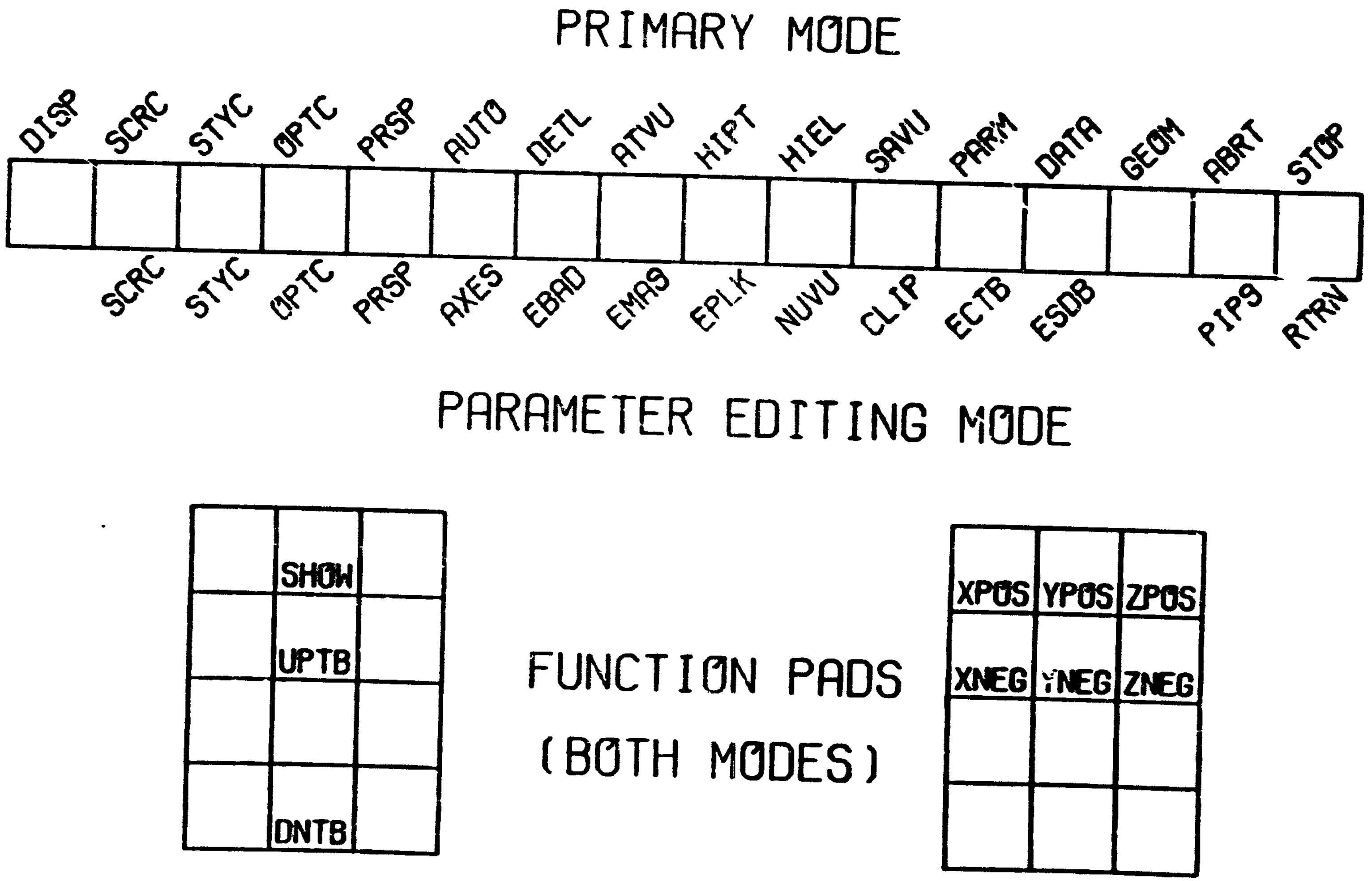


Figure A27

The Ramtek Function Keys

DISP	Display Display the geometry using the current set of display options and parameters. The associated data from the current data file are used as needed.
DETL	Detail Enlarge a detail of the last display. The user defines via the joystick driven cursor a rectangular portion of the latest display an area of particular interest. The user is cued for opposite corners of a rectangular screen area. He is to enter these by moving the cursor to a corner and toggling the joystick's enter switch. New scaling, focal point, and viewpoint parameters are calculated and the geometry is displayed as in DISP.
AUTO	Automatically scaled display Using the current viewpoint and focal point, calculate a new scale factor making a display of the geometry "fit nicely" in the display area. The geometry is then displayed as in DISP.
ATVU	Automatic viewpoint Choose an arbitrary viewpoint from which to view the geometry. Use AUTO for scaling, and display the geometry as in DISP.
STMP	Save viewing parameters, temporary Save the current geometry file and data file names along with all current viewing parameters in a temporary file.
GETP	Get temporarily stored viewing parameters Restore the viewing parameters, geometry file name, and data file name to the state extant when the STMP command was last executed.

SAVU	Save viewing parameters Save the current viewing parameters with the information in the current data file. The user is queried for the name of the new data file. The default file is the current data file.
DATA	Data file Access a new data file. The user is queried for the name of a new data file. The new file replaces the old data file in all subsequent displays. The user may elect to keep the old display parameters or to replace them by the parameters associated with the new data file.
GEOM	Geometry file Access a new geometry file. The user is queried for a new geometry file name. When the new geometry file has been chosen, a new data file must be selected as in the DATA command.
UPTB	Isarithmic color bar alteration If the current style is isarithmic and the cursor is on the color bar, the data range in which the cursor lies has a change in the color representing it. The new color used to represent the cursor chosen data range is the next higher color in the list of defined colors.
DNTB	Isarithmic color bar alteration The cursor chosen data range gets a new color as in UPTB. In DNTB the new color is the next lower color in the list of user defined colors.
XPOS,XNEG YPOS,YNEG ZPOS,ZNEG	Which axis is up Set the current vertical axis to whichever axis is indicated. This command is not relevant to bivariate displays.
PARM	Parameter editing mode Switch control of interaction to the parameter editing mode. On completion of parameter editing, control returns to the primary mode.
HIPT	Highlight points Initiate the highlighting of data points. Any

geometry point requested is numbered and starred on the latest display, and the associated data value is displayed in the upper left corner of the Ramtek screen. A flag is also displayed stating whether the data value is considered to be valid, considered to be bad, or has been massaged.

HIEL	Highlight elements
	Initiate the highlighting of elements of the geometry. The element is outlined and numbered on the last display, and the corner points are listed with their associated data values. If the plate is on the list of suppressed elements, this fact is mentioned.

The DISP, DETL, AUTO, ATVU, PARM, HIPT, and HIEL commands require the rolling into memory of completely new main programs. Information is passed to these new programs via temporary files that describe the current display options, files, and parameters. The programs executed because of the PARM, HIPT, and HIEL commands send altered display parameters back to the original (main) program via other temporary files.

The following commands can be executed from the parameter editing mode. Ramtek keyboard light 1 is lit while the user is in parameter editing mode.

SCRC	Screen cycle
	Functions as from primary mode.
STYC	Style cycle
	Functions as from primary mode.
OPTS	Options cycle
	Functions as from primary mode.

PRSP	Perspective Functions as from primary mode.
UPTB	Isarithmic color bar alteration Functions as from primary mode.
DNTB	Isarithmic color bar alteration Functions as from primary mode.
XPOS,XNEG YPOS,YNEG ZPOS,ZNEG	Which axis is up Functions as from primary mode.
AXES	Axes displayed Toggles the display of X, Y, and Z axes at the origin of subsequent displays.
PIPS	Pipes display Toggles the display of Z axis pipes from the Z=0 level to each point in the geometry. This is relevant only to bivariate displays.
NUVU	New viewing parameters Query the user for new viewing parameters including the viewpoint, focal point, and scale factor. If the user declines to enter values for any of the above, the current values of that parameter are kept.
CLIP	Clipping planes Ask the user if clipping planes are to be used. If the answer is yes, the user must define the clipping planes. If bivariate data are being displayed the user is asked about X and Y bounds instead of three dimensional clipping planes.
ECTB	Edit continuous tone color bar The user is queried for new parameters to define a continuous tone color bar and its relationship to the displayed data.
ESTB	Edit isarithmic color bar The user is allowed to edit the isarithmic color bar. Changes may be made in the colors defined,

	in the data range boundaries (the isarithms depicted), and in the correlation between colors and data ranges.
SHOW	Show the color bar Display the color bar of the current style as currently defined.
EBAD	Edit the list of "Bad" data values Edit the list of data values designated as "Bad." Points may be added to or deleted from the list.
EMAS	Edit the list of "Massaged" data values Edit the list of massaged data values. Points may be deleted from or added to the list. If a point is to be added to the list a new, or massaged, data value must also be entered.
EPLK	Edit suppressed plates list Edit the list of suppressed plates by adding and deleting entries.
RTRN	Return Return control of user interaction to the primary mode.

The following options are open to the user when highlighting points or elements.

1-N	Numerical entries Any number entered and terminated by a carriage return is taken as a request to highlight that point or element.
UPTB	Step up Highlight the next point or element as determined by the numbering of the points or elements.
DNTB	Step down Highlight the previous point or element as determined by the numbering of the points or elements.

<CR>	Carriage return A carriage return without a preceding number requests highlighting of the next or previous point or element. The direction of traversal is determined by the last UPTB or DNTB request.
EPLK	Edit suppressed plates list If elements are being highlighted, the currently highlighted element is added to or deleted from the list of suppressed elements.
EBAD	Edit the list of "Bad" data values If points are being highlighted, the currently highlighted point is added to or deleted from the list of bad points.
EMAS	Edit the list of "Massaged" data values If points are being highlighted, the currently highlighted point is added to or deleted from the list of massaged points. If the point is being added to the list the user is queried for a massaged data value.
RTRN	Return Return control of interaction to the primary mode.

Appendix B: Geometry File Format

Blanks and carriage returns are equally valid separators for numeric entries. Blanks may be embedded in the header lines or character strings.

Abbreviations

term	meaning
F	floating point
I	integer
C	character string
O	Nobjs controlled. The sequence of entries prefixed by "O" should occur Nobjs times.
G	Ngrps controlled. The sequence of entries prefixed by "G" should occur Ngrps times.
E	Elcnt controlled. The sequence of entries prefixed by "E" should occur Elcnt times.

Geometry file format

Header line one	(C)	Up to 20 characters
Header line two	(C)	Up to 20 characters
Header line three	(C)	Up to 20 characters
Header line four	(C)	Up to 20 characters
Npnts	(I)	The number of points in the geometry definition.
Coors	(I)	The number of significant coordinates for each point. If Coors = 3 the geometry is three dimensional. If Coors = 2 the geometry is for bivariate functions.

Nelmts				(I)	The number of plates or elements in the geometry definition. This includes all elements that may be in the display list, so those elements defined with four vertices must be counted twice. The software will break all four-vertex plates into two three-vertex plates.
Nedges				(I)	The number of edges in the geometry definition. Not used.
Nobjs				(I)	The number of objects in the geometry definition.
O	ObjID			(C)	A five character identifier for the current object. This is for the user's convenience in constructing and debugging geometries and is not used.
O					
O	Objnum			(I)	The object number, also inserted for user debugging only. Read but not used.
O					
O	Ngrps			(I)	The number of groups in the current object.
O					
O	G	GrpID		(C)	A five character identifier for the current group. This is for the user's convenience in constructing and debugging geometries and is not used.
O	G				
O	G	Grpnum		(I)	The group number, also inserted for user debugging only. Read but not used.
O	G				
O	G	Elcnt		(I)	The number of element definitions in the current group.
O	G				
O	G	E	Nsides	(I)	Either 3 or 4. The number of vertices of the element being defined.
O	G	E			
O	G	E	Vert	(I)	Nsides vertex indices. All Vert entries should be in the range from 1 to Npnts.
O	G				
O					

Appendix C: Data File Format

Blanks and carriage returns are equally valid separators for numeric entries. Blanks may be embedded in the header lines or character strings. Indented entries are conditionally present. A leading zero in an integer indicates that the number is octal.

Abbreviations

term	meaning
LOR	logical or
>	greater than
F	floating point
I	integer
C	character string
O	octal number

Data file format

Header line one	(C)	Up to 20 characters
Header line two	(C)	Up to 20 characters
Header line three	(C)	Up to 20 characters
Header line four	(C)	Up to 20 characters
Data entries	(F)	NPNTS entries (NPNTS read from geometry file)
Nbad	(I)	Number of data points flagged as "bad."
Bad	(I)	Nbad indices to geometry points flagging them as bad.
Nplkl	(I)	Number of elements in the suppression list.
Plkl	(I)	Nplkl indices to geometry elements to be suppressed.

Nmas	(I)	Number of data points flagged as "massaged."
Mas Dval	(I F)	Nmas pairs of entries. Mas is an index to a massaged data point. Dval is the value assigned to point Mas.
Disps	(O)	Octal flag indicating which of the following are present.
Eye	(F)	Present if (Disps LOR 010000) Floating point triple giving location of viewer in space.
Lookat	(F)	Present if (Disps LOR 010000) Floating point triple giving location of viewer's focal point.
Light	(F)	Present if (Disps LOR 010000) Floating point triple giving location of light source.
Scale	(F)	Present if (Disps LOR 010000) Scale factor to be used in display creation.
Clip1	(I)	Present if (Disps LOR 010000) Flag indicating use of clipping plane one.
Clip2	(I)	Present if (Disps LOR 010000) Flag indicating use of clipping plane two.
Clpat1	(F)	Present if (Clip1 > 0) Floating point triple giving the location of clipping plane one in space.
Clpnr2	(F)	Present if (Clip2 > 0) Floating point triple giving a vector normal to clipping plane two.
Clpat2	(F)	Present if (Clip2 > 0) Floating point triple giving the location of clipping plane two in space.
Clpnr2	(F)	Present if (Clip2 > 0) Floating point triple giving a vector normal to clipping plane two.

Ncolors	(I)	Present if (Disps LOR 0202) Number of colors present in the isarithmic color bar being described.
Nsects	(I)	Present if (Disps LOR 0202) Number of sections in the isarithmic color bar being described.
Udata & Col	(F I)	Present if (Disps LOR 0202) Nsects pairs. The first is the upper limit of the data range, and the second is the color with which that data range is associated.
Color	(F)	Present if (Disps LOR 0202) Ncolors entries giving the hues of the colors in the color bar. Red is 0.00, green is 0.33, blue is 0.66, and other colors fill the spectrum.
Ndlin	(I)	Present if (Disps LOR 02020) The number of data level labels beside the continuous tone color bar being described.
Lohue	(F)	Present if (Disps LOR 02020) The hue at the low end of the continuous tone representing data values below the limits of the spectral represenation.
Hihue	(F)	Present if (Disps LOR 02020) The hue at the high end of the continuous tone representing data values above the limits of the spectral representation.
Botsp	(F)	Present if (Disps LOR 02020) The hue at the bottom of the continuous tone spectrum.
Topsp	(F)	Present if (Disps LOR 02020) The hue at the top of the continuous tone spectrum.

Botdv	(F)	Present if (Disps LOR 02020 The lower limit of the data range represented by the continuous tone spectrum.
Topdv	(F)	Present if (Disps LOR 02020 The upper limit of the data range represented by the continuous tone spectrum.

VITA

VITA

Sheldon Lee Applegate was born on [redacted], the fourth of five children. After graduating from Frankfort Senior High School in 1969, he enrolled at Purdue University. He studied for one year at Warwick University in Coventry, England before returning to Purdue University to complete his B.S.M.E. in 1974 and his M.S.M.E. in 1975. After two years of employment as an engineer, he reentered the Purdue University School of Mechanical Engineering graduate program.