Some Recent Advances in Computer Graphics

Turner Whitted

Summary. Computer graphics, no longer a novelty, is on its way to becoming the preferred interface between humans and computers. Continued advances in the development of graphics algorithms are improving the quality and usefulness of this interface. Decreasing hardware costs have enabled an expansion of graphics into more diverse applications.

Computer users are discovering that graphical interaction is a more comfortable and friendly interface between human beings and computers than punch cards or keyboards and tabular printout. The economic consequences of this fact have made computer graphics a subject of serious study. Improvements in graphics algorithms along with a steady reduction in the cost of hardware are leading to an expansion of computer graphics into applications including design, instruction, medicine, communications, business, and entertainment as well as scientific research. This article, after a brief introduction to general principles, describes recent advances in computer graphics software and hardware and surveys some of the more interesting application areas. The reader should note that computer graphics is just one aspect of picture processing and that advances of the same magnitude have occurred in related fields such as pattern recognition, image enhancement, and image coding (*1*).

Digitally generated graphical display is nearly as old as digital computing itself. The Whirlwind I computer, which first became operational in 1950, had a facility for plotting points on a cathode-ray tube (CRT) while automatically photographing the resulting graph (*2*). Users were encouraged to use this facility not only because it saved precious machine cycles that would otherwise be wasted by printing verbose tabular results, but also because a graphical presentation was a more sensible format for the results of most scientific computing.

The birth of interactive computer graphics can be traced to Sutherland's SKETCHPAD system in 1963 (*3*). SKETCHPAD was the first good example of a system that permitted humans to communicate in a natural and efficient manner with a computer. The ideas behind it were soon extended to three dimensions, allowing designers to build and manipulate solid structures interactively (*4*). In the middle 1960's, aircraft and automobile manufacturers developed large interactive systems that made computer graphics an indispensable tool for the design process (*5*). Computer-aided design (CAD), because of its promise as a productivity enhancer, remains a vital research topic (*6*).

Beginning in the late 1960's researchers began to develop algorithms to generate shaded images of solid objects. These techniques for producing completely synthetic images are rapidly moving toward the capability to produce pictures which cannot be readily distinguished from pictures of real scenes. Making the image generation procedures efficient presently poses one of the greatest challenges to researchers in computer graphics.

Although three-dimensional computer graphics has attracted a great deal of attention, most applications in business, communications, and entertainment rely primarily on two-dimensional graphics techniques. In contrast to three-dimensional graphics, whose complexity is something of a barrier to its widespread use, hardware and software for two-dimensional graphics are readily available and easy to use.

General Principles

A desk-top computer connected to an inexpensive paper plotter is a simple graphics system that is useful for plotting the results of one- or two-dimensional simulation, for plotting pie charts or bar graphs, for creating plots that summarize an experiment, or for other noninteractive uses. The simplest interactive computer graphics system consists of only a host computer equipped with a graphical input device and a graphical display device that can be rapidly updated. The rapid update requirement currently limits interactive applications to CRT's or plasma panels. CRT's are by far the most widely used.

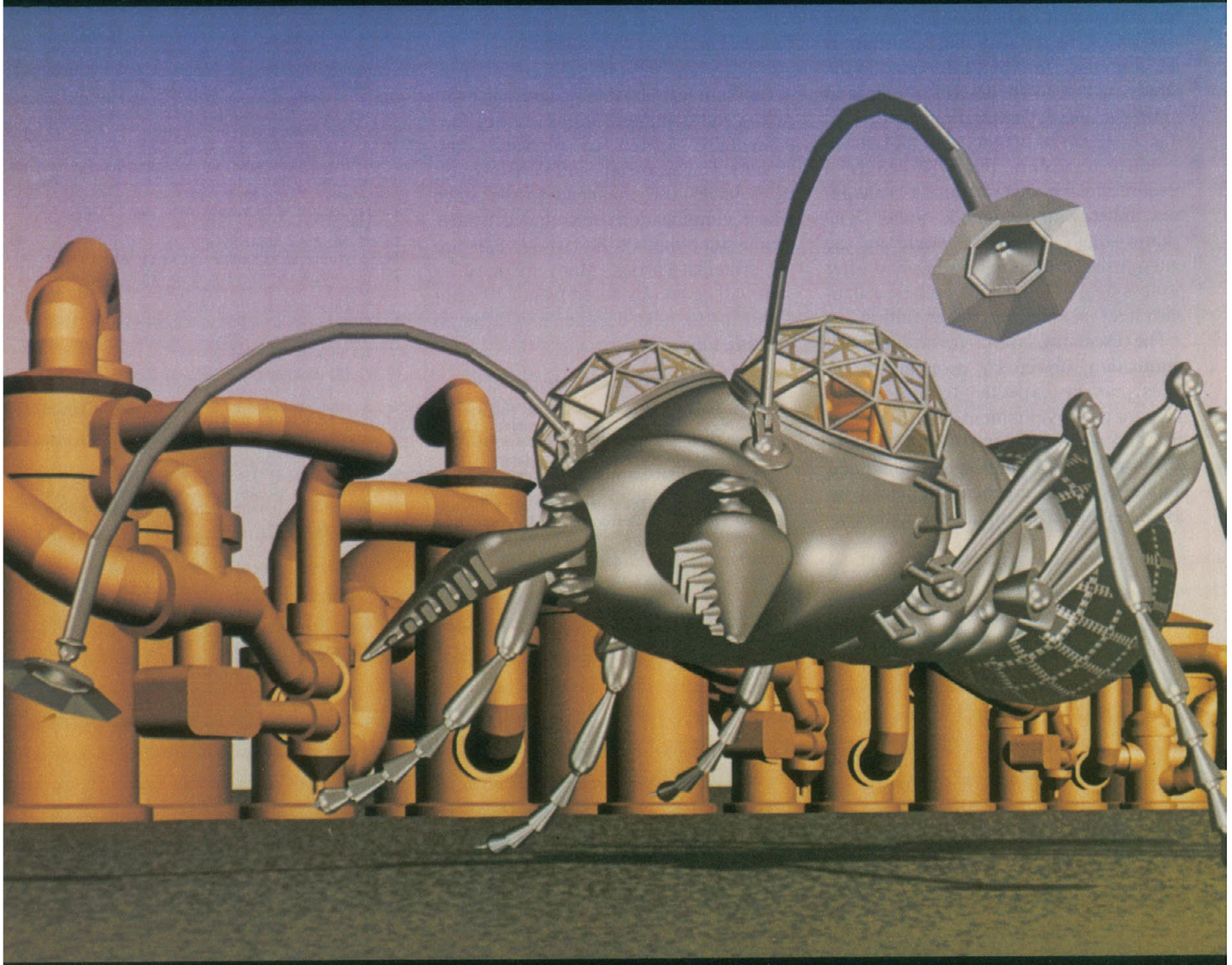
Unless scanned in from a camera or manually entered by an artist, the drawings that appear on a display screen are generated from a mathematical representation, stored in the memory of the host computer, that defines the objects shown on the display. The highest level of representation is called an object description. For line drawing display systems, this object description is translated into a display list, a set of lower level drawing commands that is interpreted by the display hardware. A very simple representation, which serves as both object description and display list, is a list of line segment endpoints for a line drawing display. Each endpoint consists of a two- or three-dimensional vector defining its location. (In addition to its conventional definition as an n-tuple of coordinate values, the term vector is often used in the graphics literature as a synonym for stroke or line segment. To avoid confusion, this article adheres to the conventional definition.)

Useful graphical interaction requires a facility for moving a picture or parts of it in ways specified by the user. The designer of a three-dimensional machine part, for example, needs to view the object from many angles to ascertain its shape. The motions can be broken down into the primitive operations of rotation, translation, and scaling, which are implemented by matrix multiplication, vector addition, and scalar multiplication, respectively. For realistic viewing of three-dimensional objects, a perspective transformation is usually concatenated with the other transformations. A more convenient method of representing positions in three-dimensional space is the use of homogeneous coordinates (*7*), which represent each point with four coordinate values. The use of homogeneous coordinates permits all transformation

The author is a member of the technical staff at Bell Laboratories, Holmdel, New Jersey 07733.

0036-8075/82/0212-0767$01.00/0

Graphics and Software



operations, including the perspective viewing transform, to be incorporated into a single matrix multiplication. The translation and scaling operations permit a viewer to magnify selected regions of an object, but the remainder of the object may be transformed to positions that lie outside the boundaries of the viewing screen. These outlying portions of the object are removed from the display list by another primitive display operation called clipping. Some high-performance display systems perform the transformation and clipping functions in special-purpose hardware.

Display Hardware

Aside from paper, the most common display media for computer graphics have been directed beam CRT's, direct view storage CRT's, and raster scan CRT's (*8*). All CRT's display information by directing an electron beam of controlled intensity at a phosphor-coated screen, which fluoresces with a brightness that is a function of the beam energy. For directed beam and storage CRT's, which are classified as calligraphic devices, the electron beam can be steered in any direction across the screen (Fig. 1). For the raster scan CRT the electron beam is swept across the screen in a fixed area-filling pattern composed of horizontal scan lines as illustrated in Fig. 2. Calligraphic displays are intended for line drawing, while raster scan CRT's are suitable for displaying solid shaded areas on the screen. The common household television CRT is of the raster scan variety.

The storage CRT is a special type of directed beam CRT which, as its name implies, retains brightness levels within a special structure on the face of the tube with no need for external refresh circuitry. The conventional directed beam tube has no special structure on its face other than a phosphor screen. Consequently, lines that appear on the screen must constantly be refreshed to remain visible. To provide a refresh rate that is sufficiently high to avoid flicker, the electronics used to deflect the beam must have high speed and power capabilities and are expensive as a result. However, with the screen constantly being redrawn, there is no penalty for changing the drawing from one refresh period to the next. This ability to provide a dynamically changing image makes the directed beam CRT a popular choice for high-performance interactive applications. In addition to the CRT, its power supply, and beam deflection electronics, commercial display systems contain circuits to perform many of the low-level graphical operations. For example, rather than have the host computer address and brighten points on the display screen to produce lines, almost all calligraphic display systems include a hardware stroke generator that can intensify the entire length of a line segment if given only the endpoint locations of the line. As mentioned previously, some also include transformation and clipping processors.

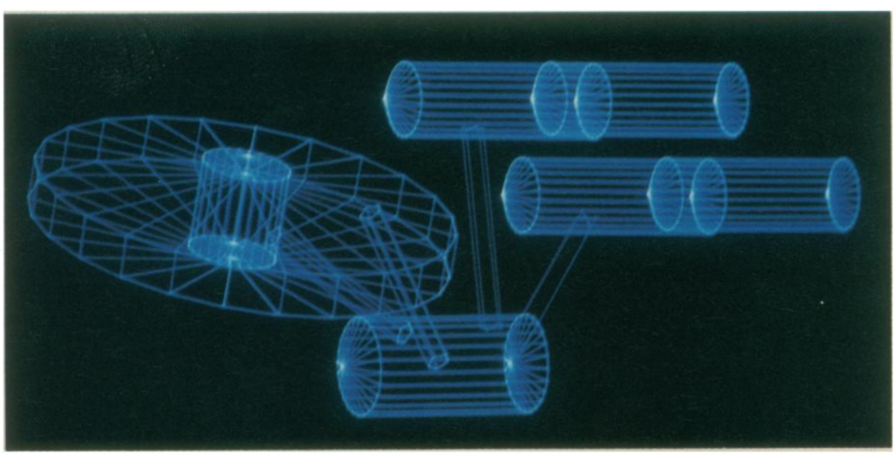


Fig. 1. Complex assembly displayed on a calligraphic cathode-ray tube. [J. F. Eastman, Hewlett-Packard]

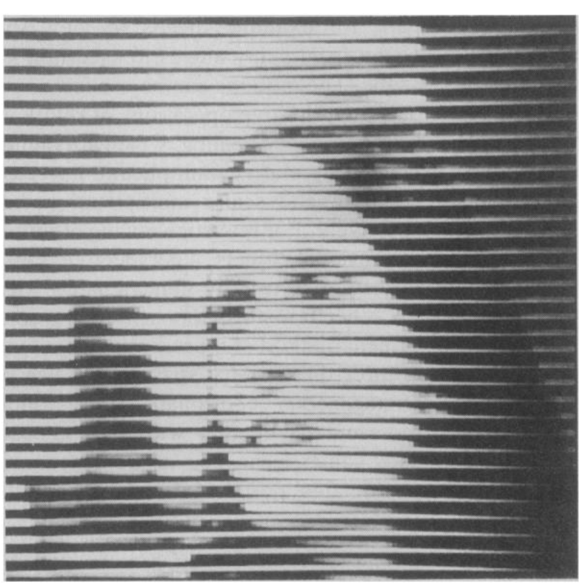


Fig. 2. A raster pattern is formed by repetitively sweeping an electron beam from left to right across the screen of a cathode-ray tube. For the return sweep from right to left, the electron beam is turned off. Images are formed on the screen by varying the beam energy during the visible sweep.

Borrowing from mass-market television technology, raster scan tubes are the least expensive type of CRT. Furthermore, the ability to paint filled areas on a screen gives raster display an advantage in applications where line drawings do not serve well. Like the directed beam CRT, the raster scan tube must also be refreshed continuously from a memory. However, the refresh data consist not of line segment endpoints but of intensity values corresponding to each point of the video signal along a scan line. If each scan line is represented by 512 sample points, called picture elements or pixels, and 512 scan lines are displayed, a resolution that corresponds roughly to the format of broadcast video, then 262,144 pixels must be stored in the refresh memory. While the memory of a general-purpose computer can sometimes be used to store refresh data (*9*), the overhead of transferring 1/4 million pixels to the CRT every 1/30 second is beyond the capacity of most computer systems. However, the low cost of semiconductor memory makes it practical to build special-purpose memories, called frame buffers, for the specific purpose of refreshing raster scan CRT's (*10*). A frame buffer has two access ports—one for the host computer to read and write and a second one read by the video refresh circuits. Most frame buffers also include processing elements to draw text and elementary shapes such as lines, rectangles, and circles. Since raster processing steps vary widely between one application and another, fixed-function, special-purpose display hardware has a limited utility. One economical approach to display system design now being offered in raster equipment is to dedicate a high-performance, user-programmable processor to the display function. The user is then able to transfer his own unique procedures from the host to the peripheral processor.

The most common way to display raster scan images in color is to use a shadow mask type of CRT whose screen is covered by an array of three types of phosphor dots that emit red, green, and blue light, respectively. Three electron beams instead of one are directed at the screen through the shadow mask, a metal sheet with thousands of carefully placed openings located between the screen and the source of the electrons. The electron beams, the shadow mask, and the phosphors are all precisely aligned so that each electron beam controls only one of the three primary colors. Any other colors are represented by a mixture of red, green, and blue. Color pictures can be produced on a calligraphic display with a beam penetration–type tube whose screen is coated with two layers of phosphors that emit different colors. The color emitted from the screen is then a function of the electron beam energy.

Each of the three CRT technologies is being constantly improved, and as a result many of the original drawbacks of each type have been overcome. For instance, storage tubes are now available which simultaneously display both static stored vectors and refreshed vectors which can be dynamically changed. Frame buffers, previously regarded as strictly a medium for static image display, are being used increasingly for display of moving images. There are also new attempts to apply less expensive raster displays to line drawing applications, but the superior resolution of directed beam displays helps to maintain their dominance of this market. The inherent two-dimensionality of the display screen is a handicap of all CRT's. True three-dimensional display can be provided by stereo viewing (*11*) or the use of a vibrating curved mirror whose changing focal length gives apparent depth to the reflection of an image on a CRT (*12*).

Geometric Modeling

Geometric modeling for two-dimensional graphics is relatively simple. A straight line can be defined by a vector at each endpoint and a circle by the location of its center and the length of its radius. A more complicated curved line may be represented by the coefficients of its equation. For three-dimensional graphics these curves generalize nicely into polygons, spheres, and more complicated surfaces. However, applications for which complex three-dimensional objects are to be assembled and interactively manipulated call for specialized geometric representations. Almost all of them fall into the category of either constructive solid geometry representations, which describe an object as a solid volume, or boundary representations, which define an object in terms of its surface (*13*).

The modeling approach for constructive solid geometry is to build complex objects from algebraic combinations of simpler objects. Solid volumes are defined by the union, intersection, or difference of elementary solids such as cylinders, spheres, boxes, and so on. For instance, a cylindrical rod with several holes drilled through it can be described as a cylinder minus its intersection with several smaller cylinders, where each of the smaller cylinders represents a drill hole.

Boundary representations are available in several varieties. The mechanism shown in Fig. 3, for example, contains both polygonal and cylindrical faces. For

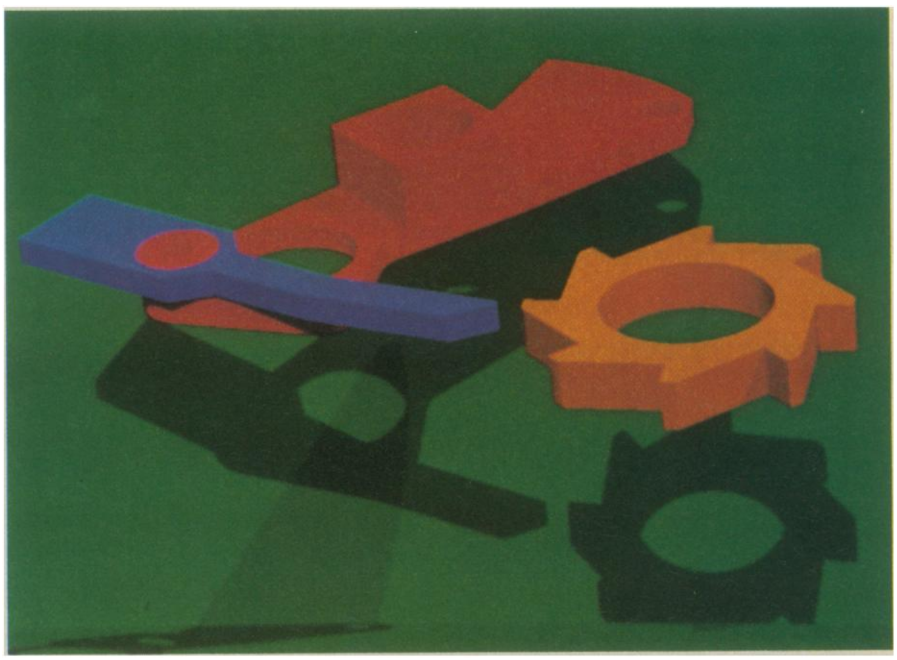


Fig. 3. The three components of this escapement mechanism were designed by using a boundary representation type of solid modeling system. [J. Davis and M. Bailey, Department of Mechanical Engineering, Purdue University]

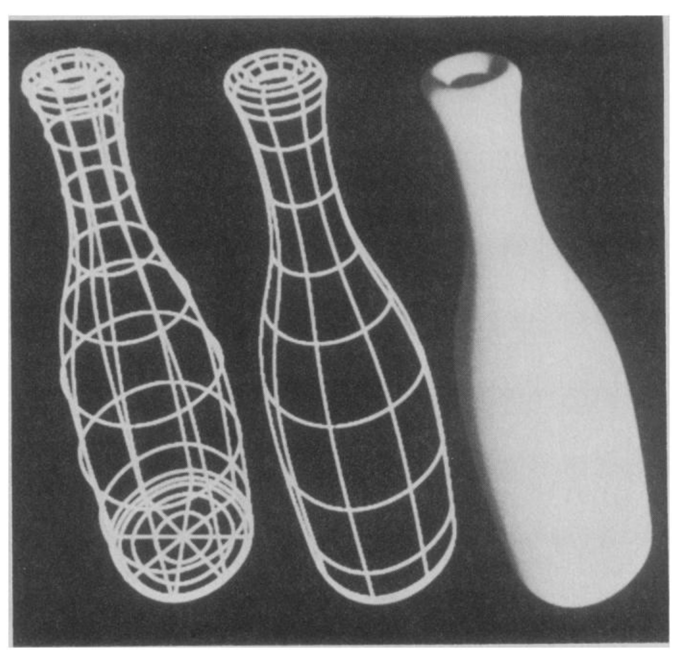


Fig. 4. Object shown without hidden lines removed, with hidden lines removed, and shaded with hidden surfaces removed.

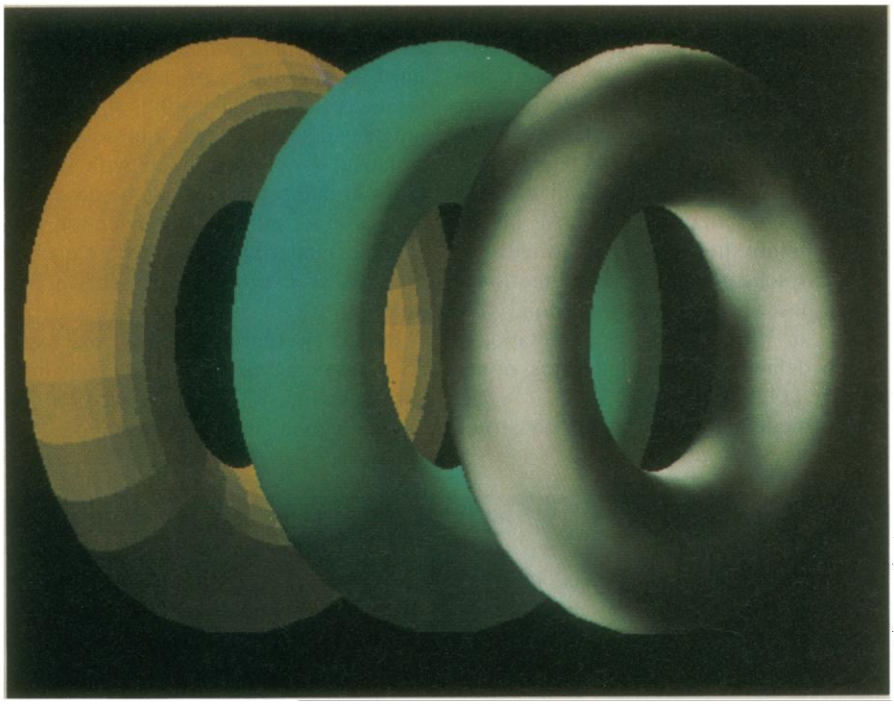


Fig. 5. Torus approximated by 841 polygons shown three times with flat shading on the left, Gouraud smooth shading in the center, and Phong shading on the right.

the design of automobile bodies, ship hulls, the exterior surfaces of aircraft, and other applications where smoothness is important, free-form sculpted surfaces serve well. These are typically piecewise polynomial functions of two variables which are formed by manipulating the surface equation coefficients (*14*). They have become popular because techniques exist for specifying the coefficient values in a natural and intuitive way and because the surfaces exhibit desirable properties that can be easily controlled.

As an alternative to describing objects with long lists of numerical data, procedural models represent objects by a short list of data to specify the attributes of the object and a pointer to a computer procedure that knows how to construct the object (*15*). Objects with symmetry, such as those bounded by surfaces of revolution, or objects formed by extrusion, can be represented compactly. Another example is a fractal curve, a mathematical construction whose statistical properties

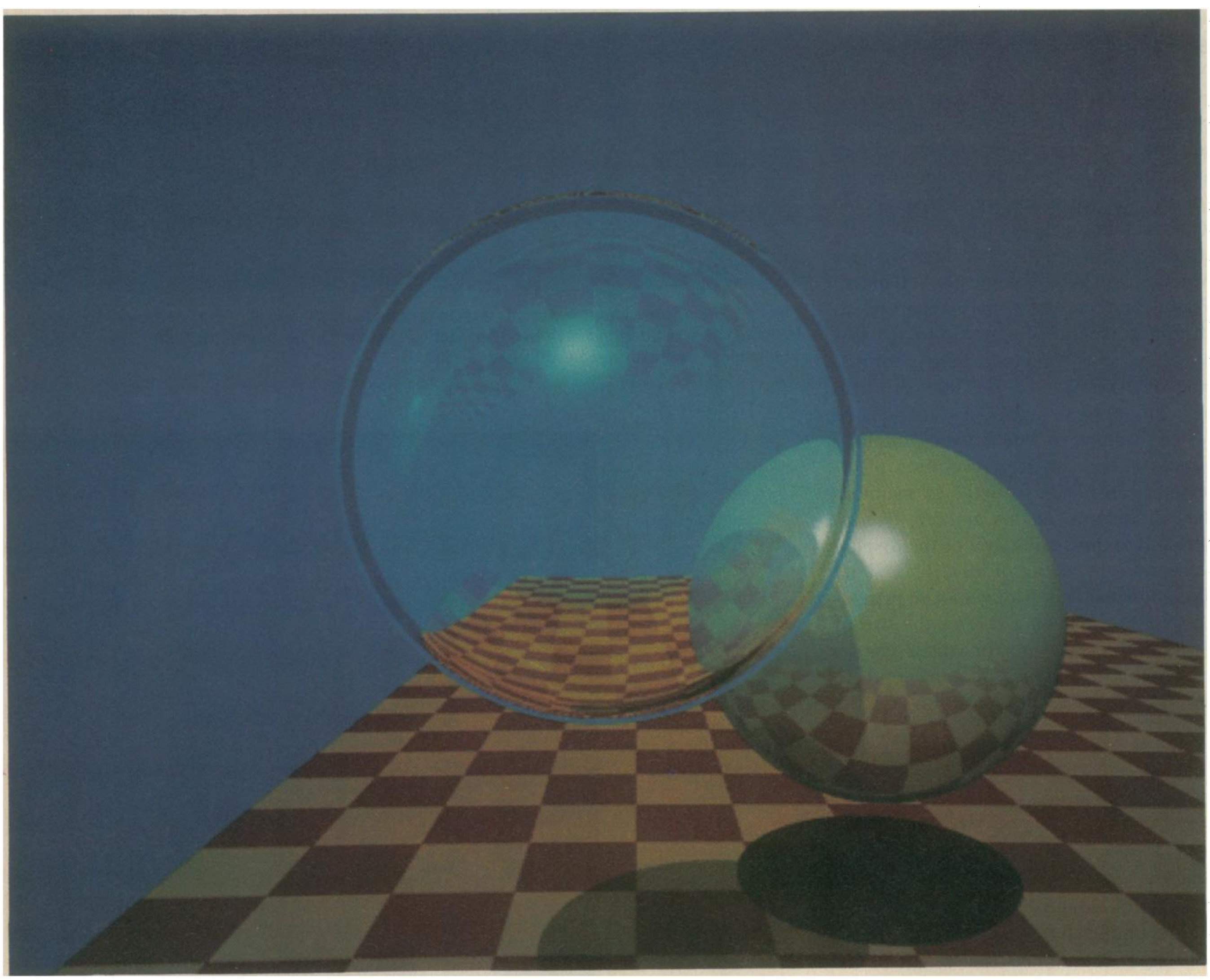


Fig. 6. The shading model used to produce this picture incorporates reflections between objects, shadows, and transparency with refraction.

are similar to those of objects found in nature (*16*). Procedures for producing fractal surfaces suitable for use in three-dimensional display applications yield pictures with remarkably rich detail, as illustrated in the cover picture (*17*).

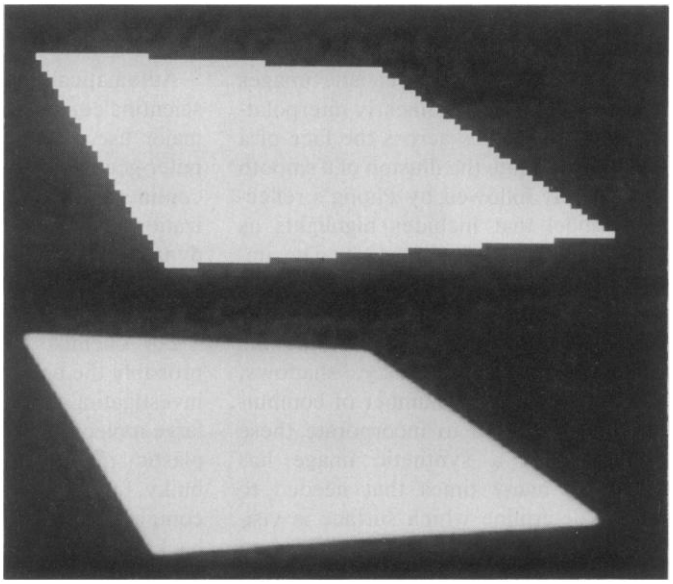


Fig. 7. Polygon rendering with and without anti-aliasing.

Algorithms

A collection of graphics hardware connected to a computer is not very useful without the support of a set of software tools for user interaction and display. Graphical user input to a computer is accomplished through operations performed by one or more input devices. A tablet-type input device transfers to the host computer a sequence of coordinate values reflecting the current position of the pen on the tablet, while a joy stick generates data representing the rate of change of position and a mouse provides a measure of incremental change of position. In their typical use, these types of devices are classified as locators. The other three classes of input device are the pick, a device for selection, such as a lightpen; the button, such as a key on a keyboard; and the valuator, a device for input of numerical values (*18*). This classification is based on a device's use rather than its specific characteristics. In many applications a single tablet performs all four classes of functions. Computers interpret this low-level graphical input through languages similar to those used for textual input. Design of graphical interaction languages is recognized as one of the most difficult problems in graphics, partly because of the need to account for human behavior in the software and partly because of the need to create a unique language structure for each application. There is an enormous amount of research yet to be done in this area.

For graphical display, well-known techniques exist for transformation and clipping, line and curve drawing, two-dimensional polygon filling, and other similar operations. Yet interest in these operations is being revived by the prospect of implementing them in very large scale integrated (VLSI) circuits. Algorithms that might not be practical for general-purpose computers can make good use of the parallelism available in VLSI implementations (*19*).

Another area of active research on display algorithms is the synthesis of realistic pictures of three-dimensional scenes. This activity began with the development of algorithms for hidden line removal. With the advent of shaded display on raster devices, attention turned to hidden surface elimination. Visible surface algorithms, sometimes called hidden surface algorithms, determine the correspondence between points on a three-dimensional surface and points on an image plane (the plane that represents the two-dimensional viewing screen), determine visibility at each point on the image plane, and determine an intensity value for each visible point (*20*). The advantage of a shaded display is its superior ability to give the viewer a sense of shape and depth, as illustrated in Fig. 4. Originally, hidden surface algorithms were applicable only to polygonal surfaces. Recent efforts have yielded algorithms for directly rendering higher order surfaces (*21*) without resorting to an a priori approximation by polygons.

Intensity values for shaded pictures are a function of surface orientation, light source direction, and surface properties. The simplest shading model is based on Lambert's law, with the reflected intensity proportional to the cosine of the angle between the surface normal and the light source direction. If a mesh of polygons is used to approximate a curved surface, the simplest shading technique will yield a faceted appearance

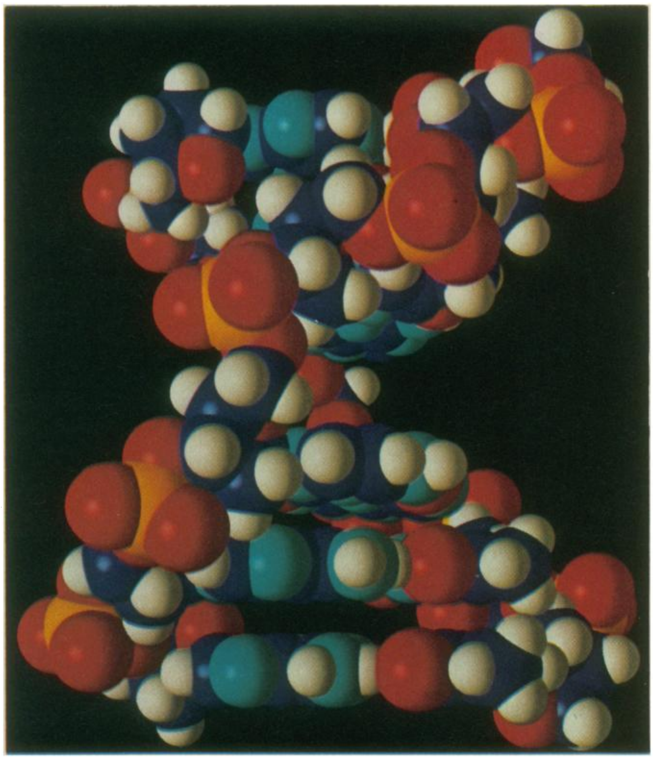


Fig. 8. This high-quality image of six base pairs of a DNA molecule is a single frame from an animated sequence. [N. Max, Computer Graphics Group, Lawrence Livermore National Laboratory]

since the surface normal is constant across the face of each polygon. Two advances toward more realistic images are Gouraud's idea of linearly interpolating intensity values across the face of a polygon to create the illusion of a smooth surface (*22*) followed by Phong's reflection model that includes highlights as well as diffuse reflection (*23*). The improved appearance produced by these models is evident in Fig. 5. Models exist for other lighting effects such as the simulation of transparency, shadows, and texture, but the number of computing cycles needed to incorporate these features into a synthetic image has grown to many times that needed to simply determine which surface is visible. Figure 6, for example, required more than 1 hour to compute on a large minicomputer (*24*).

Because images displayed on a raster device exist only as discrete sample points rather than a continuous signal, they are subject to a disturbing artifact known as aliasing (*25*). The most common form of this artifact is the jagged appearance of the boundaries of shaded regions. The effects of aliasing can be reduced (not eliminated) by using displays of higher resolution, but a more effective approach is to blend intensities along region boundaries in such a way that the jagged lines appear smooth (Fig. 7). However, calculation of the blended intensity values, a process called antialiasing, is not a trivial task.

Scientific Applications

Automatically plotting the results of scientific computation has always been a major use of computer graphics. Computer-generated animation in particular continues to provide an effective visualization tool, allowing researchers to view dynamic simulations of chemical, physical, mechanical, or strictly mathematical constructions (*26*).

For chemists, computer graphics is probably the only reasonable method for investigating the interaction between large molecules. Whereas the traditional plastic model of molecules may be bulky, fragile, and difficult to modify, the computer model has none of these drawbacks (*27*). For interactive use, chemists must rely on wire frame display, which can be manipulated in real time. If line segments in the display represent chemical bonds, the user is able to view and modify bond angles, bond length, and placement. A chemist may synthesize a new molecule graphically before synthesizing it chemically, or may interactively construct an existing molecule whose properties are known but whose structure is not well understood. Shaded pictures like Fig. 8 provide additional information by displaying the atomic surfaces and by providing distinct colors for different types of atoms (*28*).

At the other end of nature's scale, computer animation applied to astronomy permits several million years of galactic interactions to be compressed into a few seconds. Astronomers can test models of galactic evolution and interaction by comparing a computer-generated view of model galaxies to the static views seen through their telescopes. Within our own solar system, Blinn's realistic animation of Voyager spacecraft maneuvers for the Jupiter and Saturn flybys, shown to the general public on network newscasts, is a NASA mission planning aid as well as a great public relations product (*29*).

Applications in Computer-Aided Design

Computer-aided design is dominated by the two areas of electronic circuit design and geometric design. Both areas depend heavily on computer graphics.

Color two-dimensional graphics for integrated circuit layout is especially important since symbols are not always used to designate circuit elements. Instead, designers often work directly with color-coded geometric representations of the mask layers themselves. Line drawing representations or black and white shaded drawings used for interactive layout are ambiguous and therefore prone to misinterpretation. The advantage of color is evident in the example shown in Fig. 9, which is taken directly from the CRT screen of an interactive design station (*30*).

For three-dimensional geometric design, computers and computer graphics provide an interactive input medium using CRT-based design stations, a bookkeeping system using the computer's bulk memory, and an automatic drafting facility using paper plotters. Considering the size and complexity of such large projects as nuclear power plants, it is hard to imagine a design system without a computer to keep track of the parts. The once painful process of modifying an architectural drawing is simply a matter of entering a design change into the computer, and the tedium of drafting separate floor plans for several nearly identical floors of a skyscraper is reduced to entering only the differences between the plans (*31*).

Geometric CAD relies mainly on interactive line drawing displays, but when aesthetic considerations are important, only shaded renderings are acceptable. The high cost of interactive image generation hardware has prevented its use for CAD, and software implementations of shaded display algorithms are much too slow for interactive use. One possibility considered by some manufacturers is a dual set of displays: an interactive line drawing system alongside a noninterac-

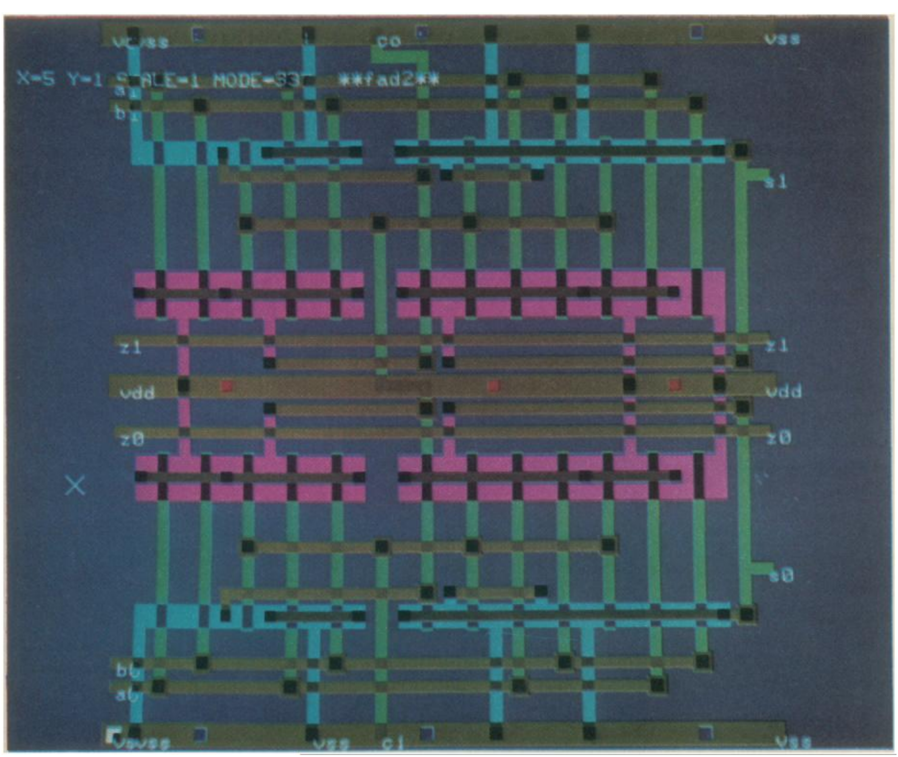


Fig. 9. In this integrated circuit layout different colors correspond to individual mask layers. [B. Ackland and N. Weste, Bell Laboratories]

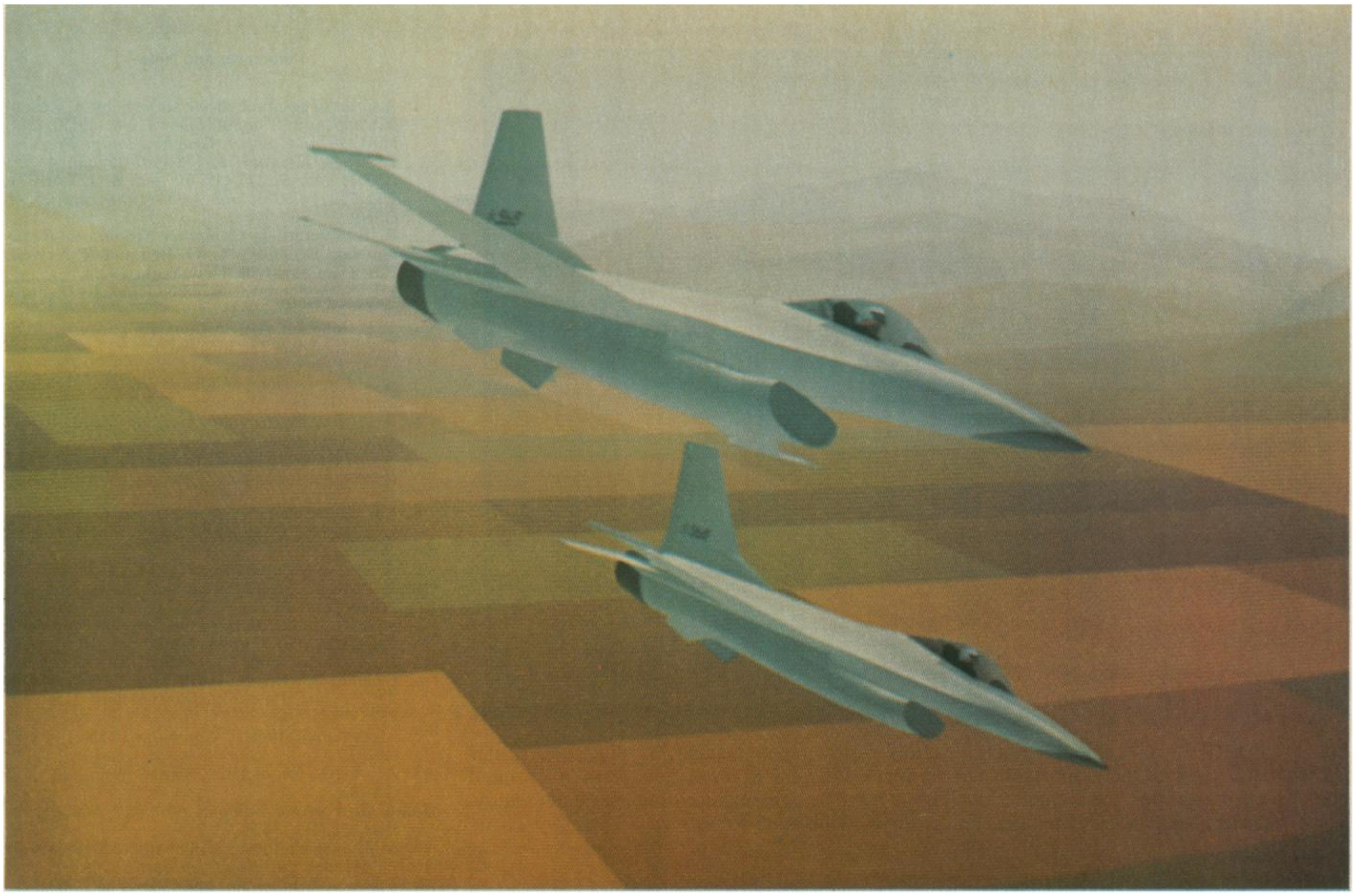


Fig. 10. One frame of a sequence produced in real time on the Evans and Sutherland CT-5 computer image generation processor for use in flight simulators. [Evans and Sutherland Computer Company]

tive raster display. A designer would build a three-dimensional structure by using the line drawing display and pause to see what it looks like on the raster display.

Applications in Flight Training

Flight training in real airplanes is not only dangerous and expensive, but in such cases as training astronauts to land on the moon it is impossible. The role of computer graphics in ground-based flight simulators is to provide to a pilot an image that closely mimics the view seen from his cockpit window in an actual airplane. Image generation for nighttime flying is a relatively easy task since the points of light seen from the air at night can be drawn in real time by a calligraphic display. For daytime simulators the problem is much more difficult because of the need to generate shaded images of complex scenes containing mountains, buildings, and other ground-based objects in full color at the rate of one image every 1/30 second (*32*). The only answer to such stringent requirements is the use of special-purpose hardware for image generation. Although features such as realistic display, freedom from artifacts, and real-time performance are all goals, real-time performance is a hard and fast constraint and the other features must be compatible with it or suffer some degradation. Most computer image generation hardware attempts to minimize the effects of the aliasing artifact since it is most detrimental to the effectiveness of training in the simulator. Visual effects that add to the realism of the simulator images are haze (shown in Fig. 10), fog, and even skid marks on runways.

Applications in Communications and Business

Computer terminals are evolving into a new generation of intelligent terminals and personal computers, most of which support the input and display of graphics as well as text. The forerunner of this trend was the Xerox Alto, a personal minicomputer that included a mouse for graphical input and a high-resolution raster display (*33*). With the rapid growth of both home computing and office automation one would anticipate that two-dimensional computer graphics will become as common as typewriting.

In order to reach a still wider market, the communications industry is experimenting with systems to display text and graphics on home television receivers (*34*). In the early 1970's the British Broadcasting Corporation introduced teletext, a system for transmitting text and graphics in an unused portion of the television signal. A decoder built into the television receiver created the display. Following that, the British Post Office introduced viewdata (known as videotex in the rest of the world), which uses a decoder to connect the television receiver to telephone lines, providing a service similar to teletext with two-way capabilities. Because each of these systems transmits an encoded version of the image itself, the image quality is severely limited by the low bandwidth of the transmission channel.

The Canadian videotex standard, Telidon, promises higher quality graphics by placing extra processing power in the terminal decoder. The decoding processor interprets a display list that requires much less bandwidth than an encoded image to transmit. However, before flexible graphics systems such as Telidon become common in homes, terminal processors must become far less expensive than they are today.

Computer graphics is becoming an essential part of business communications. Instead of armies of artists manually producing slides, charts, and overhead transparencies, much of this work is now

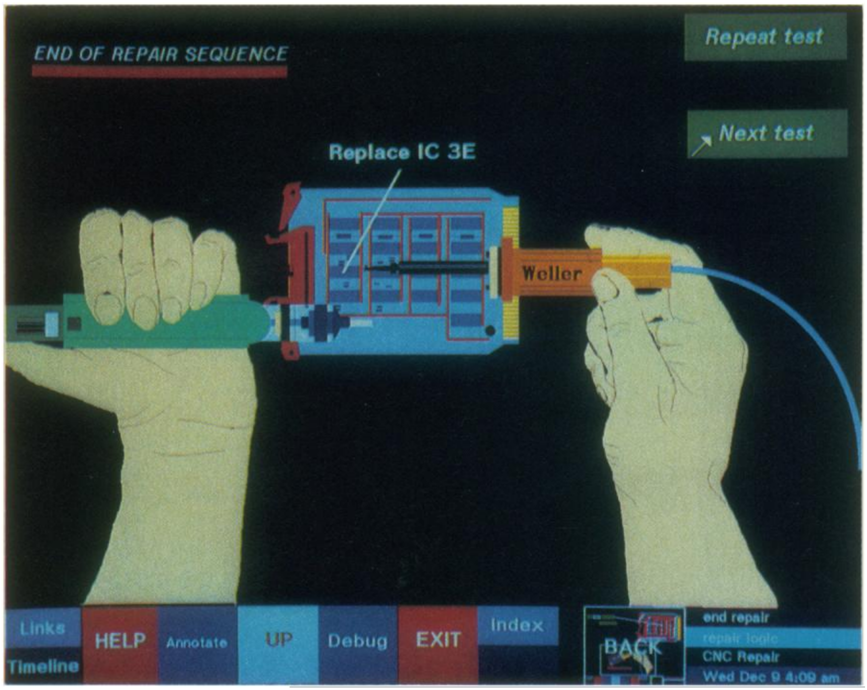


Fig. 11. Page from a "computer book," in this case a repair manual, in which the reader interacts directly with the graphics display terminal. Whereas pages and chapters in a printed book have a fixed order, the computer book has a more flexible organization. [S. Feiner and A. Van Dam, Department of Computer Science, Brown University]

being done automatically or with computer assistance. A natural extension to this trend is to bypass the printed or photographic media and bring the graphical information directly to the reader via CRT-based graphics terminals, as shown in Fig. 11.

Applications in Art and Entertainment

Artistic applications employ both two- and three-dimensional computer graphics techniques. Several commercial studios and special effects shops are attempting to create economical systems for computer-assisted conventional two-dimensional animation of anything from Saturday morning cartoons to feature-length films. The combination of an input tablet and a frame buffer is used as a paintbrush and canvas. By drawing into computer memory rather than onto paper, the path is opened for automating many of the more tedious parts of conventional animation, although formidable problems have been encountered in trying to impart the animator's skill to software (*35*).

The artwork used for television advertising is probably the form of graphics most familiar to the layman. Although most of the computer-generated three-dimensional effects used in advertising combine rudimentary computer graphics with a heavy dose of conventional optical techniques, better equipment and algorithms are bringing more of the image generation process into the digital domain (*36*). This trend demands the use of techniques for realistic display on the order of that illustrated in the picture on page 766.

New Directions

Flight simulator, entertainment, and, to a lesser extent, CAD applications need more efficient methods to produce realistic images. The lack of a real-time constraint in entertainment applications should permit an approach with fewer compromises in image quality. All applications involving raster displays need more efficient remedies for the aliasing problem. In each case inexpensive processing power provided by VLSI is expected to be the key to meeting these needs. This is a peculiar case of technological "tail chasing," since advances in VLSI are closely tied to better graphical design aids (*37*).

An inevitable effect of lower hardware costs is the wider availability of graphics systems, but their acceptance by nontechnical users depends on improved user interfaces. One is prompted to look for computer graphics in unexpected places within the home, office, and laboratory. More specific predictions are not likely to be fruitful. After all, 10 years ago who would have expected to find a computer as part of a washing machine?

References and Notes

1. A series of comprehensive surveys by A. Rosenfeld appear in *Comput. Surv.* **1**, 146 (1969); *ibid.* **5**, 81 (1973); *Comput. Graphics Image Process.* **1**, 394 (1972); *ibid.* **3**, 178 (1974); *ibid.* **4**, 133 (1975); *ibid.* **5**, 215 (1976); *ibid.* **6**, 157 (1977); *ibid.* **7**, 211 (1978); *ibid.* **9**, 354 (1979); *ibid.* **13**, 46 (1980).
2. R. E. Everett, in *Computer Structures: Readings and Examples*, C. G. Bell and A. Newell, Eds. (McGraw-Hill, New York, 1971).
3. I. E. Sutherland, thesis, Massachusetts Institute of Technology (1963).
4. T. E. Johnson, *AFIPS Conf. Proc. Spring Joint Comput. Conf.* **23**, 347 (1963).
5. D. M. Prince, *Interactive Graphics for Computer-Aided Design* (Addison-Wesley, Reading, Mass., 1971).
6. B. A. Barsky, *Comput. Graphics Appl.* **1** (No. 3), 67 (1981).
7. W. M. Newman and R. F. Sproull, *Principles of Interactive Computer Graphics* (McGraw-Hill, New York, ed. 2, 1979).
8. I. E. Sutherland, *Sci. Am.* **222** (No. 6), 56 (1970); L. C. Hobbs, *Comput. Graphics Appl.* **1** (No. 1), 25 (1981).
9. A. M. Noll, *Commun. ACM* **14**, 143 (1971).
10. J. T. Kajiya, I. E. Sutherland, E. C. Cheadle, *Proceedings of the IEEE Conference on Computer Graphics, Pattern Recognition, and Data Structures* (IEEE Press, New York, 1975), p. 1.
11. J. A. Roese and L. E. McCleary, *Comput. Graphics* **13** (No. 2), 41 (1979); I. E. Sutherland, *AFIPS Conf. Proc. Fall Joint Comput. Conf.* **33**, 757 (1968).
12. "New applications," *Computer* **10** (No. 11), 87 (1977).
13. A. A. G. Requicha, *Comput. Surv.* **12**, 437 (1980).
14. A. R. Forrest, *Comput. Graphics Image Process.* **1**, 341 (1972).
15. M. E. Newell, thesis, University of Utah (1975).
16. B. Mandelbrot, *Fractals: Form, Chance, and Dimension* (Freeman, San Francisco, 1977).
17. A. Fournier and D. Fussell, *Comput. Graphics* **14** (No. 3), 108 (1980); L. C. Carpenter, *ibid.*, p. 109.
18. J. D. Foley and V. L. Wallace, *Proc. IEEE* **62**, 462 (1974).
19. J. H. Clark, *Computer* **13** (No. 7), 59 (1980); S. Gupta, R. F. Sproull, I. E. Sutherland, *Comput. Graphics* **15** (No. 3), 71 (1981); H. Fuchs and J. Poulton, *VLSI Des.* **2** (No. 3), 20 (1981).
20. I. E. Sutherland, R. F. Sproull, R. Schumacker, *Comput. Surv.* **6**, 1 (1974).
21. E. Catmull, thesis, University of Utah (1974); J. F. Blinn, thesis, University of Utah (1978); J. M. Lane *et al.*, *Commun. ACM* **23**, 23 (1980); E. Cohen, T. Lyche, R. Reisenfeld, *Comput. Graphics Image Process.* **14**, 87 (1980).
22. H. Gouraud, *IEEE Trans. Comput.* **C-20**, 623 (1971).
23. B.-T. Phong, *Commun. ACM* **18**, 311 (1975).
24. T. Whitted, *ibid.* **23**, 343 (1980).
25. F. C. Crow, *ibid.* **20**, 799 (1977).
26. J. T. Scott, *Phys. Today* **32** (No. 1), 46 (1979); K. C. Knowlton, in *Computers for Image Making*, D. R. Clark, Ed. (Pergamon, New York, 1981), p. 131; R. Speer, W. Kovacs, R. Kovacs, *An International Guide to Computer Animated Films* (Animation Research, P.O. Box 2651, Seattle, Wash., 1979).
27. R. Langridge, T. E. Ferrin, I. D. Kuntz, M. L. Connolly, *Science* **211**, 661 (1981).
28. K. C. Knowlton and L. Cherry, *Comput. Chem.* **1**, 161 (1977).
29. J. F. Blinn, "Voyager II encounters Saturn," and other computer-generated films (Jet Propulsion Laboratory, Pasadena, Calif., 1981).
30. N. H. E. Weste, *Bell Syst. Tech. J.* **60**, 823 (1981).
31. D. P. Greenberg, *Sci. Am.* **230** (No. 5), 98 (1974).
32. B. Schachter, *Comput. Graphics Appl.* **1** (No. 4), 29 (1980).
33. C. P. Thacker, E. M. McCreight, B. W. Lampson, R. F. Sproull, D. R. Boggs, in *Computer Structures: Readings and Examples*, D. P. Siewiorek, C. G. Bell, A. Newell, Eds. (McGraw-Hill, New York, ed. 2, 1981).
34. Special issue on consumer text display systems, *IEEE Trans. Consum. Electron.* **CE-25** (No. 3) (1979).
35. E. Catmull, *Comput. Graphics* **12** (No. 3), 348 (1978).
36. F. C. Crow, *Computer* **11** (No. 3), 11 (1978).
37. B. D. Ackland and N. H. E. Weste, *Proceedings of 17th Design Automation Conference* (IEEE Press, New York, 1980), p. 285.

38. I thank P. Denes, K. Knowlton, and W. Ninke of Bell Laboratories for their help in preparing this article.