INTERACTIVE DESIGN OF VISUAL AIDS

Frederic I. Parke
Case Western Reserve University
Cleveland, Ohio 44106

Frank R. Koperda
I B M Corporation
Boca Raton, Florida 33432

ABSTRACT - The implementation of an interactive design tool using computer graphics techniques to produce high quality visual aids is described. The generated two-dimensional shaded images are composed of both pre-defined and user-defined opaque shapes used in conjunction with character strings. The user interface is simple and easy to use. User interaction is primarily via a large data tablet used to specify shapes and control image manipulation. The system alleviates many of the tedious mechanical aspects of visual aid production and allows concentration on the more creative aspects. There is immediate visual feedback for all actions. Implementation was done using a graphics extented Algol-60 in conjunction with an interactive real-time shaded graphics system. This operational system has produced a number of visual aids including those accompaning this paper.

Key Words and Phrases - computer graphics, interactive design, shaded images, visual aids, user interaction, visual feedback, image manipulation, real-time graphics.

CR Categories - 8.2, 3.41, 3.80.

1. OVERVIEW

In the academic, government, and corporate environments there is a continuing need for high quality visual images to illustrate and reinforce concepts and ideas presented in lectures, seminars, meetings, instructional video tapes, films, etc. This paper describes the implementation of an interactive design tool [1] which uses state-of-the-art computer graphics techniques [2] to produce high quality color shaded images in the form of photographic slides.

This interactive system is intended for persons with little or no computer system experience but who have a need to produce effective visual aids. The system allows the user to rework the position, color, size and shape of image components until the desired image is formed. It alleviates many of the tedious mechanical aspects of design generation and can be used to easily try out new ideas and modify old designs.

The system operates in real-time, with immediate visual feedback to each action. For example, a shape may be repositioned, scaled or erased and respecified as rapidly as the user takes the appropriate actions. Most images can be created and recorded on film within a matter of minutes.

A primary concern in developing this system was to provide an appropriate set of tools for the construction and modification of images. The following classes of user actions were found necessary:

1) Object generation
2) Object identification
3) Object modification
4) Object positioning
5) Grouping operations
6) Text string manipulation
7) Coloring operations
8) Image operations

2. GRAPHICS ENVIRONMENT

This image design system was developed using a graphics extended Algol-60 [3,4] implemented on the Case Shaded Graphics System [5]. This general purpose, interactive shaded graphics system contains sufficient computational power to produce about 30 images per second of dynamic three-dimensional environments. The environments are specified using polygons as the primitive structures.

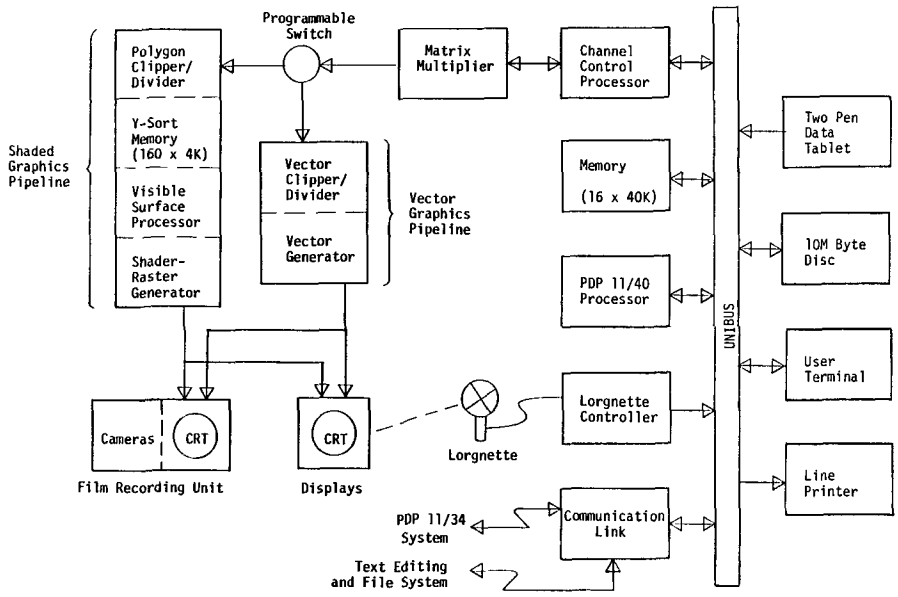


Figure 1 - The Case Shaded Graphics System

For this application a restricted environment, consisting of polygonal objects, each lying within one of a set of closely spaced parallel planes, one in front of the other is used. These parallel planes are embedded in the three-dimensional space.

The host machine is a PDP-11/40 operating under RT-11 [6]. The graphics portion of the system, see Figure 1, consists of a pipeline of specialized hardware processors. This pipeline contains an additional general purpose processor, a 4x4 matrix multipier, the hardware implementation of a polygon clipping algorithm [7], the hardware implementation of a visible surface algorithm [8], a shader [9] and a raster generator.

The two general purpose processors share the same main memory. The normal mode of operation is for a program executing on the PDP-11/40 to build up and modify a data structure corresponding to the desired image. A second process executing on the graphics pipeline processor traces through this structure and passes appropriate information to the specialized processors in the pipeline.

Images produced by this system consist of an array of 1024 x 1024 picture elements, each picture element specified by 8 bits of information in each of the three primary colors: red, green and blue. Color images are recorded on film using frame sequential techniques.

3. IMPLEMENTATION

There are three methods of shape or object generation included in the system. The first is to use one of the predefined geometric shapes; square, triangle, circle, vertical line, horizontal line or arrowhead. The second method is for the user to draw, using the data tablet pen, the outline of the desired shape. The third method is to copy or repeat an already specified object. Existing objects may also be erased.

As objects are created, they are automatically assigned a priority. The first object specified has the lowest priority; the last object created has the highest. Where two objects overlap, the one with the highest priority is visible.

An existing object may be modified by, making it larger or smaller, skewing it, deforming it, or raising or lowering its priority. When an object is skewed the x-coordinate of each vertex is modified based on its y-coordinate and a user specified parameter. Deformation scales the object independently in each dimension based on user specified parameters. Raising or lowering priority modifies the visibility of an object.

Object positions may be shifted up or down, right or left and object orientation may be changed. Objects may be grouped together using father-son and brother relationships. Related objects forming a group may be manipulated as a unit.

A special class of predefined objects are the alphanumeric characters. The user can specify character string objects; strings or characters within strings may be positioned, scaled, sized and generally manipulated in the same way as other objects.

The background as well as each object may be assigned a color. The color is selected from a color wheel consisting of 62 colors including black and white.

The entire function of this system is to generate images and record them as photographic slides. To this end the system allows the user to, save images for future use or modification, restore saved images, delete an image or record completed images on film.

The user indicates or identifies an object within the image simply by pointing at it. This is accomplished by moving a screen cursor to the object. When a new object is created the system assumes that all subsequent manipulations apply to that new object. This assumption is maintained until the user points to some other object.

To identify objects, the system keeps track of a surrounding box for each object. It compares cursor location with the boundaries of the object boxes. If the cursor is within a surrounding box the corresponding object is identified or selected. The selected object flashes off and on once, providing visual feedback. If objects conflict for selection, the one with the highest priority is selected. If the cursor is not pointing at any object or if its position leads to an ambiguous object selection, the system writes out a message to that effect, makes an audible beep and waits for the next user action.

4. USER INTERFACE

The user interacts with the system via three devices. The first is the PDP-11/40 console terminal. The user initiates execution of the design system from this terminal and receives certain prompts and error messages from it. The terminal bell, which actually makes a 'beep,' is used as an audio signalling mechanism.

The second device is an X-Y intensity modulated CRT display. The raster scan shaded images produced by the graphics pipeline are displayed on this device. These images have a resolution of 512 x 512 picture elements. Since the shader can only output one video stream at a time, these images are monochromatic. The display will show a black and white representation of any one of the three primary color components of the image. The normal default is to display the green component. This device provides feedback by continually showing the current state of the image being developed. Also displayed is a screen cursor which consists of a small flashing square.

The third device is a large dual pen data tablet. For this application only one pen is used to select command actions from a menu located on the tablet. In a similar manner the pen is used to select colors, move the screen cursor, and draw shape outlines.

The command menu is shown in Figure 2. The top two rows contain the command buttons which generate new objects. Pointing to SQUARE, TRIANGLE, CIRCLE, VERT LINE, HORIZ LINE or ARROW HEAD causes a new instance of the selected predefined shape to be created.

REPEAT causes a new instance of the currently identified existing object to be created. BOUNDARY specifies scaling information used when an object is drawn. DRAW initiates the drawing of a user specified shape. The object being drawn is visible on the display monitor as it is being drawn.

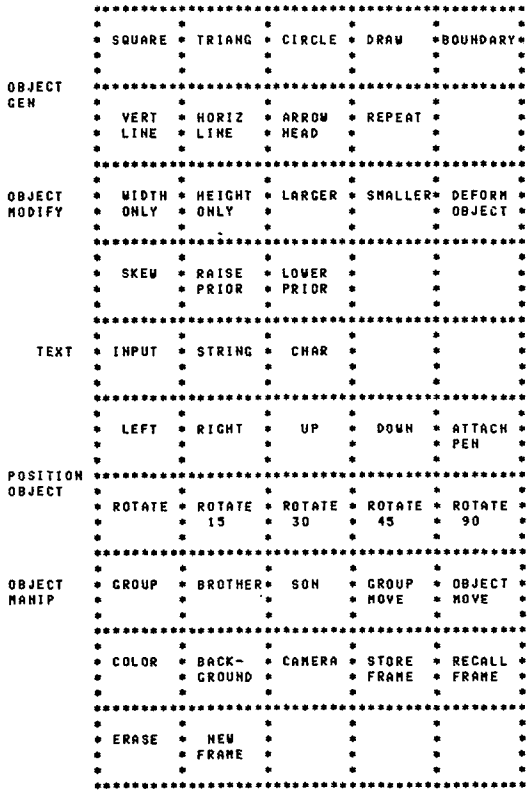


Figure 2 - The Command Menu

The third and fourth rows of the menu are concerned with modifying existing objects. LARGER and SMALLER change the scale of the identified object. The object gets continually larger or smaller as long as the pen is touching the command button. WIDTH ONLY and HEIGHT ONLY modify this action to affect only one dimension.

DEFORM OBJECT and SKEW distort the object based on pen motion. LOWER and RAISE PRIOR cause discrete changes in object priority which may affect object visibility.

The fifth row deals with character strings. INPUT initiates the input of a character string from the console terminal. STRING causes identification and manipulation of entire strings. CHAR causes identification and manipulation of characters within strings.

The sixth and seventh rows are concerned with object positioning and orientation. LEFT, RIGHT, UP, DOWN and ROTATE move the identified object in the indicated direction for as long as the pen is touching the command button. ROTATE 15, 30, 45 and 90 cause discrete rotations which occur once each time the command button is touched. ATTACH PEN causes the identified object to follow the pen motion until the pen is lifted.

The eighth row controls object grouping. Using GROUP, BROTHER and SON, tree structured object groups can be assembled. GROUP MOVE and OBJECT MOVE control whether subsequent manipulations apply to groups or objects within groups.

Pointing to COLOR is followed by pointing to a color on the color wheel. The selected color is assigned to the currently identified object. BACKGROUND is the same as COLOR except that the selected color is assigned to the background.

CAMERA initiates recording the current image on film. STORE FRAME dumps the current image specification onto a disk file; RECALL FRAME restores an image from a disk file. The user specifies the file name from the console terminal. ERASE deletes the currently identified object. NEW FRAME deletes the current image and re-initializes the internal image data structure.

5. EXAMPLES

Figure 3 shows several images produced using this design system. Figures 3(a) thru 3(f) illustrate the construction of an image. Figures 3(g) and 3(h) show variations on the same theme. The remainder of this section describes the sequence of user actions used to create these images.

First the user attaches a copy of the command menu and color wheel to the data tablet. He then initiates the execution of the design system program. The system asks him to indicate, with the pen, the locations of the menu and the color wheel. For Figure 3(a), the background was colored yellow. A predefined circle was selected, enlarged, positioned and colored blue. For Figure 3(b), the text string 'ACM 77' was input, scaled, positioned and colored blue. For Figure 3(c), the text string 'SEATTLE' was input, scaled, positioned and colored green. The color of the circle and the first text string was changed to a darker blue. For Figure 3(d), a boundary area was specified and the snow covered portion of the mountain was drawn, scaled, positioned and colored white. Next, the rest of the mountain was drawn, scaled, positioned and colored dark purple. The color of the circle and the first text string was again changed to a lighter blue. For Figure 3(e), the ground semi-circle was drawn, scaled, positioned and colored green. Also, the 'SEATTLE' text string was stretched out. For Figure 3(f), the Space Needle was drawn, scaled, positioned and colored white. The purple part of the mountain was lightened slightly. The color of the circle and the first text string were changed to a darker blue. At this point the image was considered complete.

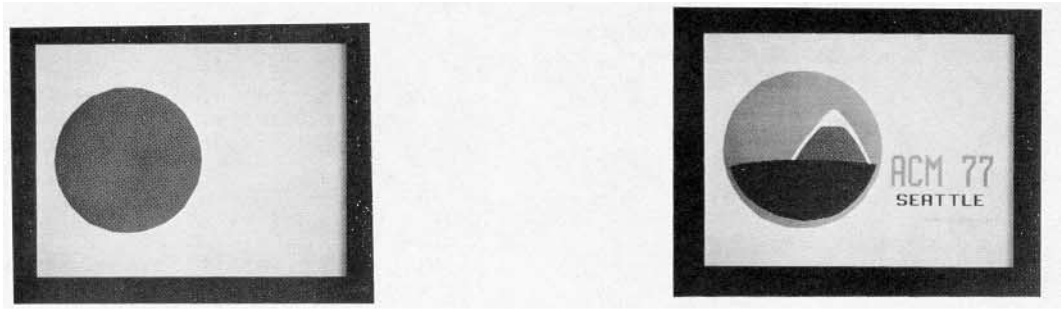
Figures 3(g) and 3(h) show other completed images which are variations on the same theme. For Figure 3(g), the Space Needle and the character strings are white, the circle is medium blue and the background is dark green. Figure 3(h) is intended to suggest a sunset. The sky is yellow-brown, the ground is dark green-brown, the Space Needle is brown, and the characters are red-brown.

6. LIMITATIONS

The system has the following known limitations. It can handle at most 20 objects. A text string is considered one object. The upper bound on the number of text characters per image is about 100. This bound depends on character complexity. These limitations are induced primarily by limited main memory to store the image data structure.

The text string character sub-structure, which was initially thought to be a valuable feature, is really not very useful and tends to clutter up the internal data structure.

Even though the user selects colors from a color wheel showing the actual final photographic image colors, it is often frustrating to interact with a monochromatic representation of the image.



(a)

(e)

(b)

(f)

(c)

(g)

(d)

(h)

Figure 3 - Examples

7. SUMMARY

This design tool is operational and has been used by several people to create a number of images. It is not, however, considered a production tool but rather a basis for continuing development in this area. User input, primarily in the form of user frustration, will serve to direct further development.

REFERENCES

[1] F. Koperda, "An Interactive Two-Dimensional Shaded Graphics Design System," M. S. Thesis, Department of Computer Engineering, Case Western Reserve University, January 1977.

[2] W. Newman and R. Sproull, PRINCIPLES OF INTERACTIVE COMPUTER GRAPHICS, McGraw-Hill, 1973.

[3] B. Jones, "An Extended Algol-60 for Shaded Computer Graphics," M. S. Thesis, Department of Computer Engineering, Case Western Reserve University, June 1976.

[4] B. Jones, "An Extended Algol-60 for Shaded Computer Graphics," Proc. of Symposium on Graphic Languages, Miami, April 1976.

[5] F. Parke, "The Case Shaded Graphics System," Technical Memo, Department of Computer Engineering, Case Western Reserve University, February 1976.

[6] RT-11 SYSTEM REFERENCE MANUAL, DEC-11-ORUGA-B-D, Digital Equipment Corporation, Maynard, Mass.

[7] I. Sutherland and W. Hodgman, "Reentrant Polygon Clipping," CACM, Vol 17, No 1, January 1974.

[8] G. Watkins, "A Real-Time Visible Surface Algorithm," University of Utah Technical Report UTECH-CSc-70-101, June 1970.

[9] H. Gouraud, "Computer Display of Curved Surfaces," University of Utah Technical Report UTECH-CSc-71-111, June 1971.