DIGITAL VIDEO DISPLAY SYSTEMS AND DYNAMIC GRAPHICS

Ronald Baecker

Associate Professor of Computer Science and Electrical Engineering
University of Toronto

President
Human Computing Resources Corporation

Abstract: Most digital video display systems have been capable of producing only text or static imagery. This paper shows that these limitations are not intrinsic to the technology, but are rather a direct consequence of the display system architecture. The paper begins by summarizing some of the background required to understand digital video display systems. The state-of-the-art is then surveyed, supported by an extensive bibliography. Existing systems are described in terms of a methodology which clarifies the effect of system architecture on capabilities and performance. It is shown how dynamic graphics capabilities can be provided if systems adhere to one or the other of two possible architectures. Examples of such systems are presented.

C.R. categories: 8.2, 6.35
Keywords and phrases: digital video display, video display, video raster system, raster display, raster graphics, dynamic graphics, computer animation, animated graphics.

1. BACKGROUND

1.1. *Digital Video Display Systems*

1.1.1. *Fundamental Concepts*

A *digital video display system* is one in which:

1) The final image is displayed on a raster scan cathode ray tube (television type terminal); and,

2) There is an underlying digital representation of the image stored in a controlling computer. (This condition excludes systems such as the Tricolor Cartograph [Kubitz 69], and Computer Image's Scanimate and Caesar [Honey 71].)

The raster scanning signal may but need not conform to an international TV standard, such as NTSC or PAL, which determines characteristics of the signal such as its frame rate, the number of lines per frame, and the method of colour encoding [Pearson 75]. The digital representation may be a *frame buffer,* in which contiguous chunks of memory (such as bits or bytes) represent contiguous pixels (discrete elements of the final picture), or it may be an encoded representation of the picture. In the latter case, it must be transformed into video format *(scan converted)* by a video display processor on its way to the cathode ray tube.

1.1.2. *Advantages*

Digital video systems emerged in the late 60s and came into their own in the middle 70s as a viable alternative to random scan displays (also known as calligraphic, stroke-writing, line-drawing, or vector displays). Initially, digital video displays were attractive because they were clustered systems which shared expensive image generators and allowed additional workstations to be added at low incremental cost. The universality of television meant that digital video workstations based on TV monitors could take advantage of their low cost and ready availability of maintenance expertise. The communication of images from site to site could also be facilitated by the use of the established technology of television transmission. (An early discussion of these advantages appears in [Hendrickson 67a,b].)

Other aspects of television gave raster scan displays an advantage over random scan terminals. Lines could easily be thickened; areas could be shaded. Grey scale, colour, and black-and-white reversal could be employed. The computer-generated images could easily be mixed with live television images, either still or moving.

1.1.3. *Disadvantages*

Raster scan displays, however, have had some significant problems to overcome. Most have been limited to resolutions such as 256 X 256, 240 X 320, 512 X 512, or 480 X 640. They have been expensive due to the cost of providing this much electronic memory. They have been slow due to the difficulty of the processing required for scan conversion. These two components, memory and processing power, will prove central to the analysis which follows.

1.2. *Display Processors for Vector Graphics*

Translation of an encoded picture into a television signal requires a *video display processor.* These are similar to the display processors which traditionally have been used to refresh vector displays from *display files.* Thus we will review our current understanding of vector displays, with an analysis and figures adapted from [Newman 74].

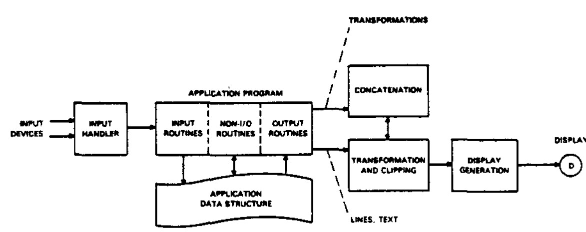

Figure 1 - Typical Structure of an Interactive Graphics System
NOTE: Figures 1 through 4 are adapted from [Newman 74].

A diagram of a typical graphics system appears in Figure 1. The output process consists of the modules beginning with and to the right of *output routines.* Output routines are the statements in the application program which define how the data is to be visualized for display purposes. The *transformation and clipping routines* carry out scaling, rotating, and translating of the pictures, and clip them to rectangular boundaries to remove parts that should not appear on the screen. The *concatenation routine* carries out the combining of multiple tranformations which are to be applied to the same object. The *display generator* usually includes a vector generator and a character generator, and converts the picture into a form suitable for the display's deflection system.

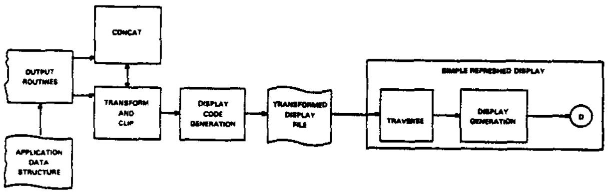

Figure 2 — A Simple Refreshed Display

These modules comprising the display output process have typically been realized in a variety of different forms, such as those depicted by Figures 2, 3, and 4. Figure 2 shows the organization of a simple refreshed display. The CPU performs the transformations, inserting new transformed and clipped pictures into a *transformed display file,* in which each distinct picture is stored as a distinct entity, or *segment.* Performance of the system tends to be limited by the speed with which segments can be recomputed by the CPU.

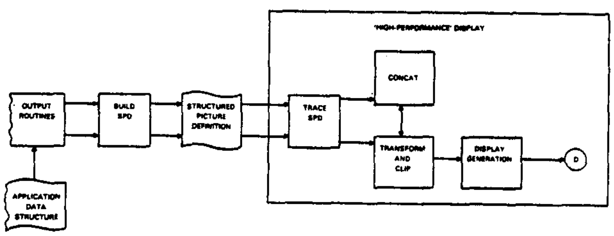

Figure 3 - A 'High-Performance' Display with a Structured Picture Definition

The "high-performance" display architecture in Figure 3 was developed as a solution to this performance limitation. Pictures are stored in an encoded or structured form, a *structured picture definition* (SPD). A display processor, usually hard-wired, interprets this structure, performs the appropriate transformation and clipping, and sends the picture to the display. The intent of the design is that changes to the SPD are to be immediately visible on the screen. However, if the display becomes too complicated relative to the capabilities of the display processor, the picture will begin to flicker.

An alternate solution, the *buffered transformation processor* depicted in Figure 4, overcomes this difficulty. As in the previous solution, we have a hard-wired transformation processor, but now its function is separated from the refresh processor. Thus transformations and refreshing can occur independently and at different rates. Because each segment is double buffered in the display file, the old version will always be visible until the new one is computed. Since the refresh process is typically faster than the transformation process, a more complex flicker-free image may be displayed.

There are other possible architectures for vector graphics, for example, the storage tube display [Preiss 78]. Storage tubes have been popular because they are cheap and because they can display a complex flicker-free image of high resolution. Digital video displays have long been regarded as

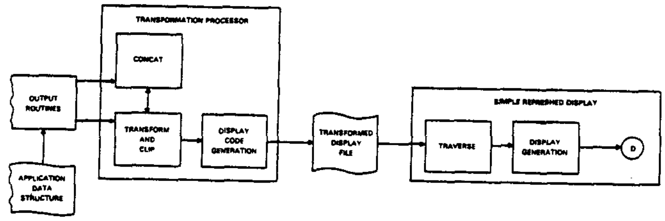

Figure 4 — A Buffered Transformation Processor

selectively eraseable storage tube replacements. This is misleading, for it suggests, incorrectly, that digital video technology is only appropriate for static graphics.

1.3. *Related Technological Developments*

The evolution of digital video display systems has been and will continue to be affected by a variety of technological and economic factors.

1.3.1. *Advances in Microelectronics*

Large-scale integrated circuit technology makes feasible digital video display system designs that were impossible a few years ago.

Most important has been the availability of large cheap semiconductor memories. The 1K MOS RAM was first delivered in 1970 and became available in volume in 1972. The 4K MOS RAM was first delivered in 1973 and became available in volume in 1975. The 16K MOS RAM was first delivered in 1976 and is now available in volume. 64K MOS RAMs are just now becoming available. Thus we already have the capability of storing a binary medium resolution (256 X 256) raster image in randomly accessible form on a single chip.

The pace of these developments is expected to continue through several more advances before encountering any fundamental obstacles, although they may be slowed somewhat by the need to move from visible light to electron beams to X-rays for drawing the integrated circuit masks. Thus the cost in the mid 1980s of buffering an entire 512 X 512 X 8-bit picture in random access memory should be under $100.

Another relevant technology is that of shift registers, or cyclic memories. Semiconductor shift registers could be used cost-effectively in frame buffers in the early 1970s when RAMs were still much too expensive. Currently, charge-coupled devices (CCDs) seem to be offering four times the storage of RAMs at comparable prices. Thus systems that would be too expensive using RAMs may be feasible albeit with reduced performance using CCDs.

1.3.2. *Home Terminals - Complementary Technologies and Competing Vendors*

The largest potential market for digital video display systems is in home computer terminals. There are three complementary technologies whose vendors will compete in the evolution of home computer terminals. These technologies are those of the computer, the television screen, and the communications link to outside resources.

Vendors of hobby and personal computers will seek increasingly to expand their machines into complete home computers and into terminals to sophisticated remote services. Since the advent of the personal computer early in 1975, over 200,000 have been sold in North America. Sales in the 1980s have been projected as high as tens of millions of units [Nelson 77].

Vendors of TV games will seek increasingly to achieve greater flexibility and versatility by providing programmable systems of greater and greater power. One vendor has predicted that every family would have its own home computer by 1983 [Roseman 77].

Vendors of interactive television will seek increasingly to enhance the range of services and imagery that 2-way cable can provide. For example, the British Post Office has for several years been working with two experimental systems, Teletext and Viewdata [Saxton 77; Malik 78]. Teletext allows the viewer to request, store, and display one of a small number of pages of text which are constantly being transmitted in unused lines of the television signal. Viewdata allows him to request a specific page from a large library of pages. In both cases, the desired page is stored in a digital memory inside a specially modified television receiver. Teletext constantly cycles a fixed number of pages; the viewer selects one by means of thumb wheel switches or push buttons. With Viewdata, the viewer requests a specific page from the library by dialing a central computer and indicating his choice with a touch-tone phone or special keyboard. Although the image display capabilities of these systems are quite limited, this need not be so, as can be seen in the Canadian Department of Communications Telidon project [Bown 78].

2. A METHODOLOGY FOR THE DESCRIPTION OF DIGITAL VIDEO DISPLAY SYSTEMS

Numerous digital video display systems have been designed and built in the past ten years. Some have been created as unique experimental implementations; others have been developed and sold commercially. Some have been single-user displays; others have consisted of clusters of terminals. Systems have been developed using a great diversity of techniques for a great diversity of applications. How can we compare and contrast existing designs if they are so diverse?

We propose a methodology for describing digital video display systems which focuses on a few key attributes of their organization. This methodology will help us organize our understanding of existing systems, and, embodied in an appropriate modelling environment, could help us predict the efficacy of proposed designs. The methodology is particularly helpful in clarifying the relationship between the design of a digital video display system and its capacity for dynamic graphics. We shall introduce the methodology with a simple example.

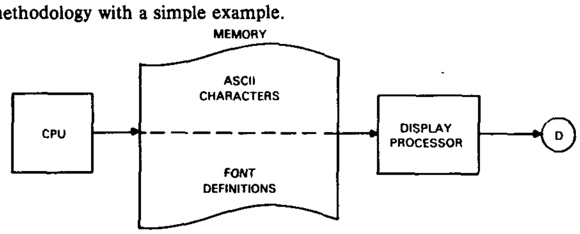

Figure 5 — A Character-Oriented Video Display Terminal

Consider the common alphanumeric (character-oriented) *video display terminal.* One possible architecture for such a system is shown in Figure 5. The CPU is a microprocessor. The memory contains a display file consisting of the ASCII characters currently on the screen, and also a font definition, a representation of the appearance of the characters. The former is always stored in RAM; the latter is often stored in ROM. The display processor combines the ASCII values, the representations of the characters in the font memory, and the positions of the characters on a coarse grid (such as 24 X 80) to provide successive intensity amplitudes for the television scan. The display processor is simple because it interprets only one instruction, i.e., display a character. It is so simple that it is now possible to synthesize it out of a few chips.

Yet this is not the only system architecture that can carry out the same task. Characters can also be displayed with systems constructed according to the architecture depicted in Figure 6. Processor P1 performs character expansion from the memory of the CPU into a *bit map,* which is a frame buffer with a single bit of memory per pixel. Processor P2 fetches successive chunks from this memory and uses them to provide successive intensity amplitudes for the television scan.

There are advantages and disadvantages to both approaches. The design of Figure 5 has typically led to a cheaper realization. It is also more dynamic, allowing more rapid change to the picture on the screen. The design of Figure 6 is more flexible, allowing characters in arbitrary fonts as well as vectors, curves, and line drawings.

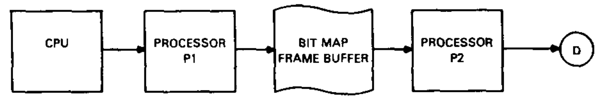

Figure 6 - Another Architecture for Character Display in Video Format

The example suggests an approach to the analysis of existing designs and the synthesis of new ones. Our methodology represents digital video display systems as configurations of processors and memories with specific performance characteristics. This technique helps us to survey, compare, and contrast the great diversity of past and present systems.

A common element in these systems is the need for *scan conversion.* Scan conversion is the transformation of a picture represented in terms of points, vectors, and characters into one represented in a raster scan format. In Figure 5, scan conversion is carried out by the display processor. In Figure 6, scan conversion is carried out by P1.

Scan conversion can be a time-consuming computation, one which can be carried out in different ways depending upon the representations in which the picture is stored before and after the conversion process, and the amount of available storage. Various scan conversion algorithms are described in [Newman 79], [Metzger 69], [Jordan 73], [Barrett 74], and [Thornhill 74].

3. A SURVEY OF EXISTING SYSTEMS

3.1. *Sequential Access Frame Buffer Systems*

The earliest digital video displays were clustered systems in which multiple terminals were refreshed from one expensive centralized image computation and storage system. They were frame buffers, in which there was a direct correspondence between chunks of memory and image pixels on the screen. Most early realizations, developed in the late 60s, were based on rotating memories. (A delay line implementation is described in [Grover 71].) Later systems, in the early 70s, incorporated electronic shift registers. These choices were the only cost-effective solutions available with the then current technology.

A diagram describing the architecture of such systems appears as Figure 7. Processor P1, when present, performs such services as vector and character expansion. The sequential nature of the cyclic memory limits performance, because P1, in updating the frame buffer, must wait until the appropriate point of the memory is available for updating. Processor P2, which reads from the frame buffer to control the television scan, is not limited by the cyclic memory, because the data is stored in the order that is required for the scan.

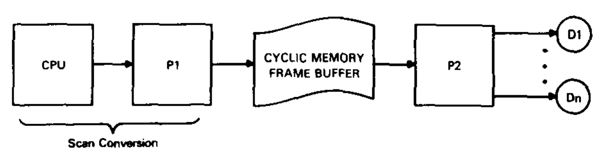

Figure 7 — A Sequential Access Frame Buffer System

3.1.1. *Systems with Physically Rotating Memories*

One of the earliest systems of this kind was the IBM 1500 Instructional System [Aziz 67; Terlet 67]. The system provided coarse resolution text in a variety of fonts to 32 terminals. Character streams and font definitions were stored in the CPU, which scan converted them (12 scan lines at a time) and wrote them onto tracks of a digital disk. Processors P1 and P2 were minimal.

A similar system was the Brookhaven Raster Display System [Ophir 68a,b]. The system provided 512 X 512 binary images to 32 terminals. The refresh memory was a 32 track rotating digital drum, with one read head per display. Scan conversion was performed on an entire image in the CPU. Again P1 and P2 were minimal.

A different approach to the problem of scan conversion was taken in the Rand Video Graphic System [Uncapher 71]. The processor P1 consisted of a microprogrammed graphic display controller, a vector and character generator, three vidicon scan-converter tubes, and three 32 position switches. Vector graphic images were written as charge patterns onto the faces of the vidicon tubes, then read off the tubes in raster format and routed via the switches onto tracks of a 32 track analog video disk. These pictures could be mixed with each other and with natural pictures from a television camera by the processor P2 and distributed to workstations located throughout the laboratory.

A similar system was implemented at the University of New South Wales [Macaulay 68; Rose 67,68]. Another style of use of an analog video disk is described in [Staudhammer 75]. Commercial digital disk systems included the Anagraph from Data Disc (now Ancomp) [Ruder 68] and the early models from Comtal, now both obsolete.

3.1.2. *Systems with Shift Register Memories*

One early commercial product of this kind was the Computek 300 (circa 1970). This terminal was designed as a selectively eraseable replacement for the storage tube. To keep the cost down, it was limited to a 256 X 256 binary picture. Possibly because of its low resolution, the product was not successful.

A successful commercial product (circa 1972) was the Ramtek GX-100. Ramtek was formed by a group of Data Disc employees who felt that the future lay with solid state memories. The GX-100 was a clustered system like those from Data Disc. P1 contained a vector generator, a character generator, and a number of generators for special displays such as rectangles and graphs. Each memory bank stored one bit per point; the outputs from several banks could be combined to yield gray scale or colour capabilities (a feature previously introduced in the Data Disc systems). Also as in the Data Disc systems, these banks could be purchased at varying resolutions from 256 X 256 to 512 X 768. The system had an awkward memory organization, where each bank consisted of 16 cycling shift registers, 8 for each TV field, which did however have the advantage of reducing the average access time for a point to 1 millisecond.

A notable experimental system was that designed by Dick Shoup at the Xerox Palo Alto Research Center [Shoup 74]. Shoup's frame buffer consisted of one very long shift register refreshing a 480 X 640 X 8-bit picture. The average time to access an arbitrary point was 16.7 milliseconds. The 256 colours available at any one time could be chosen from a total of 1 billion possible colours through the use of a *programmable colour map* which allowed 10 bits of red, 10 bits of blue, and 10 bits of green. P1 was a run-length encoder-decoder which transformed pictures between the frame buffer and the CPU. P2 included a video switching network which allowed single frames to be routed to one track of a 300 track analog video disk. This disk was used for real-time playback of animation sequences and for transferring them to video tape. Another experimental system, a 256 X 256 X 8-bit gray scale display implemented with MOS shift registers, is described in detail in [McCracken 75].

3.2. *Random Access Frame Buffers*

By 1973, the availability of 4K MOS semiconductor chips made it possible to build large random access frame buffers, whose architecture is depicted in Figure 8. These systems were usually configured as a number of memory planes with resolutions such as 256 X 256 or 480 X 640. 8 bits of 512 X 512 resolution was not uncommon. Modern semiconductor technology also often affected processor P1; it could now be implemented as a microprocessor rather than with discrete hard-wired logic, and so the entire display could be located remotely from the host CPU.

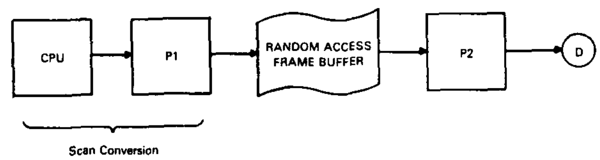

Figure 8 — A Random Access Frame Buffer

The major impact of the random accessibility of points is increased performance of the scan converter P1. Scan conversion can proceed in one pass over the data without being constrained by the cyclic availability of points in the frame buffer. A secondary impact is manifested in P2 in the more sophisticated systems. Because P2, like P1, can access the frame buffer in random fashion, it can perform a variety of reformatting and mixing functions on several images stored in a very large frame buffer.

One of the earliest examples of such systems was an experimental display system constructed at Bell Telephone Laboratories at Murray Hill, New Jersey [Noll 71; Denes 74,75]. One-, two-, or three-bit gray scale or colour pictures were refreshed directly out of the CPU's memory. Software scan conversion was aided by a hard wired processor P1 which computed the memory word and bit positions from a pixel's X and Y coordinates.

Commercial random access frame buffers, all with independent semiconductor refresh memory, are currently made by the following manufacturers: Aydin Controls, Calcomp, Child, Chromatics, Comtal, DeAnza Systems, Evans and Sutherland, Genisco, Grinnell Systems, Hewlett Packard, Intermedia Systems, Interpretation Systems, Lexidata, Matrox, Norpak, Ramtek, and Videographics. Most systems are similar to the GX-100, albeit with enhanced performance due to the use of random access memory. Their products differ in the number of possible independent displays supported by the same memory and processing logic, the available resolutions, the number of possible bits per pixel, the memory organization, and in the vector and character expansion capabilities of their P1 processors. There is also great variety in their P2 processors. They may include programmable colour maps, programmable cursors, zoom and pan capabilities, and image overlay capabilties. The Calcomp IGT 100 and the Hewlett Packard 2648A [HP 78] are single work-station black-and-white displays; most of the other products have colour capabilities, and several can handle multiple terminals. One unique feature of the Calcomp is a three-way split-screen capability which allows simultaneous display of graphics, a blown-up portion of the graphics, and text.

The most sophisticated system of the mid 70s was the Evans and Sutherland Video Frame Buffer [Kajiya 75]. The 512 X 512 X 8-bit multi-ported MOS memory is 8-way interleaved and comes with its own mapping hardware so that the entire memory can be addressed by the host CPU. Various kinds of flexibility have been provided. "Format control" allows one to vary the number of horizontal and vertical raster elements, the kind of interlace desired, and the duration of both horizontal and vertical flyback times. "Memory control" allows one to refresh different frames from different sections of memory. "Intensification control" provides a programmable colour map with 12 bits of red, 12 of blue, and 12 of green. The system can be used to present one high resolution TV frame, to play back low resolution motion sequences, to scroll through a large picture, and to display sections of a high resolution frame to a precision film recorder.

Recently, a few even larger experimental systems have been designed and implemented. [Fischer 73,75] describes a system which can handle up to 4 images of up to 640 X 480 X 8 bits, can translate and scale each image independently, and can compare and create special effects between images through a series of programmable priorities between the images. [Entwisle 77] and [Negroponte 77] describe a system which drives a variable resolution monitor, which allows a flexible trading off between resolution and the number of bits-per-point, and in which memory planes need not be of the same size, proportion, or position. Planes may be combined via programmable priorities into images; portions of these images which fall within rectangular windows can be mapped into rectangular viewports on the screen. [Taylor 77] describes a system with a 512 X 512 X 34 bit memory in which various 512 X 256 planes can be flexibly combined to form 2 video displays. A scan transfer processor allows the programming of scanning patterns which control the order of data transfers between the memory and an external CPU and array processor.

At the other extreme, [Matherat 78] presents the design of a single LSI chip to implement all the control functions of an interesting frame buffer.

3.3. *Systems with Coded Picture Definitions*

Whether frame buffers are sequentially or randomly accessible, the speed with which changes to the picture are reflected on the screen is limited by the speed of scan conversion. A variety of designs have attempted to provide instantaneous appearance of picture changes by refreshing the screen directly from a *coded picture definition.* The picture is constructed "on the fly" from a condensed digital representation. This architecture is comparable to the use of structured picture definitions with vector displays. There, increasing the complexity of the structured picture definition causes the refresh process to slow down, introducing flicker. With raster displays, the refesh process cannot slow down, so limitations must be placed on the complexity of the picture definition to match the capabilities of the refreshing (and scan converting) processor. Such an architecture is depicted in Figure 9.

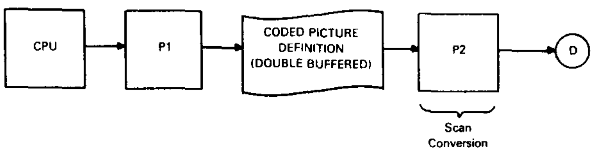

Figure 9 — A System with a Coded Picture Definition

In practice, a variety of restrictions have been placed on the picture definitions. The most common is that of the alphanumeric video display terminal (VDT). The design of a well-engineered family of such terminals is documented in [HP 75]. [Irby 74] and [Andrews 74] discuss a microprocessor-driven VDT which allows the screen to be divided into a number of independent windows for text display, scrolling, and editing. [Baskett 76] presents a novel VDT design in which the memory can be used either to refresh a 480 X 640 binary picture or to store 600 80-character lines of text. Programs in the controlling microprocessor allow one to move through this text with a variety of scrolling and page turning mechanisms.

The first commercial system to go beyond alphanumeric characters was the Ramtek FS-2000, which could also display a limited number of horizontal and vertical lines and rectangular areas. Its display processor P2 executed instructions at a rate of one every 90 nanoseconds. Although each scan line could be encoded in a "program" of up to 600 instructions, typical scan lines were encoded by a very few instructions. Thus Ramtek provided only 16,000 bits of memory to store 256 X 480 X 4-bit pictures, a 30 to 1 reduction in storage over a frame buffer, achieved at the cost of generality.

Several recent systems have been based on run-length coded display files. Run coding provides an efficient storage technique for simple graphic displays, and one which is easily decoded by a raster scanning refresh processor. [Hunter 75a,b] describes a simple run-coded colour video graphics system intended for simple cartoon animation. [Laws 75] describes a run-coded gray-scale graphic processor intended for high quality text display. [Myers 76] describes a run-coded system intended for storing and playing back digital colour video animation sequences. This is achieved via high-speed DMA transfer from a large digital disk to double-ported semiconductor memory, and from there to a run length decoding system which expands it into NTSC format for display. Another experimental design is described in [Barenholtz 76]. The only such system that is available commercially is a recent product from Three Rivers Computer Corporation.

Many recent systems have explored encodings based on generalizations of the "characters" that can be placed on the coarse grid of the alphanumeric VDT. Jordan and Barrett [Jordan 74] refer to these as *cell displays.* The simplest example of this is the Commodore PET, which has a number of special symbols which allow crude graphic images to be assembled. The Exidy SORCERER, and the Intecolor and Compucolor Systems from Intelligent Systems Corp., provide a programmable character generator to enrich the class of possible images. [Jordan 74] describes an ingenious method whereby 104 basic 8 X 8 patterns can be transformed and combined to produce all possible line drawings.

Another cell display, the Tektronix 4025 "gralpha terminal" [Tek 78a], allows a mixture of text and graphics. A rectangular graphics region is defined anywhere on a large scrollable page; the remainder of the page is left for characters. The page is divided into a coarse grid of cells. Each element on the grid contains a pointer; in the case of the character region, it is a pointer to a character generation ROM; in the case of the graphics region, it is a pointer to a bit map in a graphics memory RAM. Memory is used efficiently because null pointers are included for those grid positions where no picture exists. Tektronix calls this a *virtual bit map.* The Tektronix 4027 is a similarly designed colour display [Tek 78b].

Finally, there exist very expensive high-performance systems which produce in real time moving digital raster images from three-dimensional vector graphic representations. These are generally pipelined processors which transform an edge and surface description of a collection of three-dimensional objects into the viewer's coordinate system, remove hidden surfaces, and smooth shade the result. Examples include the General Electric NASA system for space simulation [Rougelot 69], several systems built by Evans and Sutherland which have never been described in the open literature, and two designs described in [Eastman 75] and [Tripp 75].

3.4. *Systems with Coded Picture Definitions and Double Frame Buffers*

In vector graphics, the buffered transformation processor depicted in Figure 4 allows the transformation and refresh processes to proceed at separate speeds, and a steady flicker-free image to be maintained, independent of the time required to transform the picture. When a change is introduced, the old version of the image is displayed until the new one is computed. We can apply the same idea to raster displays. The problem with the architecture of Figure 9 is that the system cannot perform the scan conversion in a frame time if the image complexity becomes too great, and so the picture will disintegrate in some graceful or ungraceful way. Therefore we introduce a double frame buffer after the decoding and scan converting processor P2, yielding the architecture shown in Figure 10. The system will switch back and forth between buffers after each new one

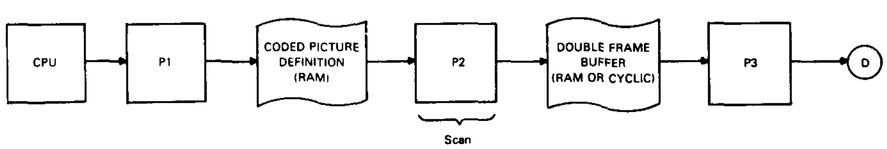

Figure 10 - A System with a Coded Picture Definition and a Double Frame Buffer

is computed. Now new images can be computed and displayed, no matter how complex they are, we may only have to wait longer to see the result. This is not a solution to be undertaken lightly, because the system now includes both an expensive scan conversion processor P2 and three representations of the picture, including a double frame buffer.

One example of such a system is the Interim Dynabook implemented at the Xerox Palo Alto Research Center [Kay 74; LRG 76; Baecker 76]. Primitive pictures are defined as 256 X 256 rasters of points, each black, white, or transparent. They are stored in a highly compact form, using a hierarchic area encoding scheme. A dynamic picture is represented as a tree structure of primitive pictures. The display processor traces the tree structure, scan converts, translates, clips, and does transparency-opacity calculations (to determine which portions of which layers are visible) in real time, approximately 3 to 10 frames per second. The results of these calculations are double buffered to enhance the perception of movement. In fact, the CPU-P1 and P2 are the same processor running different tasks, and the coded picture definition and the two scan converted frames are stored in the same memory.

A recent development of this kind is the Lektromedia LEK 330 Series [Burtnyk 77]. Here the CPU is a remote host, P1 is non-existent, the coded picture definition is actually a vector format display file, and the scan converting processor P2 the rather limp Motorola 6800 microprocessor. Systems can be configured either with a single black-and-white image buffer, thus adhering to the architecture of Figure 8, or with dual image buffers, as is shown in Figure 10.

Although no systems of the following kind exist, a cost-effective design might incorporate cyclic memories (such as MOS shift registers or CCDs) to implement the double frame buffer. P3, which simply translates memory chunks into display intensities, will not be affected by this. P2 will slow down, but may be able to compute fast enough for simple pictures appropriately coded.

4. DYNAMIC GRAPHICS

Animation has been defined as:

> "The art of giving apparent movement to inanimate objects. The word is also used for the sequence of drawings made to create the movement, and for the movement itself when seen on the screen." [Halas 71]

Animation, and all motion pictures, are made possible by the *persistence of vision* of the human eye [Madsen 69]. The retina retains the image of an object for a brief instant after the object has been removed. Thus a sequence of still pictures presented rapidly enough will blend together into a single continuous image. If the stills depict progressive phases of a single movement, the eye will perceive continuous motion. The object will be animated, that is, "come to life".

Computer animation, therefore, consists of the rapid presentation of computer graphic images to create the illusion of movement. Often, these images will have been computed from an algorithmic description, although the computations may consist of elementary combinations of hand-sketched images. Often, the computation will occur concurrently with the presentation, although images that cannot be computed in real-time may still be played back in real-time under computer control [Tilson 76].

Early work in interactive computer-mediated animation [Baecker 69a,69b,74] formalized a set of basic primitives for achieving efficent implementations of simple cartoon animation. The animator sketches static images (cels), descriptions of movement of portrayed objects (movement descriptions), and descriptions of methods for portraying these objects using the cels (selection descriptions). The computer combines these elements in real time into an animated sequence. Although early work was based on vector graphic technology, the same concepts have been realized successfully in the Interim Dynabook raster scan system described in Section 3.4.

More recent work in computer animation has tended to emphasize the process of *key-frame animation,* a technique which attempts to automate the in-betweening process of classical animation production. This technique is very expensive, and we shall not consider it further in this paper.

Computer animation is the most dramatic and effective of the forms of dynamic graphics. But there are also other forms which have relevance to the future of the home terminal. Some techniques commonly used in cinematography include [Madsen 69]:

> the *cut,* an instantaneous change from one scene to another;
> the *fade-in(out),* the gradual appearance (disappearance) of a scene from (to) a darkened screen;
> the *dissolve,* consisting of a fade-in superimposed on a fade-out over the same interval;
> the *double exposure,* combining two separate images, one appearing as a translucent phantom before the other;
> *superimposition,* combining two or more separate images, in which some appear opaquely before others, as a foreground placed before a background;
> the *pop-on,* an instantaneous appearance of a new image within a scene already on the screen;
> the *wipe,* an optical effect whereby the scene on the screen is apparently physically displaced by the following scene;
> the *flip,* an effect in which a still scene begins to revolve on its centre axis in acute perspective, and a new scene or a new title is introduced with each half revolution;
> the *spin,* the rotating of a scene around its own centre point;
> the *zoom-in(out),* a continuous approach (retreat) by the camera to (from) a subject;
> the *pan,* a horizontal scan of the subject by the camera; and,
> the *close-up,* an enlargement of a detail within a larger picture.

Dynamic graphics includes the structured use of these techniques to enhance visual communication. Many people mistakenly assume that dynamic graphics and animation are only relevant to the production of cartoons or to dynamic simulations of complex three-dimensional objects. On the contrary, dynamic graphics can enhance and enrich any dialogue with a computer; it can fruitfully be applied to computer-aided instruction, computer simulation modelling, data base browsing and retrieval, and computer teleconferencing, to mention only a few examples.

5. DYNAMIC COMPUTER GRAPHICS

5.1. *Dynamic Computer Graphics on Vector Displays*

We begin by reviewing how dynamic graphics has been achieved with random scan displays. We shall concentrate primarily on true animation, and make only brief mention of the other cinematographic techniques listed in the preceding section. However, the same basic principles apply to most types of dynamic imagery.

Consider first the simple refreshed display depicted in Figure 2. Typically the display file is segmented, with distinct components of the picture having distinct representations. Thus, to move or transform a component, we need only recompute it, and then insert it into the display file in place of the original. Double buffering the segment means that the process of computation is not visible; if the replacement occurs quickly enough, continuous motion is achieved. This is almost always the case with the basic animation primitives of movement and selection, for they can be computed trivially, the former by replacing the segment origin which controls the absolute position of a picture otherwise specified in terms of relative coordinates, and the latter by switching pointers in the display file, thus causing one picture to be shown in place of another.

Similar principles apply to the more sophisticated systems depicted in Figures 3 and 4. Typical display processors of the former class include hardwired three-dimensional transformations which operate on tree-structured display files. This extends the class of possible real-time animation capabilities to include three-dimensional translation, rotation, and scale changes, and usually perspective computations. The buffered transformation processor of Figure 4 achieves the same result, but also provides a mechanism for trading off image update rate against image complexity.

Some processes of animation and dynamic graphics are not well handled by this technology. Key-frame animation must generally be carried out by the host CPU. Zooms and pans can be handled by the display processor provided that they have good windowing and clipping capabilities; otherwise these calculations must also be done by the CPU. Fades and dissolves depend upon use of the display's intensity control, which is usually very limited.

Nonetheless, because images can be added, moved, and removed essentially instantaneously, dynamic computer graphics as we know it has become the exclusive province of this technology. It is a goal of this paper to show that this need not be so.

5.2. *Dynamic Computer Graphics on Digital Video Displays*

5.2.1. *Technological Constraints*

Any attempt to achieve dynamic computer graphics on digital video displays must deal with the technological constraints imposed by the raster scan. Conventional television images are displayed at 30 frames/second. Each frame is composed of two interlaced fields to reduce flicker. For a system with 480 X 640 resolution on a conventional 525 line monitor, each scan line must be fetched in 1/525th of 1/30th of a second = 63.5 microseconds. Of the 525 lines, 480 are visible and 45 occur during the vertical flyback of the electron beam. Since horizontal flyback occurs in 11.4 microseconds, each raster point must be fetched or computed in 81 nanoseconds.

These figures have several serious implications. The first is on memory bandwidth. Refreshing the above picture out of a frame buffer requires a memory bandwidth of roughly 100 million bits per second. Reducing the resolution to 240 X 320 lowers this figure by a factor of two, because pairs of horizontally adjacent pixels are identical. Successive pairs of scan lines are also identical, but, because of interlace, this does not lessen the bandwidth problem, although it reduces by 50% the storage required. Of course, storage is a severe problem in itself, for a 512 X 512 X 8 bit frame buffer requires 2,000,000 bits of storage.

5.2.2. *Systems with Frame Buffers*

Current frame buffers are unsuited to dynamic imagery. Consider the problem of moving a figure across the screen. Assume a 512 X 512 randomly accessible frame buffer, and a CPU capable of generating a new raster point every 33 microseconds, which is typical of the 1979 state-of-the-art. In this case only 1000 of the 250,000 screen pixels can be recomputed in a 30th of a second. This means that 500 points on the last version of the figure can be erased, and 500 points can be added to the new version. Furthermore, it is difficult to guarantee that each new picture appear correctly, because the computation must be synchronized perfectly with the raster scan to guarantee that a change of a pixel value occur after that pixel has been displayed and before it is next displayed. This problem would be even worse is we were trying to enhance the speed of screen changes with a P1 containing a vector generator.

There is also another serious problem. Consider a black-and-white picture. Pixels often represent the intersection point of two or more picture elements. If we erase one of those elements, for example a line, we leave a gap in the others that must be refilled. Although simple in concept, the search for possible intersection points between a line to be removed and all other visible lines is a time-consuming process.

Nonetheless, simple stylized dynamic graphics is possible on random access frame buffers if the required picture changes can be localized to a small enough area. These changes can best be implemented by techniques such as superimposition, the pop-on, and the wipe. Furthermore, recent conceptual advances in our understanding of raster manipulation functions (see pp. 262-265 of [Newman 79]), and work in several laboratories on high-speed microcoded processors and on memory organizations that facilitate the parallel application of these functions to small regions of a bit map, should soon allow us to update raster points 2 or 3 orders of magnitude faster than described above.

5.2.3. *Systems with Coded Picture Definitions*

A digital video display with a coded picture definition expands a full raster scan image from a condensed representation "on the fly". Any change in the coded picture definition will immediately be visible on the screen. Hence a structured succession of changes to the display file will produce an animated sequence. For example, most of the display file of an alphanumeric video display terminal can be recomputed in a 30th of a second.

To facilitate computer animation on systems with this architecture, several conditions should be satisfied:

1) Pictures to be animated must be representable in the display file format. This will usually imply a significant reduction in generality, for the display file must be simple enough to be scan converted in real time by the processor P2 (see Figure 9).

2) The display file format should allow pictures to be stored compactly, as, for example, in the use of a run-coded display file for pictures that have relatively few changes along each scan line. The format should also allow easy computation of transformations that will result in dynamic sequences. These two conditions make it possible for the CPU to compute a new image fast enough to produce the illusion of movement.

3) The display file should be segmented. Thus changes to a subpicture can be affected by changing only its representation in the display file.

4) The display file should allow double buffering of segments. This will mean that a segment can be recomputed without concern for its synchronization with the scan conversion and display process. The new version will be made visible when and only when it is fully computed.

The coded picture definition architecture has one additional advantage. The problem of the restoration of intersection points referred to in the previous section disappears, because the full raster image is constructed anew each 30th of a second from the latest coded representation. It is always up-to-date, and contains no artifacts that reflect the order in which the CPU performed additions and erasures.

5.2.4. *Systems with Coded Picture Definitions and Double Frame Buffers*

The problem with the architecture of Figure 9 is the need for the display processor to keep up with the raster scan. If it cannot keep up, the picture will degrade, gracefully or chaotically, depending upon the sophistication of the design. In fact, if the display is producing animation, it may be capable of displaying some frames but not others. This would be particularly annoying.

The architecture of Figure 10 solves the problem. We separate the image expansion and raster scan refresh processes. Now, as the images become more complex, and each one takes longer to compute, the animation no longer degrades but merely slows down. This phenomenon was demonstrated on the Interim Dynabook over five years ago.

The architecture has another less obvious advantage. The coding mechanism used with Figure 9 has to be scan-line oriented so that scan lines can be decoded in synchrony with the raster scan. Buffering the image as in Figure 10 allows us to use other encodings, such as the hierarchic area encoding scheme of the Interim Dynabook. This is because it no longer matters in what order pixel values are computed.

The architecture yields one further opportunity. Because two complete images are buffered, we can include in processor P3 the capability to compute cinematographic effects that require combinations of two images. These include cuts, fade-ins, fade-outs, dissolves, double exposures, superimpositions, pop-ons, wipes, and close-ups. Such capabilities have already been seen in the most sophisticated recent frame buffers described in Section 3.2.

6. SOFTWARE IMPLICATIONS

1) There is an increasing body of opinion which holds that systems software for raster-scan graphics can be constructed in the same manner as that for random scan graphics [Sproull 75]. Primitives for generating and transforming line drawings need to be augmented with primitives for controlling line thickness, for shading regions, and for controlling opacity and transparency.

2) Systems software for simple animation, whether it is to be performed on vector displays or on digital video displays, requires a mechanism for organizing picture change with respect to a scale of movie time [Baecker 75]. As we have already seen in Section 4, the most essential picture changes are those of movement and selection. Movement of an image is used to portray the movement of the object it represents. Selection among multiple images (cels) is used to portray the behavior of the object it represents. Thus rapid and appropriate switching among different views of a moving object will animate it, will "bring it to life".

To implement such animation, the software must contain mechanisms for the storage management of picture segments, that is, for creating and deleting them, and for making them visible and invisible. Tools for translating segments are also required. Movement of an object is accomplished by translating its picture segments; selection among various views of an object is accomplished by making one segment visible as another segment is made invisible.

3) What is required for more sophisticated dynamic graphics is a standard high-level graphics programming language. There is currently much consensus in the industry about what capabilities are required in such a language, as can be seen by the graphics standards efforts now underway [Core 77]. The capabilities include tools for applying two- and three-dimensional transformations, for concatenating these transformations, and for windowing and clipping. We do not propose that these be carried out in hardware on images in raster format. With the exception of two-dimensional windowing and clipping, which could be embedded in the scan converting processor as was demonstrated in the Interim Dynabook, these capabilities should usually be carried out by software in the CPU.

7. SUMMARY AND CONCLUSIONS

7.1. *Summary*

We have presented a methodology for describing digital video displays which clarifies the effect of system architecture on capabilities and performance. The methodology decomposes systems into configurations of processors with certain powers and memories with certain storage capacities and access characteristics. The technique has proved adequate to describe all known examples of digital video displays and to suggest design possibilities for new systems.

More specifically, the methodology helps us understand why dynamic graphics is not possible on most existing digital video displays, namely, that the method whereby frames are represented requires too much computation by processors not specialized to the task, and, furthermore, the frames are not double buffered to prevent the viewer from perceiving the process of computation. Two other architectures were found more useful for dynamic graphics. In the first, a digital video display processor expands a picture "on the fly" from a condensed encoded representation. Although limits are placed on the classes of pictures that can be handled, animation of simple images is easy to achieve. The other architecture separates the computation from the refresh processes by buffering two frames between the two processes. This removes the restriction on generality: complex images can now be represented, although their animation may proceed more slowly than desired.

7.2. *Conclusions*

1) Dynamic graphics is feasible on digital video displays.

2) The frame buffer (Figure 8) is not a desirable display system architecture for dynamic graphics. However, animation of simple images can be achieved by a display processor which scan converts and expands pictures "on the fly" from a condensed encoded representation (Figure 9). Limits on the classes of pictures that can be handled by such a system can be relaxed by providing a double frame buffer after the display processor (Figure 10).

3) Only movement and selection of images, and possibly two-dimensional windowing and clipping, should be carried out in hardware by the scan converting processor (P2 in Figures 9 and 10). Other more sophisticated two- and three-dimensional transformations should be carried out in software by the controlling CPU.

4) Although our approach clarifies the qualitative behavior of existing and possible systems, it does not yield quantitative data in its current form. It should be embodied into a simulation methodology which could be used to experiment with and evaluate new designs. This methodology would accept specifications of processor instruction sets and execution times, and memory sizes and access characteristics, and produce performance measures of the system on randomly constructed and specifically designed pictures. Such a methodology is currently under development at the University of Toronto's Dynamic Graphics Project [Galloway 79].

5) Architectures for digital video display systems that can support dynamic graphics have been identified, but few instances of such systems exist. Implementations need to be carried out to refine our understanding of the design concepts described in this paper. An example of such a system with the architecture of Figure 10 is currently under development at the Dynamic Graphics Project [Miller 79a,b]. The system will scan convert into a double frame buffer a segmented display file consisting of relatively sparse dot drawings. Since these drawings can be translated and selected with ease, real-time animation will be possible.

Acknowledgments

This paper is based in part on work carried out at the University of Toronto under the sponsorship of the National Research Council of Canada, and in part on work carried out at Human Computing Resources Corporation under the sponsorship of the Canadian Department of Communications. I am grateful to Herb Bown, Tom Horsley, Stan Kriz, David Miller, Doug O'Brien, Brian Rosen, Dick Shoup, Bob Sproull, Mike Tilson and Walter Vynohrad for helpful comments on various drafts at various stages of this work, and to Dave Sherman for editorial, bibliographic, and typographic assistance.

References

[Andrews 74] Donald I. Andrews, "Line Processor -- A Device for Amplification of Display Terminal Capabilities for Text Manipulation", *Proceedings of the NCC*, 1974, pp. 257-265.

[Aziz 67] R.A. Aziz, "An Instructional Display Terminal", *Proceedings of the Eighth National Symposium on Information Display*, May 1967, San Francisco, pp. 83-90.

[Baecker 69a] Ronald M. Baecker, *Interactive Computer-Mediated Animation*, Ph.D. Thesis, M.I.T. Dept. of Electrical Engineering, Project MAC Technical Report 61, April 1969.

[Baecker 69b] Ronald M. Baecker, "Picture-Driven Animation", *Proceedings of the SJCC*, May 1969, pp. 273-288.

[Baecker 74] Ronald M. Baecker, "GENESYS--Interactive Computer-Mediated Animation", appears in *Computer Animation* (John Halas, Editor), Hastings House, New York, 1974, pp. 97-115.

[Baecker 75] Ronald M. Baecker, "A Computer Animation Facility for Research and Educational Filmmaking: Design and Application", *Proceedings of the Fourth NRC Man-Computer Communications Conference*, May 1975, pp. 19.1-19.11.

[Baecker 76] Ronald M. Baecker, "A Conversational Extensible System for the Animation of Shaded Images", *Computer Graphics*, Volume 10, Number 2, Summer 1976, pp. 32-39.

[Barenholtz 76] J. Barenholtz and C. Dewhurst, "A Run Length Encoding Scheme for Real Time Video Animation", *Proceedings of the Northwest 76 ACM/CIPS Regional Symposium.*

[Barrett 74] R.C. Barrett and B.W. Jordan, Jr., "Scan Conversion Algorithms for a Cell Organized Raster Display", *Communications of the ACM,* Volume 17, Number 3, March 1974, pp. 157-163.

[Baskett 76] Forest Baskett and Leonard Shustek, "The Design of a Low Cost Video Graphics Terminal", *Computer Graphics,* Volume 10, Number 2, Summer 1976, pp. 235-240.

[Bown 78] H.G. Bown, C.D. O'Brien, W. Sawchuk, and J.R. Storey, "A General Description of Telidon: -- A Canadian Proposal for Videotex Systems", CRC Technical Note No. 697-E, Communications Research Centre, Canadian Department of Communications, Ottawa, December 1978.

[Burtnyk 77] N. Burtnyk, P. Carey, K. Steele, and M. Wein, "A Video Terminal for Interactive Graphics", *Proceedings of the Fifth Man-Computer Communications Conference,* May 26-27, 1977, Calgary, pp. 31-44.

[Core 77] Graphics Standards Planning Committee of ACM/SIGGRAPH, Status Report, *Computer Graphics,* Volume 11, Number 3, Fall 1977.

[Denes 74] Peter B. Denes, "Computer Graphics in Colour", *Bell Laboratories Record,* Vol. 52, May 1974, pp. 138-146.

[Denes 75] Peter B. Denes, "A Scan-type Graphics System for Interactive Computing", *Proceedings of the Conference on Computer Graphics, Pattern Recognition, and Data Structure,* IEEE Computer Society, May 14-16, 1975, pp. 21-24.

[Eastman 75] Jeffrey F. Eastman and David R. Wooten, "A General Purpose, Expandable Processor for Real-Time Computer Graphics", *Computers and Graphics,* Volume 1, Number 1, 1975, pp. 73-77.

[Entwisle 77] Jeffrey Entwisle, "An Image Processing Approach to Computer Graphics", *Computers and Graphics,* Volume 2, Number 2, 1977, pp. 111-117.

[Fischer 73] Michael Fischer, "MAPS -- A Generalized Image Processor", *Computer Graphics,* Volume 7, Number 3, Fall 1973, pp. 1-9.

[Fischer 75] Michael A. Fischer and Robert E. Nunley, "Raster Graphics for Spatial Applications", *Computer Graphics,* Volume 9, Number 2, Summer 1975, pp. 1-8.

[Galloway 79] David R. Galloway, *The Modelling of Dynamic Digital Video Display Systems,* M.Sc. Thesis, Dept. of Computer Science, University of Toronto, 1979 (forthcoming).

[Grover 71] D.J. Grover, "Low cost graphic display with serial access store", *Computer Bulletin,* Volume 15, Number 1, January 1971, pp. 33-37.

[Halas 71] John Halas and Roger Manvell, *The Technique of Film Animation,* Focal Press, 3rd Ed., 1971.

[Hendrickson 67a] Herbert C. Hendrickson, "The display/control complex of the Manned Space Mission Control Center", *Information Display,* May/June 1967, pp. 52-58.

[Hendrickson 67b] Herbert C. Hendrickson, "A high-precision display system for command and control", *Information Display,* July/August 1967, pp. 32-36.

[HP 75] Articles on the Model 2640A Interactive Display Terminal Family, *Hewlett-Packard Journal,* June 1975, Palo Alto Ca.

[HP 78] Articles on the Model 2648A Graphics Terminal, *Hewlett Packard Journal,* January 1978, Palo Alto Ca.

[Honey 71] Francis J. Honey, "Computer Animated Episodes by Single Axis Rotations", *Proceedings of the Tenth Annual Meeting of UAIDE,* 1971, pp. 3-210--3-226.

[Hunter 75a] Gregory M. Hunter, "Full-colour Television from the Computer, Refreshed by Run-length Codes in Main Memory", Technical Report Number 182, Computer Science Laboratory, Princeton University, April 21, 1975.

[Hunter 75b] Gregory M. Hunter, "Computer Animation and Run-length Codes", Technical Report Number 199, Computer Science Laboratory, Princeton University, February 25, 1975.

[Irby 74] Charles H. Irby, "Display Techniques for Interactive Text Manipulation", *Proceedings of the NCC,* 1974, pp. 247-255.

[Jordan 73] B.W. Jordan, Jr. and R.C. Barrett, "A Scan Conversion Algorithm with Reduced Storage Requirements", *Communications of the ACM,* Volume 16, Number 11, November 1973, pp. 676-682.

[Jordan 74] B.W. Jordan, Jr. and R.C. Barrett, "A Cell Organized Raster Display for Line Drawings", *Comunications of the ACM,* Volume 17, Number 2, February 1974, pp. 70-77.

[Kajiya 75] James T. Kajiya, Ivan E. Sutherland, and Edward C. Cheadle, "A Random-Access Video Frame Buffer", *Proceedings of the Conference on Computer Graphics, Pattern Recognition, and Data Structure,* IEEE Computer Society, May 14-16, 1975, pp. 1-6.

[Kay 74] Alan Kay, Presentation at the First Annual Conference on Computer Graphics and Interactive Techniques, Boulder Colorado, July 1974.

[Kubitz 69] W.J. Kubitz and W.J. Poppelbaum, "The Tricolor Cartograph: a display system with automatic coloring capabilities", *Information Display,* November/December 1969, pp. 76-79.

[Laws 75] B.A. Laws and W.M. Newman, "A Gray-scale Graphic Processor using Run-length Coding", *Proceedings of the Conference on Computer Graphics, Pattern Recognition, and Data Structure,* IEEE Computer Society, May 14-16, 1975, pp. 7-10.

[LRG 76] Learning Research Group, *Personal Dynamic Media,* Xerox Palo Alto Research Center, 1976.

[Macaulay 68] Malcolm Macaulay, "A Low Cost Computer Graphic Terminal", *Proceedings of the FJCC,* 1968, pp. 777-785.

[Madsen 69] Roy Madsen, *Animated Film: Concepts, Methods, Uses,* Interland Publishing Inc., 1969.

[Malik 78] Rex Malik, "The TV as a Terminal", *Datamation,* March 1978, pp. 213-215.

[Matherat 78] Philippe Matherat, "A Chip for Low-Cost Raster-Scan Graphic Display", *Computer Graphics,* Vol. 12, No. 3, August 1978, pp. 181-186.

[McCracken 75] T.E. McCracken, B.W. Sherman, and S.J. Dwyer III, "An Economical Tonal Display for Interactive Graphics and Image Analysis Data", *Computers and Graphics,* Volume 1, Number 1, 1975, pp. 79-94.

[Metzger 69] Richard A. Metzger, "Computer generated graphic segments in a raster display", *Proceedings of the SJCC,* 1969, pp. 161-172.

[Miller 79a] David H. Miller, David R. Galloway, William T. Reeves, Ronald M. Baecker, and H. Dominic J. Covvey, "SPIWRIT: A Proposed Dynamic Colour Video Graphics Device", Computer Systems Research Group, University of Toronto, revised version, 1979.

[Miller 79b] David H. Miller, *Two Dimensional Video Animation Processors,* M.A.Sc. Thesis, Dept. of Electrical Engineering, University of Toronto, 1979 (forthcoming).

[Myers 76] Allan J. Myers, "A Digital Video Information Storage and Retrieval System", *Computer Graphics,* Volume 10, Number 2, Summer 1976, pp. 45-50.

[Negroponte 77] Nicholas Negroponte, "Raster Scan Approaches to Computer Graphics", *Computers and Graphics,* Volume 2, Number 3, 1977, pp. 179-193.

[Nelson 77] Theodore H. Nelson, *The Home Computer Revolution,* The Distributors, 702 South Michigan, South Bend, Indiana 46618, 1977.

[Newman 74] William M. Newman and Robert F. Sproull, "An Approach to Graphics System Design", *Proceedings of the IEEE,* Volume 62, Number 4, April 1974, pp. 471-483.

[Newman 79] William M. Newman and Robert F. Sproull, *Principles of Interactive Computer Graphics,* 2nd Edition, McGraw-Hill, 1973.

[Noll 71] A. Michael Noll, "Scanned-Display Computer Graphics", *Communications of the ACM,* Volume 14, Number 3, March 1971, pp. 143-150.

[Ophir 68a] D. Ophir, S. Rankowitz, B.J. Shepherd, and R.J. Spinrad, "BRAD: The Brookhaven Raster Display", *Communications of the ACM,* Volume 11, Number 6, June 1968, pp. 415-416.

[Ophir 68b] D. Ophir and B.J. Shepherd, "ITN -- Data Terminal Network Computer System", *Nuclear Instruments and Methods 62,* 1968, pp. 285-294.

[Pearson 75] D.E. Pearson, *Transmission and Display of Pictorial Information,* Halsted Press, John Wiley & Sons, 1975.

[Preiss 78] Richard B. Preiss, "Storage CRT Display Terminals: Evolution and Trends", *Computer,* Vol. 11, No. 11, November 1978, pp. 20-26.

[Rose 67] Gordon A. Rose, "Intergraphic -- A Microprogrammed Graphical-Interface Computer", *IEEE Transactions on Electronic Computers,* Volume EC-16, Number 6, December 1967.

[Rose 68] Gordon A. Rose, "Computer Graphics Communication Systems", *Proceedings of the IFIP Conference,* 1968, pp. 211-220.

[Roseman 77] Ellen Roseman, "Video Games Are Back With A New Wrinkle", *The Globe and Mail,* Nov. 26, 1977, p. 4.

[Rougelot 69] R.S. Rougelot, "The General Electric Computer Color TV Display", appears in *Pertinent Concepts in Computer Graphics* (M. Faiman and J. Nievergelt, Editors), University of Illinois Press, 1969.

[Ruder 68] Don Ruder, "Data Disc Television Display System", *Proceedings of the UAIDE Annual Meeting,* 1968, pp. 338-357.

[Saxton 77] William A. Saxton and Morris Edwards, "Data Base Access Through Home TV Is Now A Reality", *Canadian Datasystems,* December 1977, pp. 56-57.

[Shoup 74] Richard Shoup, Presentation at the First Annual Conference on Computer Graphics and Interactive Techniques, Boulder Colorado, July 1974.

[Sproull 75] Robert F. Sproull and William M. Newman, "The Design of Gray-scale Graphics Software", *Proceedings of the Conference on Computer Graphics, Pattern Recognition, and Data Structure,* IEEE Computer Society, May 14-16, 1975, pp. 18-20.

[Staudhammer 75] John Staudhammer and Deborah J. Ogden, "Computer Graphics for Half-tone Three-dimensional Object Images", *Computers and Graphics,* Volume 1, Number 1, 1975, pp. 109-114.

[Taylor 77] M.M. Taylor and J.A. Norton, "A Versatile Colour Raster Display System for Visual Psychophysics and Image Processing", *Proceedings of the Digital Equipment Computer Users Society (Canada),* February 1977, pp. 923-929.

[Tek 78a] Articles on the Tektronix 4025 Computer Display Terminal in *Tekscope,* Volume 10, Number 1, 1978, Tektronix Inc., Beaverton Or.

[Tek 78b] Articles on the Tektronix 4027 Computer Display Terminal in *Tekscope,* Volume 10, Number 4, 1978, Tektronix Inc., Beaverton Or.

[Terlet 67] R.H. Terlet, "The CRT Display Subsystem of the IBM 1500 Instructional System", *Proceedings of the FJCC,* 1967, pp. 169-176.

[Thornhill 74] Daniel E. Thornhill and Thomas B. Cheek, "Raster-scan tube adds to flexibility and lower cost of graphic terminal", *Electronics,* February 7, 1974, pp. 95-101.

[Tilson 76] Michael D. Tilson, *Editing Computer Animated Film,* Technical Report CSRG-66, Computer System Research Group, University of Toronto, January 1976.

[Tripp 75] Edwin R. Tripp III and Jimmie R. Suttle, "An Interactive Three-Dimensional Color Graphics System", *Computers and Graphics,* Volume 1, Number 2/3, 1975, pp. 211-214.

[Uncapher 71] K.W. Uncapher, "The RAND Video Graphics System -- An Approach to a General User-Computer Graphic Communication System", RAND Report P-753-ARPA, April 1971.