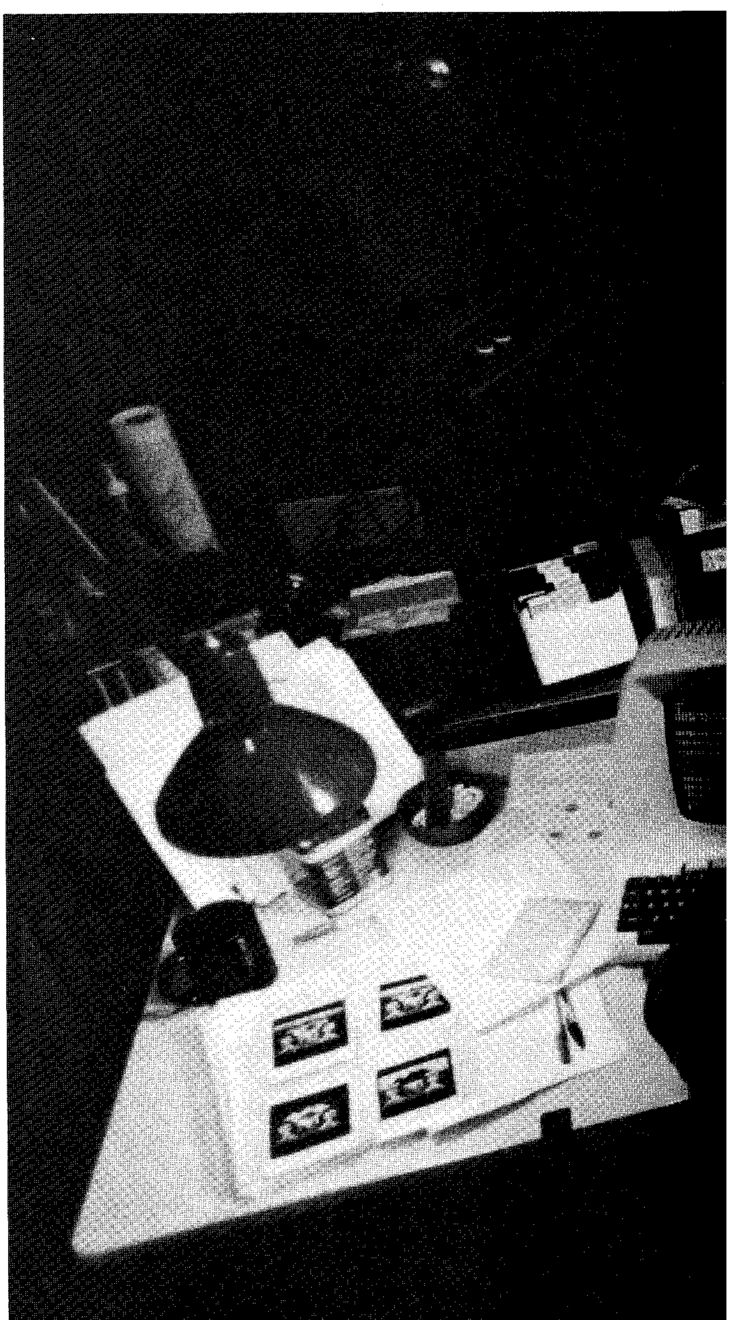
COMPUTER ART as a WAY OF LIFE

By Gene Youngblood

When I think of computer art I think of Chicago, and of four people there—Jane Veeder, Phil Morton, Dan Sandin and Tom De Fanti—who are pioneering a visual art form that ultimately is not visual at all, but rather the creation of language, and of conversational "environments" out of which will emerge our future images and the images of our future. Together with Steina and Woody Vasulka in Santa Fe, they inspired me, taught me, and changed me profoundly over the last ten years. With infinite patience and dedication, with the passion of visionary seekers, they guided me ever deeper into the digital domain and caused me at last to understand that computer graphics is something more than art, that it is a kind of practical philosophy, a way of life, a way of being in the world and a way of creating a world to be in.

No story reflects this more vividly than that of Jane Veeder, an artist-programmer whose life and work epitomize both the unique computer-art community in Chicago and a personal pathway of growth and discovery that will become representative of the life of the artist in our time. In my opinion, Veeder stands with Ed Emshwiller and Larry Cuba as one of the most gifted computer artists working in America today. Relatively unknown until recently, she is beginning to get the recognition she deserves. Her 1982 animation *Montana* is the only work of computer graphics in the Museum of Modern Art's video collection, and her interactive paint program/arcade game *Warpitout*—the sensation of the SIGGRAPH '82 Art Show in Boston—will later this year be installed at the Ontario Science Center in Toronto, one of the most prestigious science museums in the world.

Gene Youngblood was a member of the jury for the SIGGRAPH '83 Exhibition of Computer Art. He is working on a new book, *The New Renaissance: The Computer Revolution and the Arts.*



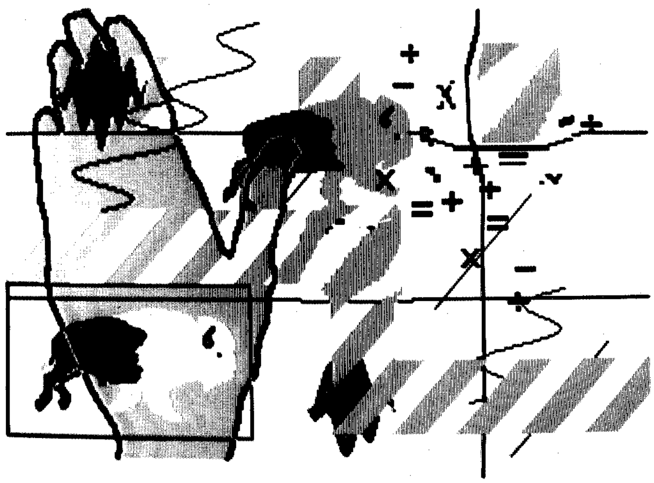
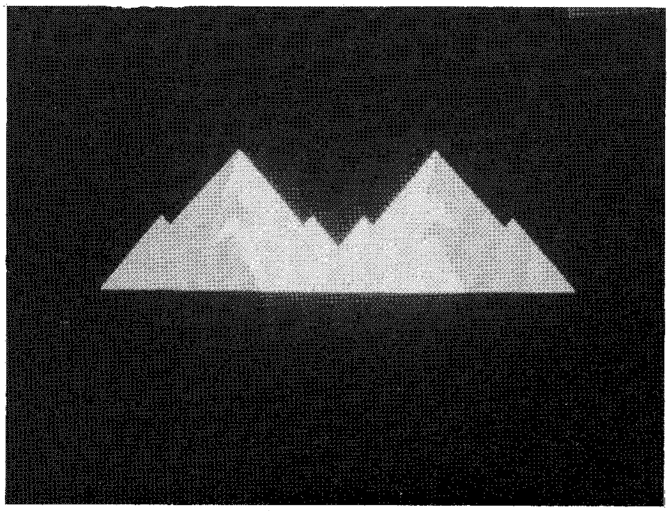
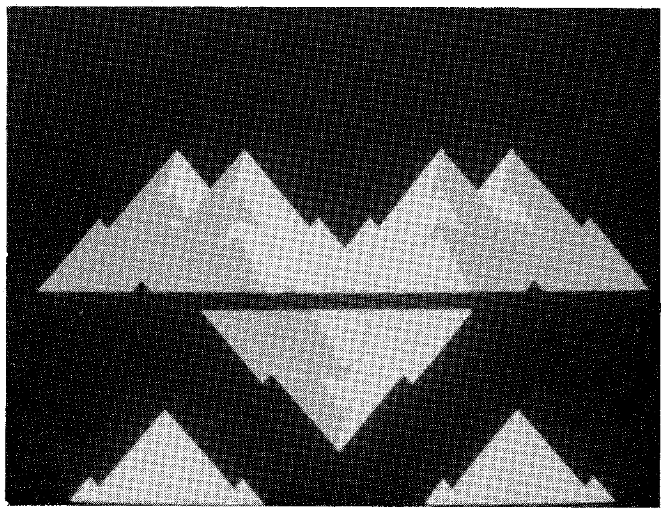


Photo credit: Chip Dodsworth

"The computer is the future . . . and remain autonomous and at the same connected with the future . . ."





MONTANA: A constellation of totem-like icons dancing on a dreamworld stage.

The road to success has been a long one for 39-year-old Veeder, whose personal journey as an artist—involving excursions into theatre, painting, ceramics, sculpture, filmmaking and video—reflects a widespread disenchantment with traditional art forms, a conviction that the leading edge of culture is no longer defined by what's exhibited in art galleries or purchased by museums. "The go-nowhereness of traditional art forms just kills me," she said. "First painting. The go-nowhereness of painting killed me off it. And then ceramics. I did ceramics for ten years and after a while you sort of know everything. You start reinventing the wheel. First there was abstract ceramics, then sculptural ceramics, then utilitarian, then Japanese, then you run out of things and you go around and around. But with electronic stuff, especially computers, there's always so much you don't know about. It's this incredible carrot that just keeps hanging out there in front of you. The computer is the future of everything and it's a tool for building tools: you can remain autonomous and at the same time literally create the tools that keep you connected with the future of the world."

Veeder's migration from the analog world of video synthesis into the digital domain of computer synthesis is a trailblazing journey that more and more electronic artists will follow as the advantages and satisfactions of computer life are recognized. "I like video but it's very expensive, so I couldn't afford personal tools. Also video skills don't get much in the way of rewards. Video's very easy to do. Many people have these skills. It's not very strategic. But having my own computer graphics system has made a huge difference. It's like doing video without the videotape recorder. Videotape recorders are horrible machines. They're complex, they're expensive, they don't last very long. That whole electromechanical combination is problematic.

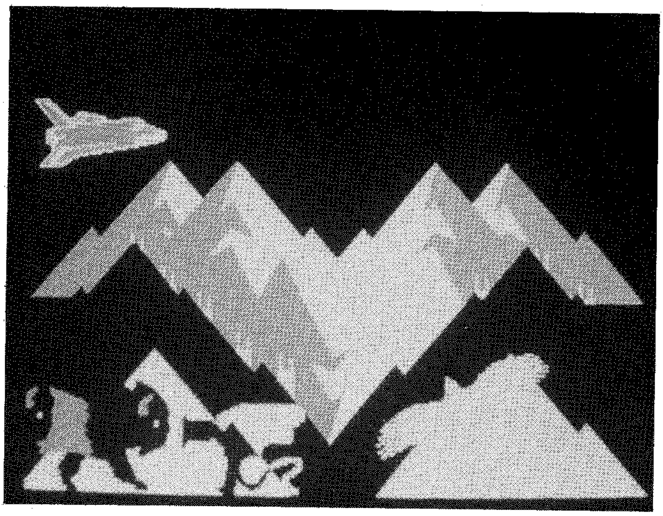
"Also, being collaborative, video always involves hassles with other people. But with a computer, if the software is any good at all, the conversation is incredibly direct. It's encouraging to know that if you keep trying you can make it work. You actually are encouraged to apply your energy. That's so satisfying. How many relationships have you had with humans where you knew that no matter how hard you tried it just wasn't going to get any better? So on one hand I just want to talk to the computer because we understand each other so well. And on the other hand it gives me a whole other level on which to communicate with people. With the computer I can get at my dreamlike images; I can visualize my made-up pictures with a lot of power and control and detail; there's this detail tweakability that enables me to get at this weird place; the computer offers more control in support of that space than video ever did."

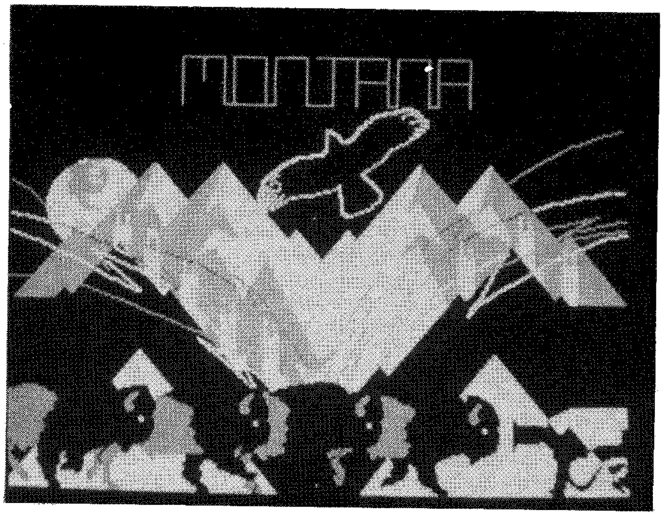
ZGRASS: AN ENVIRONMENT FOR ARTISTS

Veeder works in ZGRASS, a language written by Tom De Fanti exclusively for real-time interactive motion graphics. Incorporating the best features of Pascal, Smalltalk, Lisp and BASIC, ZGRASS is by all accounts the best graphics language around. Among other things, it is self-teaching and user-extensible—the users can create their own commands and integrate these commands into the language, expanding and evolving it to fit their personal style of making images.

ZGRASS runs only on the Datamax UV-1 Graphics Computer which contains three custom chips designed for Bally Arcade games like *Wizards of War* and *Gorf*—one chip for processing instructions, one for controlling the display, and one for sound synthesis. This makes ZGRASS faster than anything around at its price (about $10,000), and speed is essential for real-time animation. There's also a 16-screen memory that allows you to create and store sixteen different full-screen images and then call them up in a real time animation cycle. Output is standard NTSC video. Veeder works with a single Sony 5850 videotape recorder and builds her editing commands into her graphics programs. "Every tape can be first-generation," she explained. "It's just as easy to run the computer and make a first-generation tape as it is to do a dub of the master. And with the 16-screen memory I

it's a tool for building tools: you can time create the tools that keep you





can work up the individual screens and then program them sequentially and the animation just happens."

The UV-1 offers less resolution and fewer colors than larger, more expensive systems—only 320 x 202 x 2 bits per pixel (256 colors with four colors per area). This represents a tradeoff in favor of real time operation while keeping the cost down; but if you're moving images you don't need that much resolution anyway; moreover, although Hi-Res has been elevated to a quasi-religion in the commercial graphics world, the fact is that for many artists real-time interaction is far more important. "Photographic realism is not my goal," Veeder explained. "I can see its utility but I don't think it would be a triumph, you know. I have no desire to use a super-high resolution system because I don't draw in super-high resolution. I just go so far and beyond that it's the integration and manipulation that I enjoy—movements and gestures."

In fact, animation *per se* is not the primary motivation for Veeder's involvement with ZGRASS. "I'm just interested in real time graphic interaction that results in a dynamic visual event," she explained. "What I want is to integrate my eye, hand and brain with the computer's ability to perform complex relationships very rapidly. The fact is I'm addicted to the high-speed personal evolution and perceptual education you get from continuous contact with a real-time interactive machine intelligence." Nevertheless, *Montana* and Veeder's most recent computer animation, *Floater*, rank with the best works of experimental hand-drawn animation. The quirky, totemic, dreamlike, cyclical image-events of *Montana* are reminiscent of Harry Smith's *Early Abstractions*, whereas *Floater* inspires comparison with Robert Breer, Paul Glabicki, and the optical-printing films of Pat O'Neill. Both works are essentially autobiographical.

MONTANA

For five years, Veeder and Phil Morton spent their summers camping out and making video in the mountains of the Western Badlands—Montana, Wyoming, Utah. The trips were part of a unique lifestyle approach to video, a desire to live with and through the medium conversationally, incorporating computer graphics into semi-didactic "simulations" of imagined and desired video realities. They "processed their life" in the electronic domain, producing video "communiques" of the realities they created by living and processing them. The visual signature of these tapes is absolutely unique, whith Jane's poignant, ideosyncratic computer graphic complementing Phil's sublime image processing—all of this superimposed over, and integrated into elegant black-and-white tapes of animal life and geological splendors of the American wilderness. The tapes have a joyous, chatty, conversational, grass-roots flavor that resonates against the otherworldly strangers-in-a-strange-land consciousness that pervades them. They are pioneering works by two genuine pioneers of the Electronic Life.

Gradually Jane evolved away from video and deeper into computer graphics. She and Morton no longer work together, but her love of the wilderness, and much of the graphic material she evolved for the videotapes, continue to inform her work in computer animation. This imagery—soaring hawks, mountain peaks, stampeding buffalo, erosion patterns in the terrain—are combined in *Montana* with icons of the technological world: video cameras, the Sears Tower in Chicago, the Space Shuttle. Veeder organized the material into 16-screen sequences programmed as a loop that increases in speed over eight cycles, with each of the elements moving at different speeds. The effect is a constellation of totem-like icons dancing on a dreamworld stage—hence the similarity to Harry Smith. Equally remarkable is the stereo soundtrack, which consists of an audiotape of birdcalls from the Audobon Society combined with sound synthesis performed on the Sandin Image Processor. The effect of the narrator numbering and naming the birdcalls in a cascade of electronic echoes perfectly complements the imagery and gives the tape a poignant, haunting, otherworldly flavor.

"Being able to earn my living by sitting at home with my intelligent machine . . . the sense of power and control over my life is immense."

WARPITOUT an installation in the form of an arcade game whose target is everyone's favorite image—their own, sensation at SIGGRAPH '82.

FLOATER

As impressive as *Montana* certainly is, the six-minute *Floater* (1983) is an even more sophisticated work, a tour-de-force of subtle animation, a masterpiece of simplicity, nuance and economy of gesture. The complex canvas of *Montana* is here reduced to a minimal set of emblematic objects which serve, in Veeder's words, "either as foils for dynamic graphic processes or manifest a particular motion behavior intimate to their identity." All that is left of the Badlands are two buffalo that gallop in place and two simple gestures that suggest mountains, water and reflections: squiggly lines and a diamond shape that transforms into a ripple and floats offscreen. (The ripple, which Veeder refers to as a "frequency being," also represents a sinewave, a natural phenomenon found not only in the electronic domain but in water and erosion patterns as well.)

In place of the video camera are two simple icons that represent Veeder's evolution into computer graphics: a rotating constellation of mathematical signs floating cloudlike in a black void, and two white lines, one vertical, one horizontal, manifesting the x-y coordinates of the two-dimensional graphic space. A few diagonal grid patterns constitute the only other imagery, except for the outline of the artist's hand that periodically "wipes" the screen as a transitional device. *Floater* ends with a remarkable figure-ground reversal as a grid of black squares emerges from the black background, obliterating the image: the ground becomes a figure that is indistinguishable from the ground, and nothing is left.

The lyrical economy of gesture in *Floater* is strongly reminiscent of Robert Breer's *Gulls and Buoys*, while the rhythmic clustering of iconographic objects in a black void is as effective as Paul Glabicki's *Five Improvisations*; and the cyclical orchestration of the whole dance calls to mind Pat O'Neill's earlier films like *Downwind* and *Runs Good.* The sound for *Floater* was created simultaneously with the animation using the audio-synthesis chip (originally used for sound effects in those Bally arcade games) that is resident in the UV-1 system; in other words, the audio was built into the animation programs, sharing variables with them. The result is a beautiful composition of small, sensitively-chosen sonic events (Veeder calls them "skeletal figure-sounds"), punctuated with intervals of silence, that have a droning/ringing quality like musical Morse Code. It is a reminder that she is as talented a sound composer as she is a gifted visual artist.

Ultimately, of course, the images and sounds are all that matter; they are the whole point, the reason why Veeder is simply a good artist, computer or no computer. Indeed, art is always independent of the medium through which it is practised: the domain in which something is deemed to be art has nothing to do with how it was produced. The use of the computer in the creation of animated graphics does not suddenly transform this noble tradition into "computer art"—it remains animation, whose status as art will ultimately be determined by art-historical concerns relevant to animation, not by any consideration of the computer as a creative instrument.

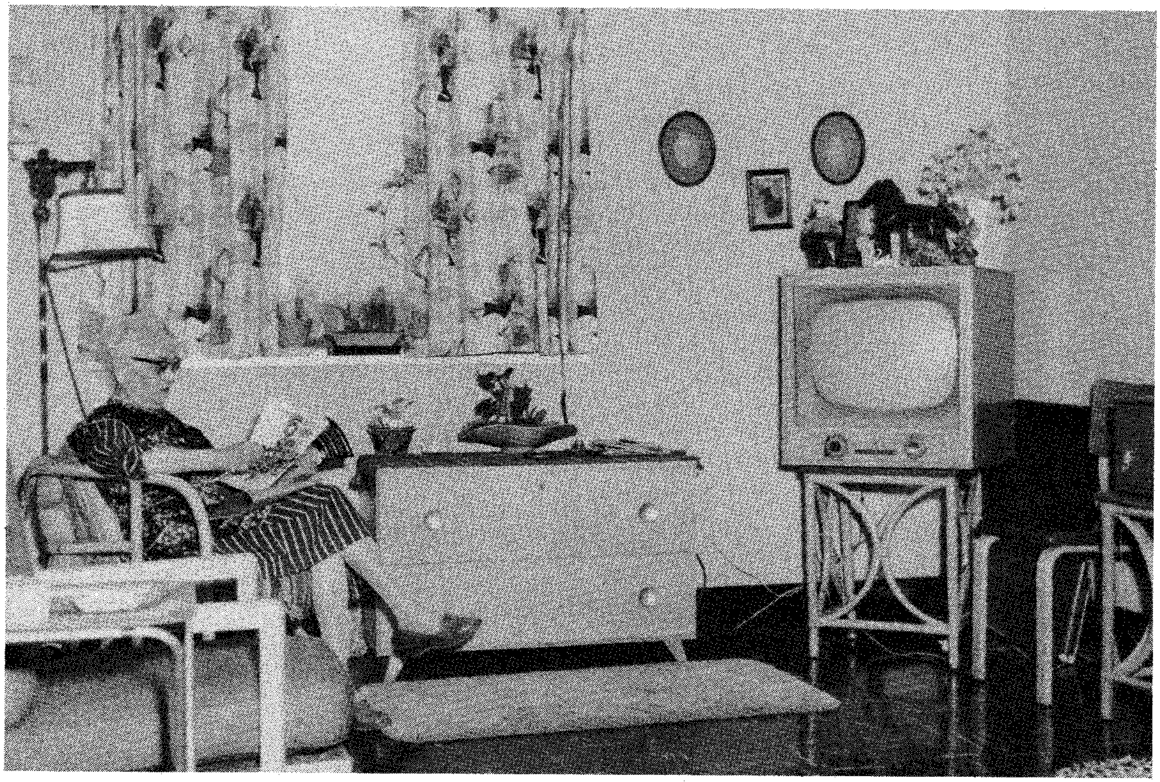
WARPITOUT

If there is an art unique to the computer, I suspect it will not have much to do with producing a drawing or sculpture or videotape; indeed it might not involve producing anything at all—for what is most unique about the computer is precisely its intelligence, that is, its *interactivity*. In other words, the great value of the computer is *ontological* rather than phenomenological—it has more to do with processes of being in the world (ontology) than with the commentaries and judgements we make as a consequence of being here (aesthetics, phenomenology). This is repeatedly confirmed by computer artists themselves, whose testimonies are almost always ontological, seldom aesthetic—always about the processes of producing art through conversation with the intelligent machine rather than about the art itself. This is what Dan Sandin means when he says that one cannot understand computer art by looking at it. And it is why *Warpitout* is such a brilliant approach to the idea of computer graphics.

Warpitout is an interactive installation in the form of an arcade game whose target is everyone's favorite image—their own. A menu-driven environment of programs written by Veeder in ZGRASS allows the "player"

Continued on p. 29 ▶

SAN FRANCISCO VIDEO FESTIVAL CATALOG



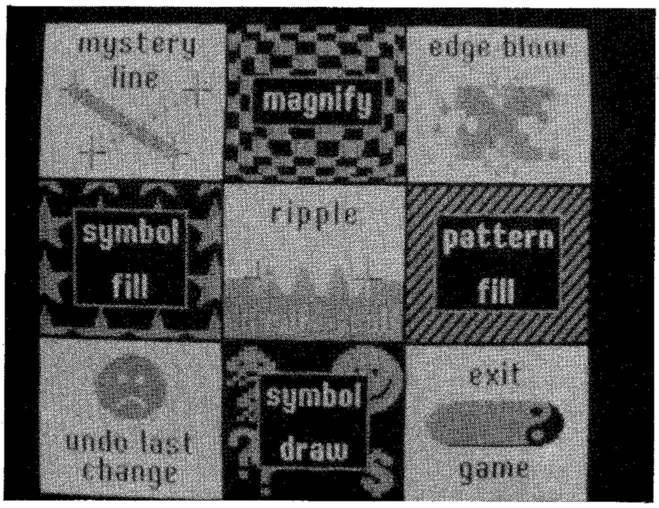
THE 1983 SAN FRANCISCO VIDEO FESTIVAL

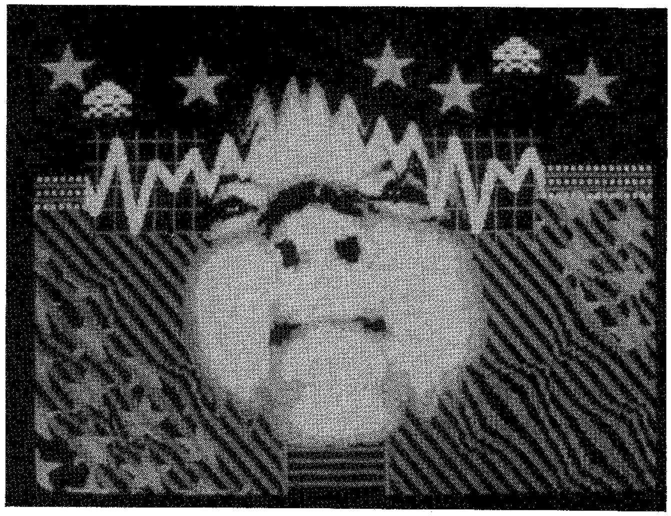
OCTOBER 6 TO 15

Theatrical use of prop and stage device; manipulation of the narrative form; computer animation; television production and documentary; and works expressive of the video medium's fine arts tradition: this array of formal approaches promises to free artists' video from both the iconoclastic stranglehold of the seventies and the hardware fascination of the past few years. We may all rejoice.

The diversity of style found in contemporary artists' video is reflective of the variety of traditions now influencing production. The recent recognition of the power and significance of artists' video has attracted creative workers from many fields. (Admittedly an awkward admission from a reticent art world struggling with its own often questionable economic traditions.) Each group is crafting its own special genre of expression.

The catalog which begins on the next page fully represents this new body of art, with twenty-four videotapes and collections, installations and performances:





of *Warpitout* to make computer art with an image of their own face, experiencing interactive graphics much more directly than is possible in the restricted, essentially one-way environment of video games. The system is housed in a *Wizards of War* cabinet that contains a 21-inch TV screen with a black-and-white video camera mounted above it, aimed at the player. There's a joystick and two buttons; push one and your face is digitized. ("It's one of the few things I've ever done that only makes a still picture.") Next you select a color map and the image is colorized. Then you are presented with a main menu consisting of nine options for visually processing the image. The joystick is used to select a particular option. Some of them, such as *Symbol Draw, Symbol Fill* and *Pattern Fill*, cause secondary menus to be displayed; from these you either select objects to be drawn as foreground figures, or fill in areas of the image with symbols or patterns which thus become a background field. Still other options, like *Magnify, Ripple* and *Edge Blow* modify the image geometrically. For example, the *Ripple* option redraws a swatch between two points designated by the player using the joystick; the redrawing process is modulated by a sine wave, rippling the image.

Unfortunately, the arcade game metaphor of *Warpitout* will remain only a metaphor: it's too expensive to manufacture and too slow for commercial environments. "One-way stuff like arcade games have to be fast and snappy because one-way is boring," Veeder pointed out. "Also, arcades want to kill the kids off quickly so they can get the next quarter. But *Warpitout* doesn't destroy you—*you* get to destroy you, and you get to sit there and watch the computer do it. That's the joy of the whole thing. It's fascinating to watch it all unfold before you. It gives you this tremendous sense of power."

AUTONOMY: THE COMPUTER LIFESTYLE

Power is a word used frequently in reference to computers, and for good reason, since a computer is a universal machine that can contain and become all media, conferring autonomy upon the user and erasing the distinction between professional and amateur insofar as that is determined by the tools to which we have access as autonomous individuals. Jane Veeder is among a rapidly-growing population of computer programmers who, by working at home with powerful personal tools, exemplify the emerging 'electronic cottage' lifestyle.

"I never visualized my future in career terms," she said, "you know, jobs. My images were always of pure lifestyle with no visible means of support. Pure lifestyle issues. I've never been able to get into the Professional world for more than a year at a time. I get dissatisfied. I can't possibly integrate those junk jobs into my life. But I *can* integrate the computer into my life. I can respect its power. There's something to deal with there. Who wouldn't want to integrate that kind of power with their life? I've always been into control, you know, and what better instrument than a computer? By teaming up with a computer and the ZGRASS language I have finally found a way to make being Jane Veeder a prime career slot."

She earns her living as a freelance graphics designer doing animated sequences for ads and spots, and as a programmer, writing programs for ZGRASS graphics packages. In computer jargon, a package is an environment, a collection of small programs that mediates between a user and the software language, in this case, ZGRASS. "My graphics packages stand between the user and ZGRASS the way ZGRASS stands between me and assembler code," she explained. "So I create environments for the ZGRASS user. I turn the computer and its controls into different things. You have menus and you choose stuff." The software modules for *Warpitout*, for example, were originally written as graphics packages for commercial applications.

"Being able to earn my living by sitting at home with my intelligent machine has made a huge difference," she said. "The sense of power and control over my life is immense . . . I want to evolve so far out there that I don't even know where I was before. I was reading recently some MIT pundit who was warning of the 'dangers' of using powerful interactive computer languages. One of the dangers he sees is that you get these feelings of powr. You start thinking that you're evolving into this sort of far-out electronic person. Well, the fact is that you *are*! And I mean what else do you want to do with your life?

A PRIMER Computer Animation

by Larry Cuba

Computer animation has been in development and use for years, but only recently has become a common feature of our visual world. We see it in movies, TV commercials and video games.

There are hundreds of computer animation techniques just as there are hundreds of ways to make music. To help understand these techniques, we can group the computer animation systems into categories, just as musical instruments are grouped into woodwinds, strings, percussion and brass.

In this article we will discuss seven generic systems as illustrated in Figure 1.

All the computer animation systems that we look at are composed of at least two component parts: the computer and a drawing device. The drawing device makes the images which are photographed onto film or videotaped. The computer controls the drawing process by sending the device electronic signals, and the artist controls the computer.

There are two kinds of computers: Analog and Digital. Usually when we use the word *computer*, we're talking about digital computers. These are the ones that send out utility bills and run video games.

Few people have even heard of analog computers, because when we encounter one, it's called something other than a computer. An electronic musical synthesizer, for example, is actually an analog computer.

Since the machine is not usually called a computer, calling the work produced with it *computer animation* has led to a confusion between analog and digital work. Rather than argue the semantics of the term 'computer animation,' we'll consider techniques based on both kinds of computers.

PART 1—ANALOG SYSTEMS
1. OSCILLOGRAPHICS

Analog computers were being used by artists to make films as early as the 1950s. Figure 2 is from a film by Hy Hirsch. The system used to create this film consists of an analog computer and an oscilloscope, and the technique is called oscillographics.

An oscilloscope is a screen and a beam of electrons which hits the screen and draws a point of light. The point of light can be controlled in three ways. The beam's position is

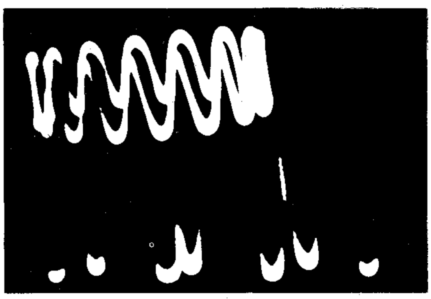


Figure 2

controlled by turning 2 knobs, one for Horizontal movement and one for Vertical movement. A third knob controls the intensity of the beam, and consequently the brightness of the point on the screen.

Instead of using knobs, we can attach wires to these functions and control them by changing the voltage in the wire. The amount of voltage flowing in the wire substitutes for the 'position' of the knob, and determines the position of the beam.

The position of the beam is *analogous* to the amount of voltage in the wire. That's why this is called an analog system. In digital systems, numbers are used to specify the position of the beam rather than voltage levels. The pattern of voltage channels in the wire is called an electronic signal.

So the problem of making drawings on an oscilloscope is a matter of generating electronic signals. This can be accomplished with an electronic music synthesizer. The classic music synthesizer (Moog, Buchla, Arp) is an analog computer.

The film or video artist creates animating images on the scope by generating signals with the music synthesizer and using them to control the movement of the beam. This process of composing an electronic signal by patching modules together with cables is used on all analog systems. The differences are in how that signal is used to make an image.

2. IMAGE PROCESSOR

By generating two independent signals and using them to control the vertical and horizontal movements, a pattern can be formed which com-

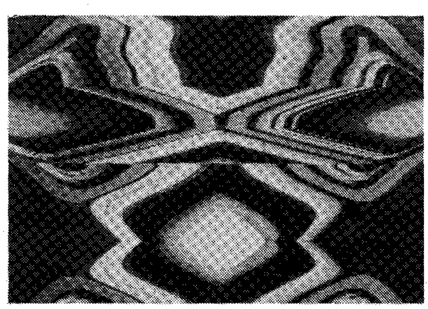


Figure 3

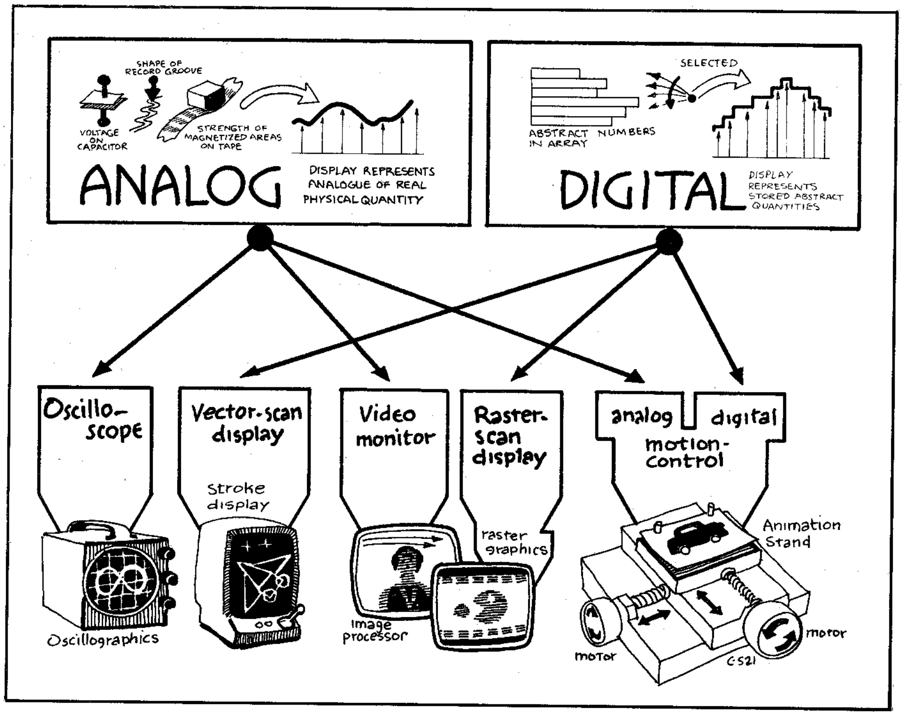


Figure 1

pletely fills the screen from top to bottom with horizontal lines. This is the scanning pattern used in video and the pattern is called Raster scan.

A television picture tube is similar to an oscilloscope except that the horizontal and vertical motion is fixed in such a way so that the beam is constantly scanning out a raster pattern. If the horizonal and vertical movements are fixed, then that leaves only the Intensity signal to form the image.

Variations in the beam's intensity during the scan produces an image composed of various shades of grey. The levels of grey in the image range from black to white and correspond to the range of possible intensity levels.

If we use signals generated with a synthesizer to control the intensity, then variations in the voltage level of the signal will result in variations in grey level. This is how a video synthesizer forms an image. The image in Figure 3 is from *Electronic Masks* by Barbara Sykes and was produced with the Sandin Image Processor (Figure 4).

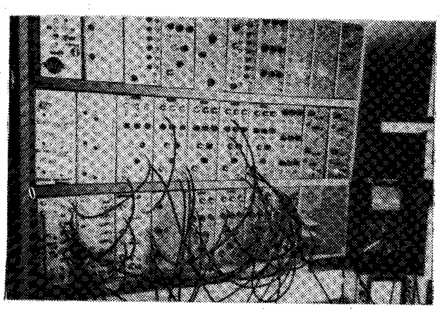


Figure 4

3. SCAN PROCESSOR

The next analog system we'll discuss is the scan processor. Video artist Woody Vasulka used a scan processor to produce the image in Figure 5. A scan processor, like the music synthesizer system, uses an oscilloscope-like display.

In a scan processor, the raster is scanned as in video, but unlike standard video, the pattern can be modified by the user. Any of the three signals forming the raster pattern (vertical movement, horizontal movement, and intensity) can be modified. These

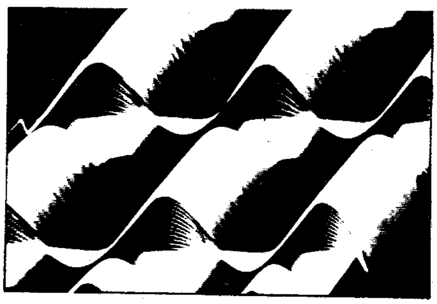


Figure 5

modifications come from either synthesized signals or a video camera.

In both of the video synthesizers we've looked at, image processors and scan processors, signals could be generated internally or with video cameras.

An image processor is designed to process the image as encoded in the standard video signal (that is, the intensity signal only). A scan processor, on the other hand, can actually alter the scanning pattern itself (that is, it can modify the horizontal and vertical movements of the beam).

If the scanning pattern is modified, then the result is no longer a standard video signal. (So, it can't be fed directly to a TV or tape recorded.) The results appear on an oscilloscope-like display and so scan processors belong with oscillographics on our chart (Figure 1). And just like oscillographics, to be recorded the screen must be filmed or rescanned with a video camera.

If rescanned with a video camera, then a video signal is created which can be processed with an image processor. So you could produce an image on a scan processor and process it and colorize it on an image processor.

These two functions are combined in a single system called the *Scanimate*, which was developed by Lee Harrison in the late 60's. Ron Hays used a Scanimate system to create his videotape *Canon* (Figure 6).

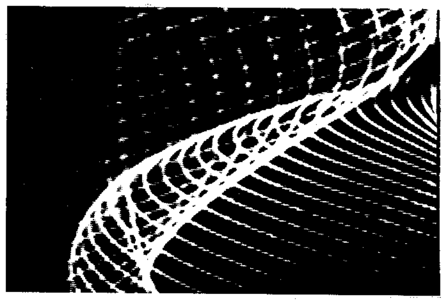


Figure 6

REAL TIME VS. CONVENTIONAL ANIMATION

In the systems covered so far, we've seen analog computers used to generate and process electronic signals. These signals were used to drive two types of displays: oscilloscopes and video monitors. With each of these two displays we could see the images moving on the screen in front of us. This display could then be filmed or videotaped like a live action scene. Should we call this animation, though?

In any film, what we see as a moving image is really a series of stills projected in sequence at great speed: 24 per second. In live action, these stills were created by the camera capturing a series of snapshots of the constantly changing real world at the same speed as projection: 24 per second.

In conventional animation, each of these stills is a photograph of a drawing. The drawings are placed on an animation stand and photographed, one at a time. When projected back at 24 frames a second, movement is synthesized. Film is the mechanism which allows us to project at great speed drawings which were created and photographed at slow speed (or one at a time).

In contrast to conventional animation, the systems we've looked at so far create and display the sequence of images at high speed so movement is visible directly on the display.

This is called Real Time Animation. The sequence of images is being produced and displayed at a speed which produces the illusion of movement. This eliminates the need for film as the mechanism of animation. Once a sequence of images is visible on the screen, then either film or video can be used to capture it at live action speed.

VIDEO ANIMATION

In conventional frame-at-a-time animation, the recording medium must be capable of recording in a still-frame mode.

This is easy for a film camera, hard for a videotape recorder. So animation has traditionally been made on film. More recently, single frame videotape recording has been developed, making "video animation" possible.

But this "video animation" has been confused with work produced on a video synthesizer, which has also been called "video animation." In one case, you have video record single frames, so you can shoot drawings with it, and in the other, you use video technology to synthesize animating images in real time.

4. ANALOG MOTION CONTROL

The movement in conventional animation comes from two sources:

1) changes in the image from one drawing to the next, and

2) slight movements of the stand between exposures to animate pans and zooms. These stand movements can be automated and controlled with computers, and since these movements alone can create animation, the animation stand can be considered a drawing device. This technique is called Motion Control.

Analog computers, because of their speed, have an advantage over digital computers when it comes to generating real time animation.

But for motion control, which is frame-at-a-time animation, you're better off with digital computers because of their many other advantages. And today's motion control systems are all digital. Still, the first motion controlled stand used an analog computer. It was called The Whitney Cam Machine. Motion Control techniques opened the door to a whole new world of special effects magic in Hollywood. When the movie *Star Wars* came out in 1977, it started a science fiction craze in Hollywood with dazzling effects like the jump into "hyperspace."

This jump into "hyperspace" effect is called a streak, and was invented by John Whitney, Sr., who invented motion control in the 50's by building the first computer-controlled animation stand in his garage with an army surplus analog computer.

The analog computer controls the movement of the artwork which is back lit on a light table. The film camera looks down from above and photographs the artwork as it moves. The principles of the Whitney Cam Machine were copied in machines that Doug Trumball built for the *2001* stargate corridor sequence, the ones Bob Abel built to make TV commercials, and the one John Dykstra built for *Star Wars*. Since *Star Wars* , motion control has blossomed into an industry standard, but now all systems are controlled with digital computers. We'll see them shortly in Part 2 on Digital Computers.

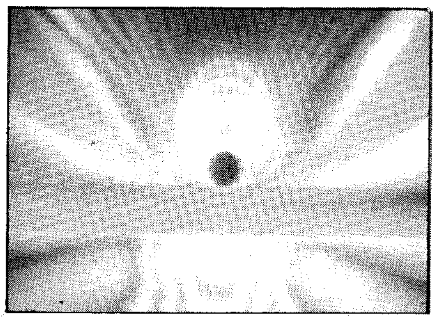


Figure 8

WHITNEY CAM MACHINE

The control mechanisms have advanced and become quite elaborate since motion control was invented, but the basic vocabulary of effects was established on the Whitney Cam Machine:

1) multiple exposures,
2) streaks, and
3) slitscan.

One benefit of motion control is the precise repeatability of animation stand movements. This ability can be used to produce multiple exposures. If, while the shutter is open, an object that emits light is moved into several positions and its light turned on, then multiple images of the object will appear on the film.

This principle is applied to animation in motion control: on an animation stand, artwork can be lit from behind. When the artwork is moved around in a circle and the lights flashed several times during the exposure, a multiple image is created. This is the technique used by James Whitney (John's brother) to produce the film *Lapis*. (Figure 7)

Another way to use motion control is to leave the lights on while the artwork is moving. This creates a time exposure effect like a photograph of car headlights whose movements are visible as a streak of light. A film, *Terminal Self* was consisting entirely of streaks were made by John's son, John Whitney, Jr.

This streak effect has often been mistakenly called slitscan, but streak and slitscan are two totally different effects. The only thing they have in common is that they are both

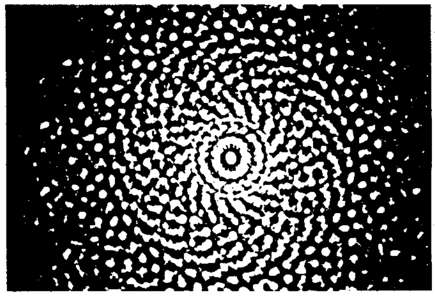


Figure 7

time exposures. Slitscan is a 3rd technique developed on the Whitney Cam Machine. In slitscan, the artwork is not seen by the camera lens all at once. The artwork is completely covered except for a sliver revealed through a narrow vertical slit in the covering. While the camera shutter is open for a time exposure the slit moves across the artwork and reveals it progressively to the camera which records an image on film.

If, during the exposure, there is no movement of the artwork or the camera but only the slit, then the art appears on the film as it would without the scanning slit, i.e. normal. However, if while the slit is scanning across the artwork the camera zooms in toward it, then a distortion is introduced: a bulge is produced in the figure. For example, a rectangle is transformed into a trapezoid.

These three elements of the basic vocabulary of the Whitney Cam Machine can be found in the later (and current) crop of motion controlled animation stands that use digital computers.

PART 2—DIGITAL SYSTEMS

For each of the 4 types of analog animation systems we have covered, there was a different KIND of analog computer. One was a Scan processor, another was an Image processor, and so on.

We'll look at several digital animation systems, but only one kind of digital computer. That's because all digital computers do the same thing fundamentally: they follow lists of instruction and can store these lists (and other data) internally.

The major variations among digital computers are:

1) the speed at which they operate,
2) the size of their internal storage.

These are usually a function of the size, cost, and age of the computer.

What distinguishes the digital animation systems on our chart in Figure 1 is the type of display. We'll consider three different kinds (vector, raster, and animation stands), but they're not mutually exclusive; a single system can include more than one type of display.

ANALOG VS. DIGITAL

The usual method of distinguishing between analog and digital computers is by comparing the way they process signals internally: analog computers use a gradual scale, whereas digital computers use discrete on-off elements. But these characteristics are at a microscopic level

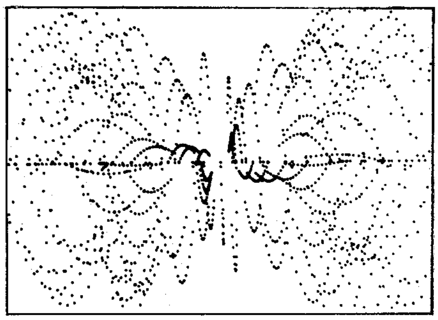


Figure 9

and do not directly affect the user. A more useful distinction is at the level of user control. The way in which an instrument's control is structured will determine what will be easy to do and what will be hard.

For example, imagine trying to drive a car by typing commands on a keyboard, or trying to write by guiding a pencil with a steering wheel. Clearly, then, the control structure of a tool can determine its usefulness to a large degree.

If we compare the control structures of analog and digital computers we find . . .

Analog computers are controlled by routing signals through modules with cables, and by turning dials. Digital computers can be controlled in a number of ways, which may include turning dials of drawing with a pen, but primarily they're controlled by typing in commands at a terminal. A list of these commands is a computer program. A program is just a description of the steps required to complete a task. In the past when someone might have built a mchine to perform a particular task, now one need only describe the process in a program. A digital computer is a robot which will exhibit the behavior specified by the program and so become the machine desired by the user.

This is the most important feature of digital computers, and apparently the least understood by the computer illiterate. For example, when a TV journalist reports a story with the sentence, "Some students at MIT have developed a computer that can play chess," they invariably mean, not that they've built a computer but

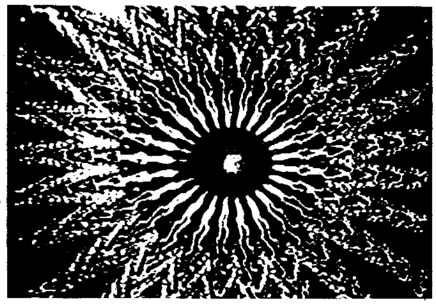


Figure 10

that they've developed a program. We use the word 'computer' to mean a general purpose programmable machine. As soon as it's designed for a specific purpose, then it's no longer a computer. Also, once a program is stored in a digital computer, you can use it at any time in the future.

With analog systems, the way in which the modules are connected is the 'program,' and there's no way to store it. If you take it down so someone else can use the machine, you'll have to put it back together again next time you want to use it.

And there is another difference between analog and digital computers: in the analog systems, we saw that values, such as how far the beam is to be moved on the screen or how bright it is, were encoded in the amount of voltage in the wires. On digital computers, when using a vocabulary of commands to control the display, these values are conveyed with numbers. Inside the computer, electronic signals encode these commands and numbers, but unlike the situation in analog systems, the user does not have to deal with them as much.

5. DIGITAL MOTION CONTROL

The Analog Cam Machine movements are controlled by an electronic signal. There is a correspondence between amount of voltage in signal and position of artwork being moved.

In the digital system, the animation stand is structured the same, but the user doesn't control its movements with signals per se. Instead, commands are sent via the computer with numbers to specify the desired positions of the artwork and the camera. For example, for a simple streak the program might look like this:

```
Open shutter
Zoom IN 50 units
Close shutter
```

This program puts one streak on one frame of film. For a stationary image to appear on film, this streak must go on every frame. So what we want to happen is for the computer to repeat this sequence of commands once for each frame. A 5 second effect lasts 120 frames, so we add a Repeat command, causing this group of commands to be repeated 120 times:

```
Repeat 120: [Open shutter
  Zoom IN 50 units
  Close shutter
  Zoom OUT 50 units]
```

After the frame is exposed, the command, Zoom OUT 50 units is used to reposition the camera to begin the

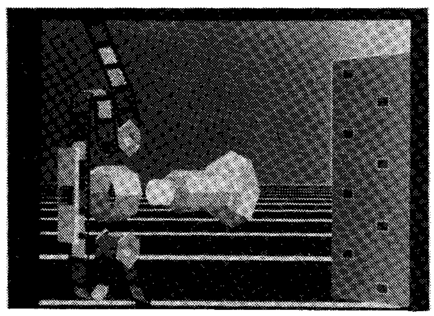


Figure 11

next frame. (There is also an assumption here that opening and closing the shutter advances the film one frame). This puts the same streak on each frame, producing a static image. For a moving streak, we introduce a slight change between frames, such as a small vertical movement of the artwork. This we do with the command, "Move UP 2 units":

```
Repeat 120: [Open shutter
  Zoom IN 50 units
  Close shutter
  Zoom OUT 50 units
  Move UP 2 units]
```

So as we zoom in and out to streak each frame, the artwork slowly moves up on the stand and the streak moves up in the frame. Multiple exposures and slitscan are also produced on digital motion control stands. Figure 8 is a slitscan image from Ron Hays' videotape *Tristan and Isolde*.

6. VECTOR GRAPHICS

If we look at our chart in Figure 1 again, we see two types of electronic displays: Raster scan and Vector scan. The difference between the two is the way in which the beam scans the screen to create an image (like the difference between an oscilloscope and a video monitor). Raster scan, as we learned before with analog systems, is the scanning pattern that video monitors use. The whole screen is scanned, one line at a time, from top to bottom. The image is formed by varying the intensity of the beam during the scan.

A vector scan display, on the other hand, is like an oscilloscope, in that the image consists of lines (called vectors) which are drawn by the beam as it moves around the screen.

But again, in digital displays the user doesn't create the signals that control the beam. These signals are created internally by the display in response to commands. The commands instruct the display where to draw points and lines by means of numerical coordinates.

There are various kinds of vector scan displays and they can be grouped into three classes based on three ways of recording an animation sequence: film recorders, storage tubes, and real time displays.

In a typical film camera, like a Bolex, we can identify three ways to shoot animation: single frame with a time exposure, short exposures (still single framing) and live action speed. In single frame mode, we can shoot time exposures by holding the release down. This holds the shutter open. The second way is also single framing, but pushing the release gives us a short exposure (1/30th sec.) no matter how long we hold down the release. And the third way, live action speed, is 24 frames/sec.

1) time exposure—A film recorder is a vector-scan display with a camera mounted facing it so a lengthy drawing process can be captured on each frame of film as a time exposure (not unlike the streaking process of motion control).

Since the drawing accumulates on the film over time, it can't be seen in its entirety by the user until the film is developed. This isn't very convenient, so these displays are usually used in conjunction with another type of display which can be used for previewing the image. This type of display is called a storage tube.

2) short exposure (but still frame-at-a-time)—the storage tube display has a screen which retains the image of the points and lines as they're being drawn instead of allowing them to fade, so that the user can see the whole drawing all at once and a camera can record the drawing on one frame of film with a short exposure (rather than a time exposure).

The process is still one-frame-at-a-time-animation, though. You can see individual drawings, but you can't see movement until the film is developed and projected. There are displays on which you can see movement directly without filming, and they're called real time displays.

3) live action speed—Real time displays draw and redraw images fast enough to see the animation without filming or taping (just like the process we saw in the analog systems). Figure 9 is from my film *3/78* which was animated on a real time system.

Since the vector images are animating in real time, we can point a video camera at the display and process the signal with an Image Processor. Figure 10 is from *Spiral 5* by Tom DeFanti and Dan Sandin and was produced on a system which combines real time vector graphics and image processing.

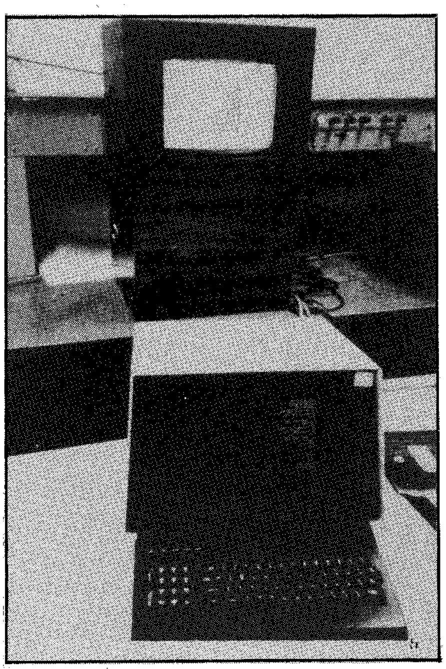
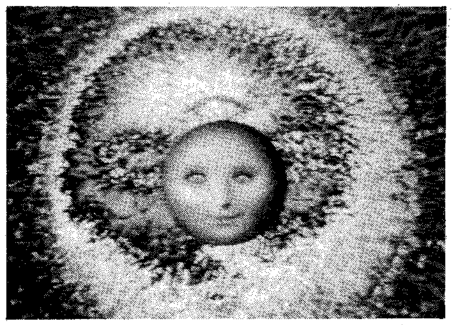


Figure 12

SOFTWARE VS. HARDWARE

Regardless of how the image is captured on film, the programming is a function of the main computer and the sofware available for it. The word 'Software' has two connotations. One contrasts Software with the word 'Hardware.' Hardware consists of all the pieces of the system which are material objects, such as the main computer, the display, the terminal, the printer, the disc drive and so on. Software, in this context, is the information that is stored in the various memory devices. Software includes programs which help you edit other programs, programs that draw pictures, or data such as the numbers in a phone book.


SUNSTONE

But in another context, like the phrase 'Software Development,' the word software does not refer to data, but to a class of programs which are designed specifically as tools. A program which helps you edit text, for instance, is a software tool. Software, in this sense of the word, can determine how useful a piece of hardware will be for a particular task. Software has a hierarchical nature which allows users to extend the tools provided, with their own. This is one of the more powerful aspects of digital computers and it doesn't exist in analog. This extendable nature of software makes it unique among man's tools in that the tool user can also be the tool builder.

7. RASTER GRAPHICS

Each of the three methods of recording animation (on film or tape) that we identified for vector graphics apply to raster graphics as well, namely:

1) time exposures—the same film recorder that's used for vector graphics is capable of operating in a raster mode, too. As in the vector mode, each frame in a sequence is stored on film via a time exposure. Figure 11 is from a commercial produced by Digital Effects using a film recorder.

2) short exposure (frame-at-a-time)—in vector graphics, a storage tube was used to preview images and to avoid the need for time exposures during filming. But there is no such comparable screen that can store a raster image. So the problem of disappearing images is solved by creating a special section of the computer's memory (called a frame buffer) for the storage of the data required to draw one frame. The display screen is constantly refreshed from the data in this frame buffer memory. So, as the main computer builds the image by changing the numbers stored in the frame buffer, the results are continuously visible on the screen.

The resolution of a frame buffer is measured in number of pixels. The word pixel is a contraction of Picture Element. The pixels fill the screen the way squares fill a page of graph paper. The finer the grid of your graph paper, the smaller is each pixel and the more you have. The more pixels, the higher the resolution.

For each pixel on the screen, there is a number stored in the frame buffer which determines the shade of grey (or color) for that pixel. Changing the number assigned to a pixel changes its shade (or color). The range of numbers is limited and determines how many colors are possible for any pixel. Since this number is stored in binary form, its size is measured in number of bits per pixel. But you don't need to know binary arithmetic to compute the resolution of a frame buffer—just this relationship: 1 bit means 2 colors, 2 bits means 4 colors, 3 bits, 8 colors and so on. The amount of memory devoted to each pixel (in number of bits) is a measure of the color resolution of the buffer. The complete resolution of the buffer, then, would combine spatial and color, and would be specified as number of pixels Wide by number of pixels High by number of bits per pixel, or Depth. For example, the resolution of the frame buffer in the Datamax Zgrass computer is 320 Wide x 2 bits/pixel. (Figure 12) High resolution frame buffers were used by Ed Emshwiller to produce his videotape, *Sunstone*, (Figure 13).

3) live action speed—if these buffers can be udpated fast enough, the animation can be seen in real time. From video games to flight simulators, real time animation is required when the images must respond to human interaction. Stan Vanderbeek used the flight simulator at NASA's Houston center to create the images of his film, *Euclidean Illusions*, in real time.

The difference in resolution, realism and visual quality between video games and flight simulators is obviously enormous. This reflects the large difference in size and cost of the computers. But as these factors continue to drop, who can say how long it will be before playing a *Star Wars* computer game will look like being IN *Star Wars*, the movie?

A videotape of this presentation will soon be available. For more information write: Larry Cuba, 1803 Mission Street No. 123, Santa Cruz, CA 95060.