# PROVIDING HIGH PERFORMANCE VISUAL SIMULATION AT LOW COST

Richard N. Moon
Evans & Sutherland Computer Corporation
580 Arapeen Drive
Salt Lake City, Utah 84108

## ABSTRACT

For years, the users of visual systems, in both the military and commercial worlds, have made a plea to the developers to significantly lower the costs of those visual systems. Then, in a second breath, they have continued to demand high visual fidelity. Not surprisingly, the trend among visual system developers has been to provide more capabilities for higher cost. This paper describes a system design where a serious and concerted effort has been made to lower costs, while still maintaining the features most essential and effective for training tasks. During the highly selective process of determining system features, some capabilities, such as smooth shading, color blending and transparencies, were seriously questioned. Other capabilities, such as texture, scene detail management, resolution, dynamic coordinate systems, and reasonable image quality, remain high on the list. The paper describes how the essential features were incorporated in a highly cost-effective design with surprising flexibility and modularity. Painstaking efforts were made to optimize the hardware efficiency, making use of custom, semicustom, and commercial VLSI. These latest technologies have made possible a parallel processor architecture in the geometric processor--a departure from the traditional pipeline approach. The architecture provides system capabilities ranging from low-cost, night-only operation to high-resolution, daylight scenes rivaling the current high-end systems. All of this can be contained in a single card cage per channel. The challenge of the visual users has been met.

## INTRODUCTION

Some time ago, Evans & Sutherland Computer Corporation and Rediffusion Simulation determined that the time was right to develop the next generation of NOVOVIEW visual systems. With the exciting possibilities now made possible by VLSI and other new technologies, in combination with the experience gained with the NOVOVIEW and CT[1] products, it was desirable to offer a new family of visual systems.

The user community has continually made the plea for lower cost visual systems. We have heard those pleas. We have also heard the equally loud demand for higher capabilities and fidelity. We realized from the beginning that current capabilities and features could not be reduced; in fact, we well knew that we would have to increase capabilities. However, we have made a conscious effort to keep the lid tightly screwed down on cost. In this development, it wasn't the goal to surprise the market with new advanced features, but rather provide a carefully selected set of well-proven features at a reduced cost. The resulting product line has become known as SPX.

## SELECTING THE CAPABILITIES

With this overall goal well established, it was no easy task to determine what the capabilities for the system should be. We started by enumerating all of the features of the current NOVOVIEW product line. These systems have provided very realistic night and dusk scenes for training in most of the commercial airlines as well as some military applications. The realistic night scenes were made possible by the use of calligraphically drawn light points. It was felt that the new system should include this capability. The highly specialized and calibrated visibility and environmental effects should also carry over. Other features were also evaluated relative to their effectiveness. Nearly all of them must be included in order to meet the FAA Phase II and Phase III requirements.

After considering the current features, the door was opened to suggestions for additional enhancements. The suggestions came pouring in. It seemed as though the list would never end. Some of the suggestions seemed beyond reach, almost frivolous; but many had great value.

Although all of the design goals cannot be enumerated in this short paper, a few of the most important ones should be mentioned. Texture was a must. Expanded capability for moving objects was high on the list. Increases in image quality and resolution seemed necessary. The system would need to be very modular and flexible in order to cover a wide variety of needs. With the increasing need for larger and more detailed models, the effective capacity of the system must be large. Here, it was known that very significant gains could be achieved by providing better data base management and scene culling early in the computation process.[2,3] Such techniques would not only increase the system capacity, but increase the effective display capacity by ensuring that each polygon and light processed and drawn is an effective part of the scene. And in considering large models, the process of creating these data bases would need to be streamlined so that the models remain affordable.

However strong the pressures and temptations were to get carried away in including everyone's wishes, one overriding requirement remained strong: keep the cost under control.

If selecting the requirements was difficult, implementing them within the cost constraints was even harder. Although the system requirements were established early in the development, their solidification was tied to the actual hardware and software design. Designs were optimized--several times. VLSI, both custom and commercial, was exploited. New packaging methods were examined and implemented where appropriate. Capabilities were re-examined. Software programs and tools were developed. Gradually the obstacles yielded; the results were rewarding.

The next sections of the paper describe the major parts of the image generator, their architecture and capabilities.

## REPLACING THE GENERAL PURPOSE COMPUTER

Traditionally, visual systems have included general purpose computers at the front-end of the image generators. These computers have ranged anywhere from a small minicomputer to a fairly large and powerful super mini. The general purpose computer contributes significantly to the cost of the system, and has become so ingrained in the visual system, that many commercial and military requirements documents spend considerable effort specifying the attributes of the computer.

As visual systems have been developed, the use of the front-end computer has undergone significant changes. Early systems actually performed major portions of the standard graphics algorithms in the computer. However, as system capacities have increased, more and more of the graphics algorithms have, of necessity, been committed to special purpose hardware. In many of the latest systems, very few, if any, of those algorithms are performed in software. In fact, many additional tasks, such as special effects algorithms, are now being performed by special purpose hardware.

With many of these tasks off-loaded to the special purpose hardware, the computer, during the actual simulation process, acts primarily as an interface device. It serves as the interface to the host computer, interfaces the disk to the environment data memory, provides overall system control and executes some special effects programs. For these few highly specialized tasks, the question can be asked: "Why require a general purpose computer?" Additionally, the super mini usually has a powerful (and slow) operating system, complete with extensive file management, multi-user software, and a whole set of unneeded capabilities which drive up the cost of the system and sit idle during the simulation task.

Most of the raw computation and interface capabilities can be contained in a handful of commercially available VLSI chips including microprocessors, DMA controllers, serial and parallel interfaces, and memory. The SPX system makes use of a single-card, front-end processor based around the Motorola 68000 microprocessor. The capabilities of this card are highly suited to host computer and mass media interface tasks, while providing adequate computation and system control functions. The card provides a standard SCSI interface to a variety of cost effective mass media devices such as Winchester disks and tape streamers. Ethernet has been selected as the host interface, providing sufficient speed at low cost. If more front-end capabilities are needed, additional Multiprocessor (MP) cards can be inserted to operate in a parallel processing mode.

In an off-line mode, the Multiprocessor card can support full data base development capabilities as well as system diagnostics and maintenance functions. Again, the cost effectiveness of the system is increased.

## EFFICIENCY THROUGH OBJECT MANAGEMENT

One of the major goals of the development is to provide for an increase in system capacity as well as an increase in the apparent polygon capacity of each channel. This would require a front-end processor which would provide for culling scene detail which was out of the field of view or too distant to be of value to the scene, and submitting to each channel only the data most likely to appear in that channel. Through early scene management, each Geometric Processor and Display Processor can spend its energy processing data base elements which will appear on the display rather than wasting time processing elements which will eventually be rejected anyway.

A small set of hardware, called the Object Manager (OM), was designed to perform five basic functions. First, it provides large area data base management by traversing the hierarchical data base structure and retrieving from disk those scene elements which are close enough to the eye point that they have the potential of appearing in one or more of the visual channels. Second, the OM sorts through the same structure and creates a list of the objects, in highest to lowest visual priority order. Third, it provides level-of-detail management so that objects can be displayed with less detail when they appear at far distances and increase in detail as they get nearer. Fourth, it provides for object instancing, where objects need be modeled only once in the data base, but can be repeatedly used throughout the scene. Finally, each potentially visible object is tested against each channel's field of view to see if the object is in a given channel.

Some portions of these functions deal with system issues which are independent of the individual channels. Other portions are channel dependent. We had the option of designing a system-general management section or placing individual object managers in each channel. The decision was made to place small management units in each channel. This configuration has several advantages. It allows the management function to expand as channels are added. It also gives significant flexibility to system configuration, offering offset eyepoints or multiple independent eyepoints.

## THE GEOMETRIC PROCESSOR

The basic functions to be performed in the Geometric Processor (GP) are fairly universally understood. This is the section of the image generator which performs vector translations, vector rotations, polygon clipping, and perspective division. The need exists for a set of hardware which can perform large numbers of these calculations in a very short time. Previous NOVOVIEW systems had performed many of these calculations using bit-serial arithmetic circuits arranged in a pipeline architecture. Such an architecture is low cost and also lends itself well to high speed VLSI implementation.

Another approach to performing the GP algorithms is to use parallel arithmetic circuits in a pipeline fashion. This approach offers greater throughput than the bit serial approach, but usually at commensurate cost.

The SPX design team analyzed the pipeline approaches, only to repeatedly find that they suffered from three basic problems. The first is that there are inefficiencies associated with processing a mixture of entities which require different amounts of time to compute. For example, light points can usually be processed more rapidly than polygons. However, if lights and polygons are intermixed, the lights can flow through a multistage pipeline no faster than the slowest polygon. This variance in computation between different elements can be smoothed by placing FIFO buffers between the various stages. Again, the cost goes up. The second major problem is that pipelines have difficulty being sufficiently flexible to perform a large number of varied algorithms. Third, pipeline architectures do not lend themselves to expansion. It is difficult, short of duplication, to increase their throughput.

Graphics for simulation requires much more than the traditional algorithms such as translations, rotations, clipping and perspective divisions. In the visual system, the hardware should process many special effects such as rotating and flashing lights, curved strings, directional illumination characteristics, VASI systems and a variety of other special cases. Special texture functions are also required, and visibility effects add to the tasks.

In order to provide the flexibility and modularity desired, the SPX engineers decided in favor of an alternate to the pipeline approach. It employs a number of independent processor sets, each capable of performing all of the required algorithms. Figure 1 illustrates how SPX uses multiple copies of an arithmetic processor card which contains high speed arithmetic and logic elements connected to a data memory. The entire structure is 32 bits wide, providing sufficient accuracy to eliminate most floating point operations. A writeable control store memory allows the on-card control to hold the algorithmic code for all geometric and special purpose functions. Several cards operate independently, but in parallel, to provide sufficient throughput. Additional cards may be added as system needs require. This non-pipeline approach provides efficient use of the computation elements, almost limitless algorithm flexibility, high throughput, and expansibility.

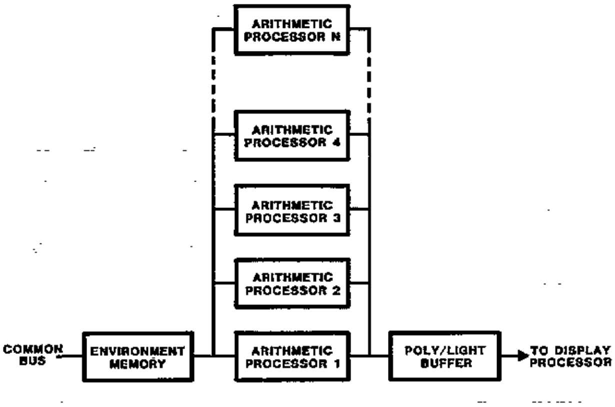


Figure 1. Geometric Processor Block Diagram

## THE DISPLAY PROCESSOR

The real struggle in developing a cost effective visual system came in the Display Processor area. The Display Processor is that portion of the image generator which follows the Geometric Processor and actually forms the pixel elements to be drawn by the display device. Many architectural approaches were examined, and the one finally chosen uses an area processing algorithm. This approach produces superior quality images and polygon counts which increase nearly exponentially with processing time. It was first developed by Evans & Sutherland[4] and forms the basis of the CT5 and CT6 systems. This approach has subsequently been used by other visual system manufacturers.

When the area processing architecture was first considered for SPX, its size was almost overwhelming, since the minimum amount of hardware which must be in place to make the approach work seemed very large. The real challenge was to define the smallest complement of hardware which could process the entire screen area several times each field with sufficient resolution to give excellent image quality. In order to reduce the cost of the hardware involved, each capability and algorithm of the Display processor was carefully examined. In containing costs, it would be necessary to limit capabilities to those felt to be the most essential.

As the development progressed and each area was examined, it was discovered that some advanced features required surprisingly little hardware. A good example was the transparency feature. With the area processing architecture, transparencies come with the territory. A similar situation arose with variable amounts of antialiasing capability. It was supposed that a savings could be realized by reducing the amount of antialiasing, particularly in the low-end night/dusk configuration. However, it was found that the capability would cost more to take out than to leave in. Thus, all configurations of SPX will provide image quality identical to that of CT6.

On the other hand, the amount of hardware required to perform the interpolations necessary for smooth shading, color blending and the traditional distance related fog was significant. After careful consideration of the actual use of smooth shading in many applications from the CT5 Visual System, it was decided that smooth shading could be eliminated, and the system would still maintain the capability to offer effective training in almost all areas. Fog was a more difficult problem. It can't just be eliminated! The process of determining the distance to each vertex of a polygon and then interpolating across the interior of the polygon was expensive, especially when calculating several pixels in parallel. One alternative was to implement an algorithm which would scan the fog correctly for the ground plane, and then fog 3-D objects on an object by object basis. All polygons of a given object would receive the same fog value. This approach would probably be better than prior systems of this class, although it wouldn't give very good results on large objects like mountains. During the struggle to define the best fog algorithm, a new approach was discovered (or rediscovered) which paralleled many of the computations necessary for texture. This approach, which has been implemented in SPX, provides distance related fog for all polygons and lights in the scene and varies smoothly across polygon interiors. Its beauty lies in the fact that much of the hardware necessary for its computation provides simultaneous input to the texture section.

Since texture has already been mentioned, it is appropriate to discuss it here. When texture was first implemented at Evans and Sutherland in the SP3T system, its computation was simplified due to the use of rolled raster. It also, of necessity, was restricted to the ground plane or planes lying parallel to the ground plane. Since SPX would no longer use rolled raster, the more complex algorithm would need to be implemented. Once the algorithm was expanded to handle the case of unrolled raster, it was a straightforward addition to provide texture on polygons of any orientation.

SP3T provided four texture patterns with many different combinations of all four patterns possible on each polygon. The most useful combinations of patterns involved combining only two patterns at a time. The database designers also expressed a need for more patterns. Thus in SPX, more patterns will be available; but, in the interest of reducing hardware cost, each polygon can specify only the more effective two-pattern combinations.

Another factor which needed careful consideration was the number of polygons and lights which should be processed by the visual system in order to provide effective training. With the area processing architecture, the number of polygons which can be drawn increases rapidly with small gains in processing power or time.

The temptation is to go to the extreme in polygon counts. However, polygon and light counts impact the geometric processor in a nearly linear relationship. Several of the low cost visual systems have provided very effective training with 200 to 450 polygon capacities. Ofttimes these numbers referred to the total polygons being submitted to the image processor, while many fewer were actually drawn. With good data base management, level of detail management, and scene element culling based on distance and individual channel fields of view, the effective polygon capacity can increase many fold without actually increasing GP and DP capacities. In addition, the advent of texture significantly reduces the need for large numbers of polygons to define the ground scene. With these factors in mind, the polygon capacity of SPX was set at about 500 displayed polygons in each channel. This polygon count is significantly less than that of the top-end visual systems, but probably five to ten times higher than the effective count of systems of this class. Remember, SPX was designed to provide effective training at much lower cost.

Point lights also play a large role in training simulation. Calligraphically drawn lights have always given superior performance than raster drawn lights. For effective nighttime and low visibility take off and landing training, the small, high contrast calligraphic lights are preferred. The new architecture provides for the option of calligraphic lights in addition to raster lights. If the calligraphic option is removed, all lights will be integrated into the raster structure. The display drivers are capable of driving either calligraphic displays or raster scan displays.

A further reduction in Display Processor hardware came from the incorporation of custom and semicustom VLSI. A conservative estimate revealed that the custom and semicustom VLSI in the Display Processor reduced the size of the design by 35 percent. This, of course, does not count the use of commercial VLSI.

## SYSTEM FLEXIBILITIES

The SPX system is much more than a single system; it is a family of visual systems ranging from a quality low resolution, monochrome system, to a powerful high resolution system which will rival even the high-end systems. Figure 2 shows a block diagram of the basic structure of the SPX system. It shows the channel expansion capability, where each channel processor contains an optional Object Manager, a Geometric Processor, and a Display Processor. Each system can be expanded to include up to eight channels. Within certain constraints, individual channel processors belonging to a system may be configured differently. Notice that multiple Multiprocessor cards can be added to the system to accommodate special user needs. Multiple disks, and other mass storage devices are also possible.

Each Channel Processor is housed in a single backpanel which is prewired to accept all hardware options. Thus, systems originally configured as low-end systems can easily be expanded by simply

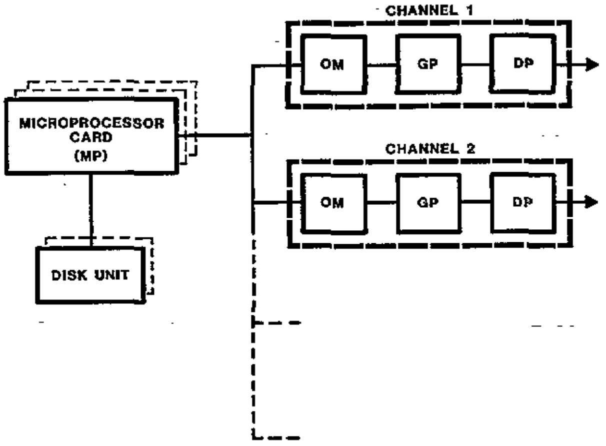


Figure 2. SPX System Block Diagram

plugging in expansion options. System packaging is also compact. A fully expanded three-channel system is packaged in only two 19-inch cabinets, excluding terminals and displays.

The hardware Object Manager itself is an option. The standard low-end system will not be equipped with an OM, but will rely on the Multiprocessor card to perform limited data base management. A hardware OM can be added to provide additional capabilities. All larger systems require a hardware Object Manager.

Geometric Processor capacity can be expanded simply be adding more Arithmetic Processor cards. As cards are added, the system automatically expands to use the additional cards; neither the software or any other part of the hardware needs configuration changes to accommodate the new cards. On the other hand, the system can operate with degraded performance in the event an Arithmetic Processor card fails. In addition to the flexibility of adding more cards, the programmable nature of the card offers future flexibility.

The Display Processor hardware is modular in the following six areas.

1. Resolution is modular in three basic steps, ranging from about one quarter to three quarters of a million pixels at 50Hz operation.

2. Surface rendering can be either monochrome or full color.

3. Surface texture is an option which can be included in both the low-end and high-end configurations.

4. Landing light lobes can be added or removed.

5. Calligraphic light capability is a plug-in option (provided the displays are compatible).

6. The image generator can be interfaced to a number of different types of displays and projectors, including calligraphic and raster scan displays.

Figure 3 is a block diagram of a minimum configuration Display Processor. A channel of this configuration could provide monochrome scene simulation such as might be required for sensor simulation.

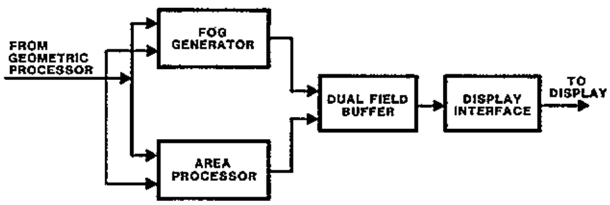


Figure 3. Minimum Configuration Display Processor

Figure 4 shows a maximally configured system, complete with full color, texture, calligraphic lights, landing light lobes and high capacity and resolution. Many possible configurations lie between the minimum and maximum configurations.

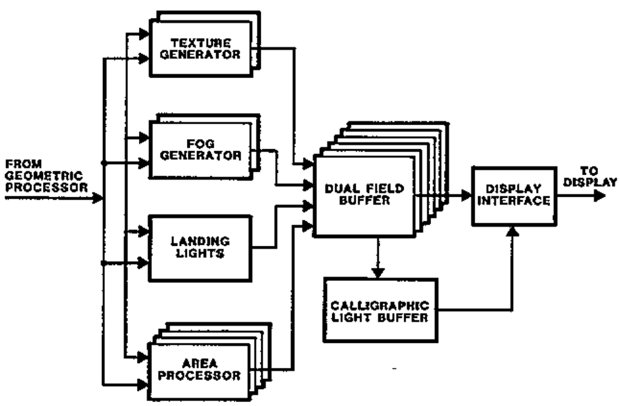


Figure 4. Maximum Configuration Display Processor

In addition to flexibility provided by hardware configuration, many system parameters such as raster scan standards, gamma correction functions, etc. are stored in RAM and can be modified to suit a particular user.

## CONCLUSION

The SPX family of visual products is the result of a conscious effort to provide an effective visual simulation training device at the lowest price possible. Great efforts have been made to simplify the architecture and then implement it efficiently through state of the art technology. The photographs in Figure 5 give some indication of the image quality and fidelity attainable with SPX. Its modular architecture will provide the flexibility to meet the majority of visual system needs. Further, since the SPX architecture is similar to that of CT5 and CT6, additional growth can come from future exchanges.

## REFERENCES


1. Cosman, M. and R. Schumacker, "System Strategies to Optimize CIG Image Content." Proceedings of the IMAGE II Conference, June 1981.

2. Mayer, N. and M. Cosman, "Enhancing the Computer Generated Illusion." Proceedings from the 4th IITEC Conference, November 1982.

3. Wales, Cary E. and Cosman, Michael A., "DMA and CIG: A Shotgun Wedding." Proceedings from 5th IITEC Conference, November 1983.

4. Schumacker, Robert A., "A New Visual System Architecture." Proceedings from the 2nd IITEC Conference, November 1980.


## ABOUT THE AUTHOR


Mr. Richard N. Moon has been with Evans & Sutherland for the past 11 years. He has been actively involved in the development of both hardware and software for the NOVOVIEW visual systems. He currently manages the SPX development effort. Mr. Moon received his MS and BS in Electrical Engineering from Brigham Young University at Provo, Utah.

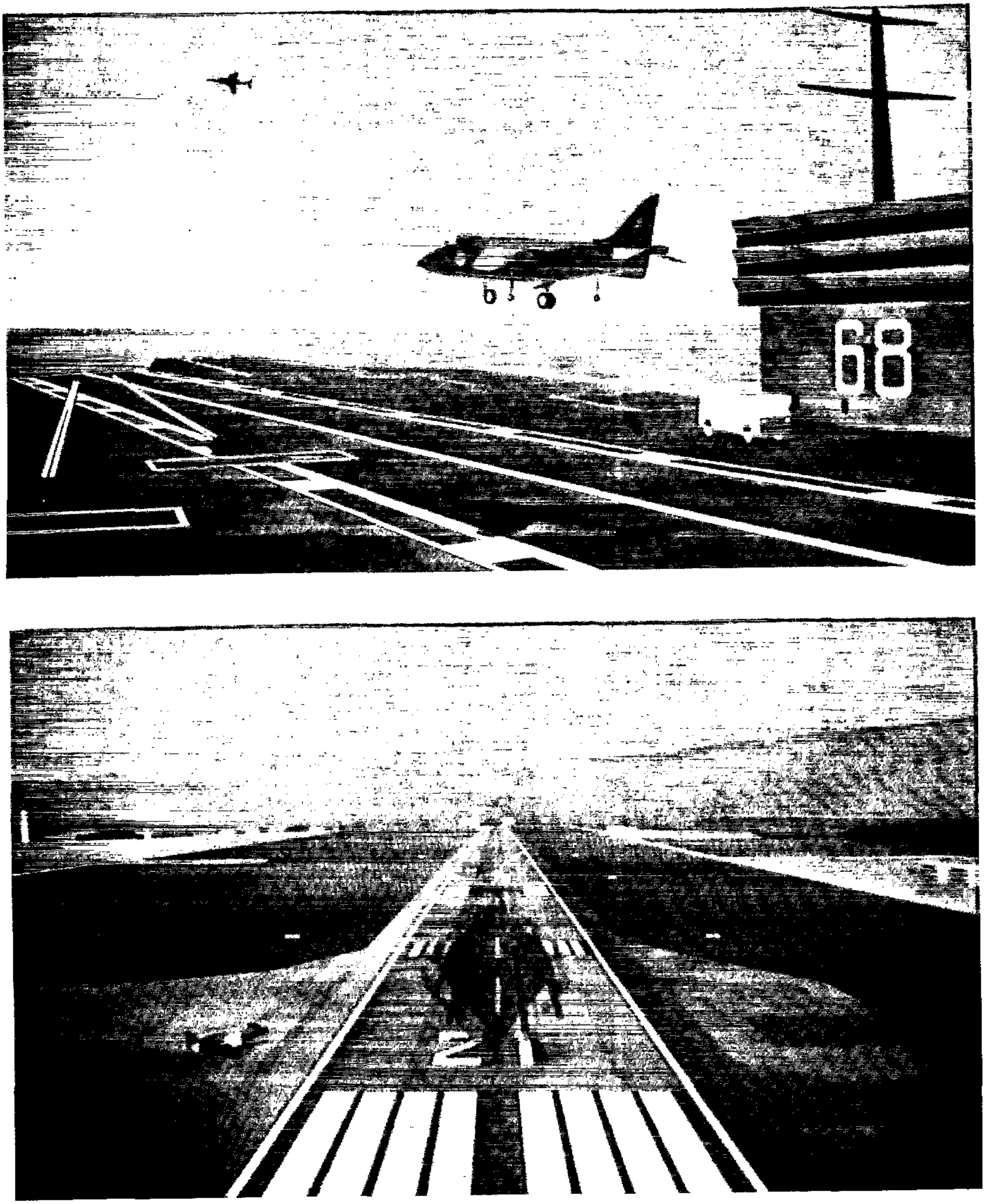


Figure 5. Scenes Representative of Those Produced by SPX