Computer Science Technical Reports

Computer Science

01 Mar 1988

Intensity Blending of Computer Image Generation-Based Displays

Elizabeth Scheppler Reidelberger

Daniel C. St. Clair
Missouri University of Science and Technology

INTENSITY BLENDING OF COMPUTER
IMAGE GENERATION-BASED DISPLAYS

E. S. Reidelberger* and D. C. St. Clair

CSc-88-1

Department of Computer Science

University of Missouri-Rolla

Rolla, Missouri   65401   (314)341-4491

ABSTRACT

State-of-the-art combat simulators require a 360 degree field of view, allowing the pilot and radar intercept officer to have the same visibility in the simulator that they would experience in the aircraft. The sky/earth display must be computer-generated and displayed with a minimum of two channels to provide the most realistic display possible. The two channels of display come together in the dome, forming an equator, that must be as indiscernible to the aircrew as possible. To accomplish this, an algorithm has been developed for controlling the video output which makes the two separate channel displays appear as one continuous 360 degree display.

PREFACE

The computerized blending algorithm discussed in this
thesis was developed to provide a means to join two large
field-of-view displays in an indiscernible seam. The
blending algorithm is a flexible, computerized model used
to control the video intensity of displays when combining
two displays together to result in the appearance of one
continuous display. This research should interest users
and designers of visual systems used in air combat and
other types of flight simulators.

The first two sections of this thesis, Introduction
and Computer Generated Imagery, provide introductory
information. The section entitled Hemispherical Display
Continuity considers the nature of a hemispherical
display and the effect of the projection system. This
section also deals with different methods of joining
displays, and the advantages associated with each.

The next section, Blending Region Definition, expands
on the discussion in the previous section by defining the
overlap area in which blending will occur.

The section on Standardized Space covers an integral part of the blending algorithm, the definition of standard space. This will simplify the blending algorithm significantly, without loss of accuracy.

The next two sections discuss how the blending algorithm is implemented and address the raster filling process, the final step in blending. Finally, the results and conclusions are presented in the final section.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

INTENSITY BLENDING OF
COMPUTER IMAGE GENERATION-BASED DISPLAYS

E.A. Reidelberger
McDonnell Aircraft Company
St. Louis, Mo. 63166

Daniel C. St. Clair
University of Missouri-Rolla
Graduate Engineering Center
in St. Louis
St. Louis, Missouri 63121

INTRODUCTION

Air combat simulators provide training to pilots and radar intercept operators (RIOs) in fighter aircraft. Of critical importance is providing the most realistic simulation possible when training crews in air combat maneuvering. Air combat simulators should be thought of as training devices that can complement "real world" training [1]. Critical skills can be developed in simulators without risk of life or equipment, and for a fraction of the cost of actual flight time. Training in this manner can make actual flight time more valuable and more productive [2]. To accomplish the above goals, the simulator must provide the best possible training for the aircrew.

Modern air combat simulators take advantage of state-of-the-art designs and techniques to provide the most realistic environment possible for pilot training. An air combat simulator consists of a cockpit enclosed by a projection surface on which a visual scene is shown. As the pilot operates the aircraft controls, the visual scene responds.

Providing a highly realistic and dynamic background visual scene is difficult. It is imperative that the proper cues be provided which will elicit the correct response from the trainee [3]. The combined effects of high speed and low altitude make it a technical challenge to change the visual scene and apparent motion of the simulator realistically in response to the pilot's actions. A high-quality computer-image generation (CIG) system produces superb quality real-world images which provide the flexibility to enhance training with realistic sky/earth background scenes. The result is an accurate, meaningful visual training environment that can meet the complex military needs for simulated visual scenes in simulators.

COMPUTER GENERATED IMAGERY

Computer image generation systems use a digital
object/terrain data base to compute geometrical images of
real world scenes. The system consists of equipment to
generate, in a visual scene relative to a current
viewpoint, a variety of terrain and person-made cultural
features [4]. Imagery is generated in video format,
usually a 1023 line raster which can be projected on a
screen by special purpose light valve projectors. The
images are viewed from a design eye with six degrees of
freedom. The images are updated rapidly enough to
accurately simulate translational and rotational motion
cues. Potentially visible scenery is extracted by
processing the data base, an organized hierarchy of
models, objects and faces against the dynamic data
defining the position and attitude of the viewpoint. The
environment, containing moving models and terrain
environment models, is scanned to select the models that
are viewable within the display channel boundaries. The
models are projected into the viewplane and transformed to
2-D projections as observed from the viewpoint. The 2-D
information is filtered to eliminate that portion which is
imperceptible in size or is hidden from the viewpoint.

Wide Angle Visual Displays. The goal of realism in
visual displays has led to the use of wide angle displays
which are projected onto spherical screens, or domes.
Domes are used to overcome the 180 degree limitation of
flat screens by allowing several projectors to project
imagery onto different areas [5]. A dome provides full
360 degree visual coverage. It offers wrap-around
background visual imagery of earth and sky scenes which
provide distinct, unambiguous altitude, attitude and
velocity cues. This is particularly critical when dealing
with flight simulators for training pilots to carry out
low-level flying, ground attack, air-to-air combat and
other missions requiring critical maneuvering and visual
tracking [6]. The light valve projectors with specially
developed optics packages provide wide angle projection
capability. This allows each projector to be able to
place an image on an area covering greater than a 180
degree field of view, requiring only two light valve
projectors for full 360 degree coverage. This provides a
highly realistic display with an unrestricted field of
view for both occupants of the cockpit. A typical domed
air combat simulator is shown in Figure 1. Both pilot and
RIO have unrestricted vision; anything that would be
visible to the aircrew in the real aircraft with the
canopy closed and head motion limited to that experienced
by the aircrew wearing helmets and unlocked shoulder
harnesses would be visible to the aircrew in the dome.
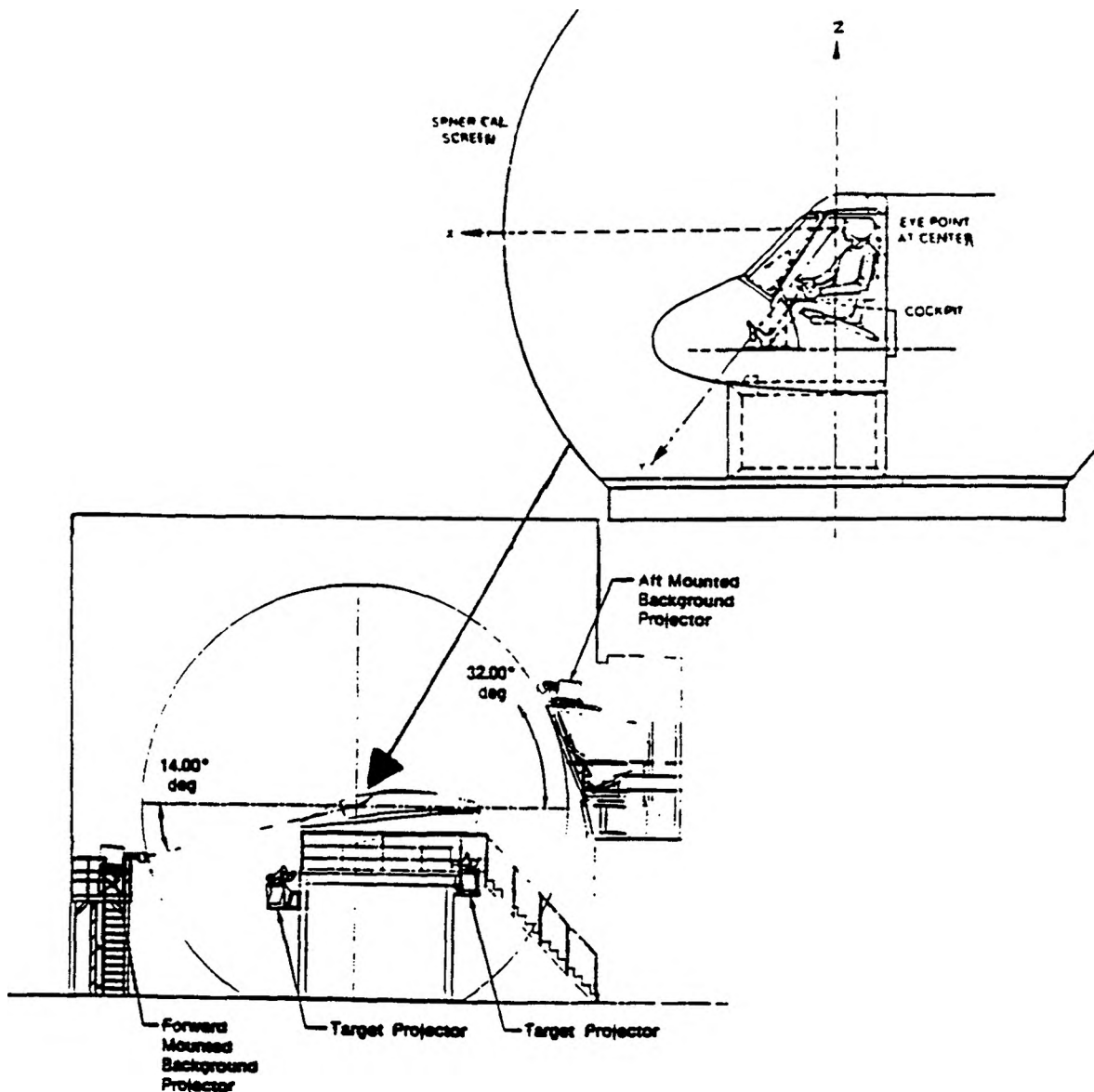
Figure 1. Typical Domed Aircraft Simulator [7,8]

The cockpit itself restricts some vision due to its structure. For example, the aircrew is unable to see below the aircraft because of the airplane structure. The areas outside of the cockpit can be used to their best advantage by positioning equipment such as projectors which will be unseen by the aircrew, but can display

images on the dome surface. However, there is no location inside the dome for a projector which will be out of the aircrew's field of view but can provide a full hemispherical display without an obstruction. Therefore, when projecting a hemispherical display, the projector must be located outside the dome, projecting through a small hole in the dome which, if positioned correctly, is invisible to the pilot and RIO. This allows each light valve to project a hemispherical display, providing a full 360 degree display coverage in the dome. Using only two projectors, some areas in the aircrew's field of view will be shadowed by other trainer equipment, such as target projectors or the cockpit itself, or will not be covered by images projected from either background projector, as shown in Figure 2. If the areas not covered by the display but within the aircrew's field of view are minimized by optimizing the location of the background projectors, they will not adversely affect the quality of the display. Figure 1 represents an example of optimization of projector locations given the constraints of the equipment located in the dome which results in the visibility plot shown in Figure 2.
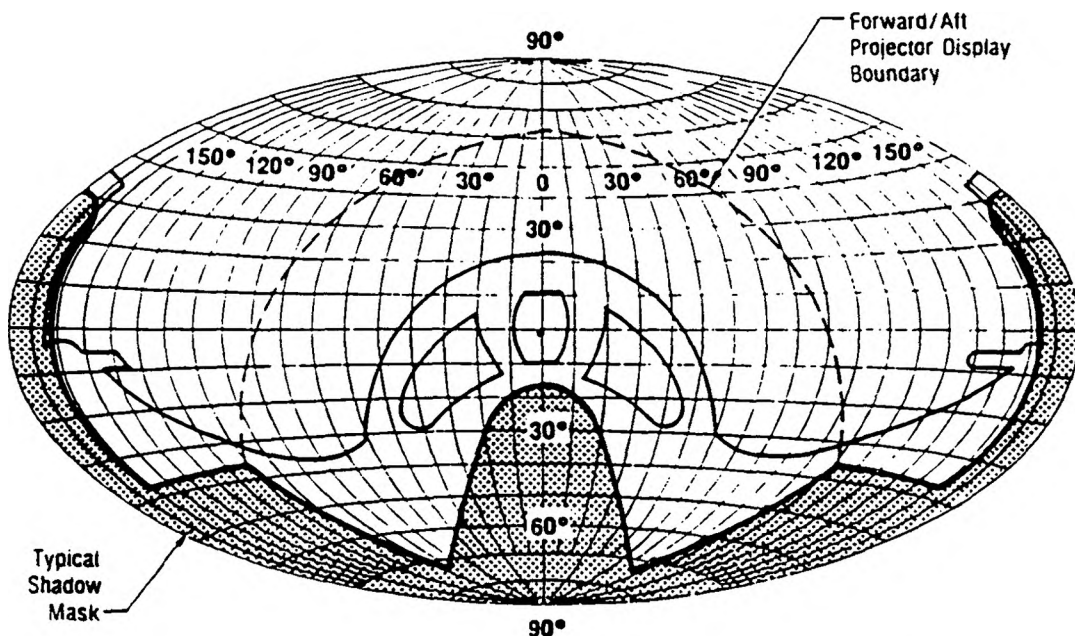
Figure 2. Display Coverage Plot [7]

Display Distortion. Ideally, the exit pupil of the
projector and the viewpoint should be located at the
center of the dome to eliminate distortions. However, the
viewpoint and the projector exit pupils cannot be
coincident. Consequently, images of straight lines are
seen as curved lines when viewed from the viewpoint
[8,9]. Scenes which are generated as described above can
be made to appear free of distortion when viewed from any
viewpoint location by predistorting the image. The display
distortion is corrected using a non-linear mapping in the
image generator. Object points are remapped in the CIG
computation plane to provide the required object's
predistortion before being placed on a rectangular raster
for projection. In this way the visual system is able to

compensate for geometrical and optical distortions of
images projected onto a spherical dome surface [9,10].
This allows the scenes to appear free of distortion when
viewed from a single viewpoint by predistorting the image
and projecting the predistorted image. After
predistortion takes place, video can be generated in a
line-by-line, pixel-by-pixel basis. Figure 3 shows how
predistortion is handled to provide an image which, when
displayed on the dome, is visually correct.

Figure 3. Predistortion Process [11]

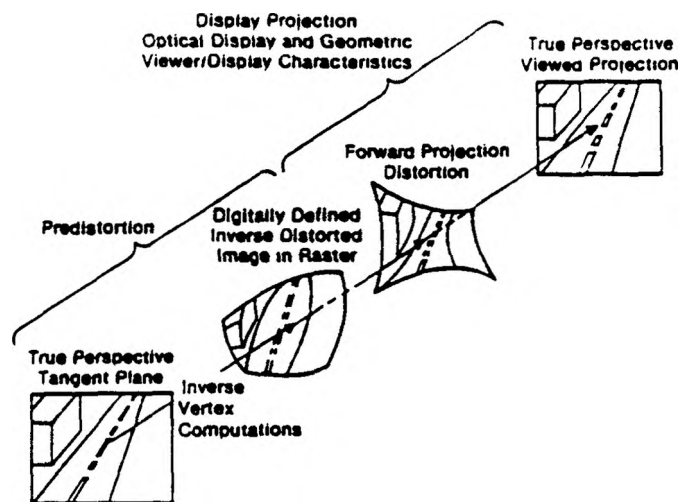Combining Displays. Because more than one projector is used to provide a continuous image throughout the dome, any discontinuity between the separately displayed images results in an undesirable artifact. The projected images must be joined together in a seam that is indiscernible, yielding a composite image comprising the out-the-window display. When using a configuration as in Figure 1, the two projected hemispheres must provide one continuous display with no abrupt distortions or discontinuities between the two hemispherical images. Providing one continuous display from two hemispherical images can be accomplished by adjusting the intensity of the video before it is processed.

By defining a region in the dome over which both hemispherical displays provide coverage, it is possible to adjust the intensity of every element in this region such that the region is blended together to provide one continuous display. The control of the intensity of the area which is to be blended is handled through a software algorithm which defines how video memory must be modified for blending to result. Blending makes images as realistic as possible while providing a low-cost, easily implemented method of joining displays. The following describes the algorithm and how it is currently being used in flight simulation.

HEMISPHERICAL DISPLAY CONTINUITY

Using two projectors to provide full wrap-around display coverage in a dome demands that the two displays come together somewhere within the aircrew's field of view. Of utmost importance is that these two images provide one continuous display with no abrupt discontinuities. Two approaches can be considered in solving the critical problem of joining two hemispherical displays together in an indiscernible seam: 1) butting displays together, or 2) blending displays together.

Butting Displays Together. Butting two hemispherical displays together is the simpler, though potentially more inaccurate, method of providing one continuous display. In theory, this method can be successful. If both hemispherical displays are adjusted to exactly the same color, it is possible to position the two light valve projectors so they are perfectly aligned. However, light valves change or "drift", with time, both in color and position. Light valve color output can be adjusted manually; however, due to drifting, the adjustment is not guaranteed to hold for any significant duration of time. Furthermore, the drifting experienced by light valves manifests itself differently for each light valve. The ambient conditions surrounding each light valve can also add significantly to the drifting.

Any movement of the projector or the video output from the projector, including the positional drift mentioned above, results in geometrical inaccuracies which produce discontinuous alignment of the two displays. These limitations prohibit this method from serving as a viable solution for providing hemispherical display continuity.

It is possible to accomplish edge matching by positioning the view windows in the CIG system through the position and attitude controls of the CIG [12]. This method allows for small variations in the position and drift of the light valves. However, it requires significant manual intervention which precludes its use as a simulator tool.

Blending Displays Together. The second method, blending hemispherical displays together, suffers from none of the limitations discussed above. Blending can be handled optically or through software-controlled intensity modifications. To blend two images together optically, a significant amount of special hardware must be designed and developed [13]. Using a software-controlled blending algorithm requires modulation of video intensity signals along a blending band such that the result is a modified video signal wherever blending is to occur. Figure 4 shows the dome with the channel boundary surrounded by a three degree blending region. With two adjacent channels matched in luminance and color, the system will render the
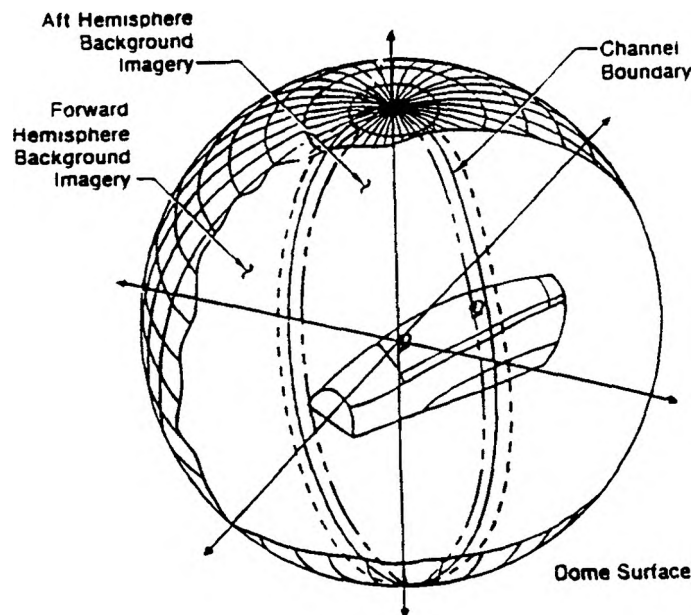
Figure 4. Dome Blending Region [7]

boundary between the two hemispherical channels nearly
indiscernible to the aircrew. There will be no noticeable
variation in luminance, color or contrast across the
boundary. The blending of the two hemispherical channels
is accomplished in the region formed by overlapping the
hemispherical images. Hardware postprocessing is used to
accomplish this blending. Figure 5 shows a diagram of
this process. The system consists an IBM-PC/AT fitted
with a specially developed graphics controller board. It
also includes special purpose video multiplier boards
which are mounted inside the IBM-PC/AT. These video

Image Generator → PC → Forward Light Valve →

PC → Aft Light Valve →

To Dome Surface
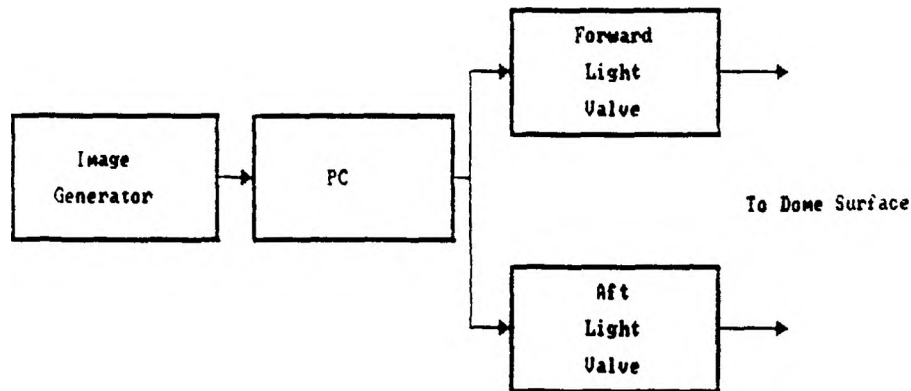
Figure 5.   Blending Process Flow Chart

multiplier boards, in conjunction with the graphics
boards, accomplish the intensity blending. The blending
system provides smooth matchup of the discontinuities at
the boundary between adjacent fields by adjusting the
intensity signals of each channel within the blending
region. This adjustment is performed using a specially
designed software algorithm which controls the video
signal gains at the edges of the projection hemisphere.

The algorithm actually modifies the elements of a
lookup table. The values in this table specify the
intensity of each pixel in every raster line. The
intensity of all (x,y) elements within the blending region
will be modified according to the algorithm to be
discussed in the following sections. The output of the

table lookup is then used to control the gains for
intensity of the color video signals.

The actual data written to video memory consist of
indices into a color pallet. Each entry in the color
pallet contains three video level data parameters, one
each for red, green and blue. If all three color
parameters are the same, a shade of grey from black to
white is the result. The blending algorithm does not deal
with colors separately, but with intensity. It is a
prerequisite that the output of the projectors be matched
in color by a manual adjustment of the projectors. By
adjusting brightness the blending band covered by both
displays will not appear brighter than the surrounding
display.

BLENDING REGION DEFINITION

To provide a region of overlap in which blending
occurs, each of the two hemispherical displays must cover
an area greater than 180 degrees. In the work described
here, three degrees was chosen as the width of the region
which provides a sufficient region for blending yet a
narrow enough region to minimize the additional images
that must be generated during real-time operation. Using
a three degree blending region requires each hemispherical
display cover 183 degrees. Figure 6 shows an expanded

hemispherical display. The area covered within the inner
solid line represents the 180 degree coverage. The outer
solid line represents the extended 183 degree coverage
which allows a one and a half degree overlap in all
areas. When both displays are combined, a three degree
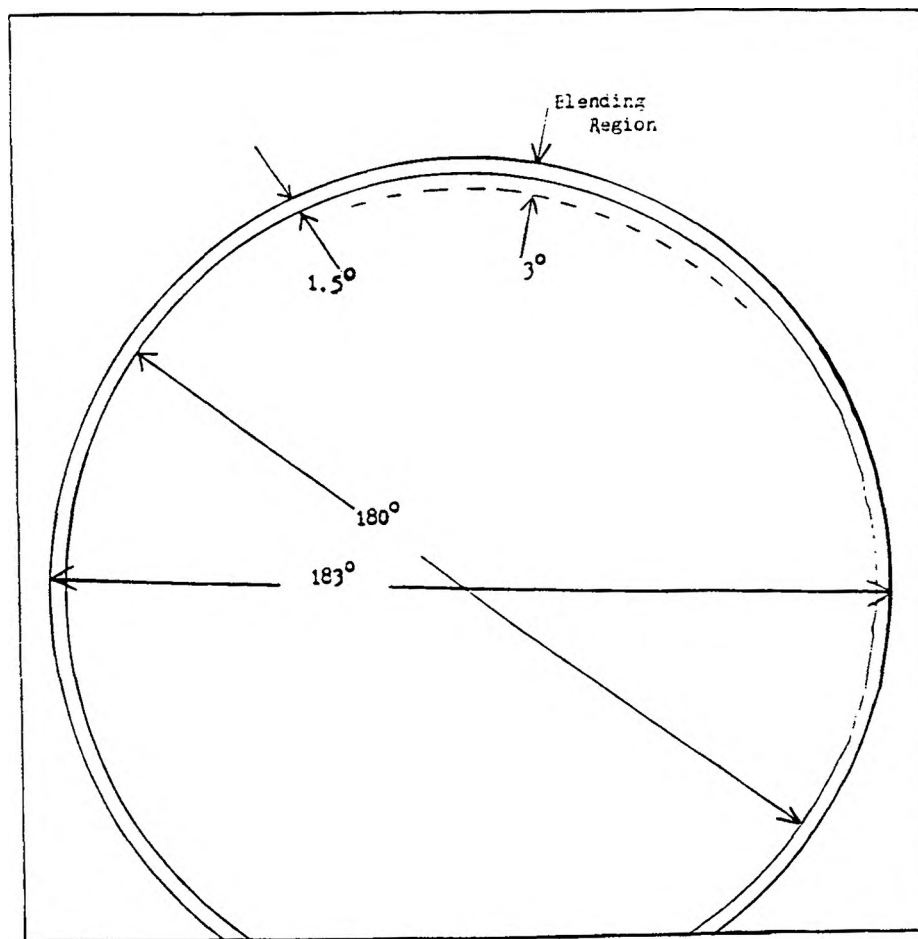blending region results.

Figure 6. Single Projector Display Coverage

Actual Blending Region. By placing indicators of some
sort along the equator separating the hemispherical
displays, the ideal location of the 180 degree field of

view can be determined. These indicators are placed at 10 degree intervals. With each centerline indicator is an associated indicator. The associated indicators are placed on alternating sides of the centerline at one and a half degrees from the centerline, forming the three degree blending region is defined as shown in Figure 7. The indicators are represented by Xs. The exact locations of the indicators are determined when they are installed in the dome. However, due to light valve drift, the perceived location of each indicator may not stay constant. If it were possible to assure that no effect from light valve drift would occur, blending would have to be done only once as the perceived location of the blending region would stay constant as long as the location of the image on the dome stayed constant. To deal with this problem, the perceived location of the indicators defining the blending region must be found.
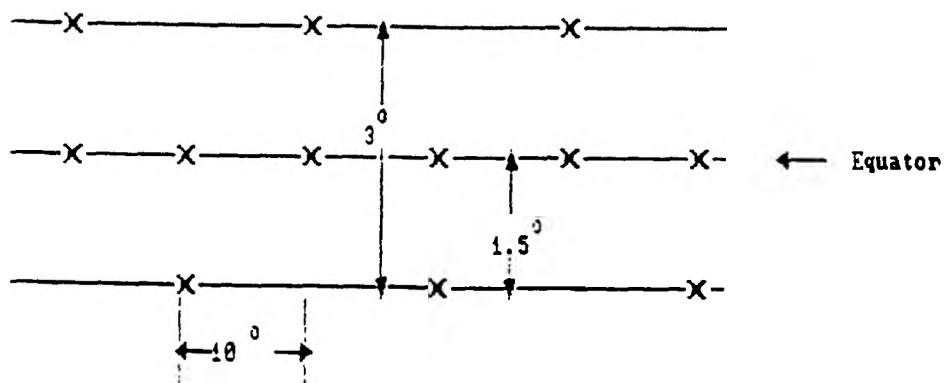

Figure 7. Blending Region Indicator Locations

Perceived Blending Region. During the adjustment phase, the CIG projects a pair of crosshairs on the dome. The intersection of these crosshairs is positioned over one of the indicators defining the blending region, and the coordinates are noted. The perceived azimuth and elevation of that indicator can be found and stored for use in later computations. This procedure is repeated for each indicator. All references to position indicators which follow refer to the perceived locations.

The blending region is comprised of blending segments defined by indicators in the dome. Each blending segment is defined by two sets of indicators, one with center and an outer indicator and the other with center and an inner indicator as seen in Figure 7. After the perceived location of all indicators is known, the inner and outer boundary points associated with each centerline point, but not represented by an indicator, can be found. This is done by taking the vector from the existing boundary point, either inner or outer, to the associated centerline point, scaling the result by two and adding it to the existing boundary point to generate the imaginary boundary point. This results in an exact definition of the blending band and of each blending segment as is shown in Figure 8.
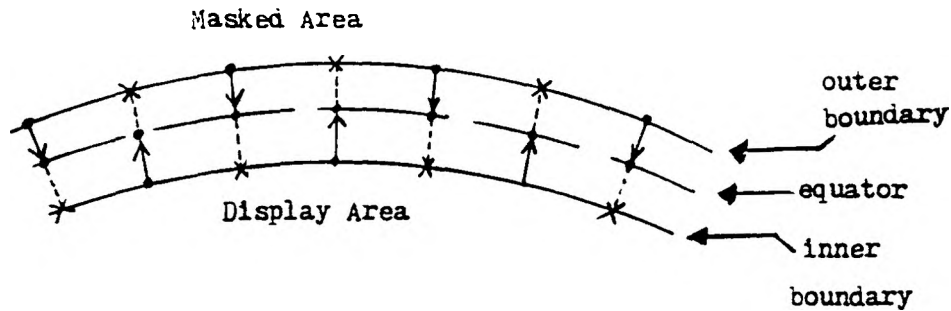
Figure 8.   Imaginary Boundary Point Locations

This process allows the position of the perceived boundary points as well as the imaginary boundary points to be defined. The location of the inner and outer boundary points associated with each centerline boundary point allows the blending region to be divided into blending segments. By forming the segments defined by each associated inner and outer boundary point, the boundary segments appear as shown in Figure 9. These segments allow for a more accurate definition of the blending elements which can be determined after their relationship to standard space is found.

Figure 9. Blending Region Segments

STANDARDIZED SPACE

The definition of the blending region and each
blending segment is used to determine how video memory
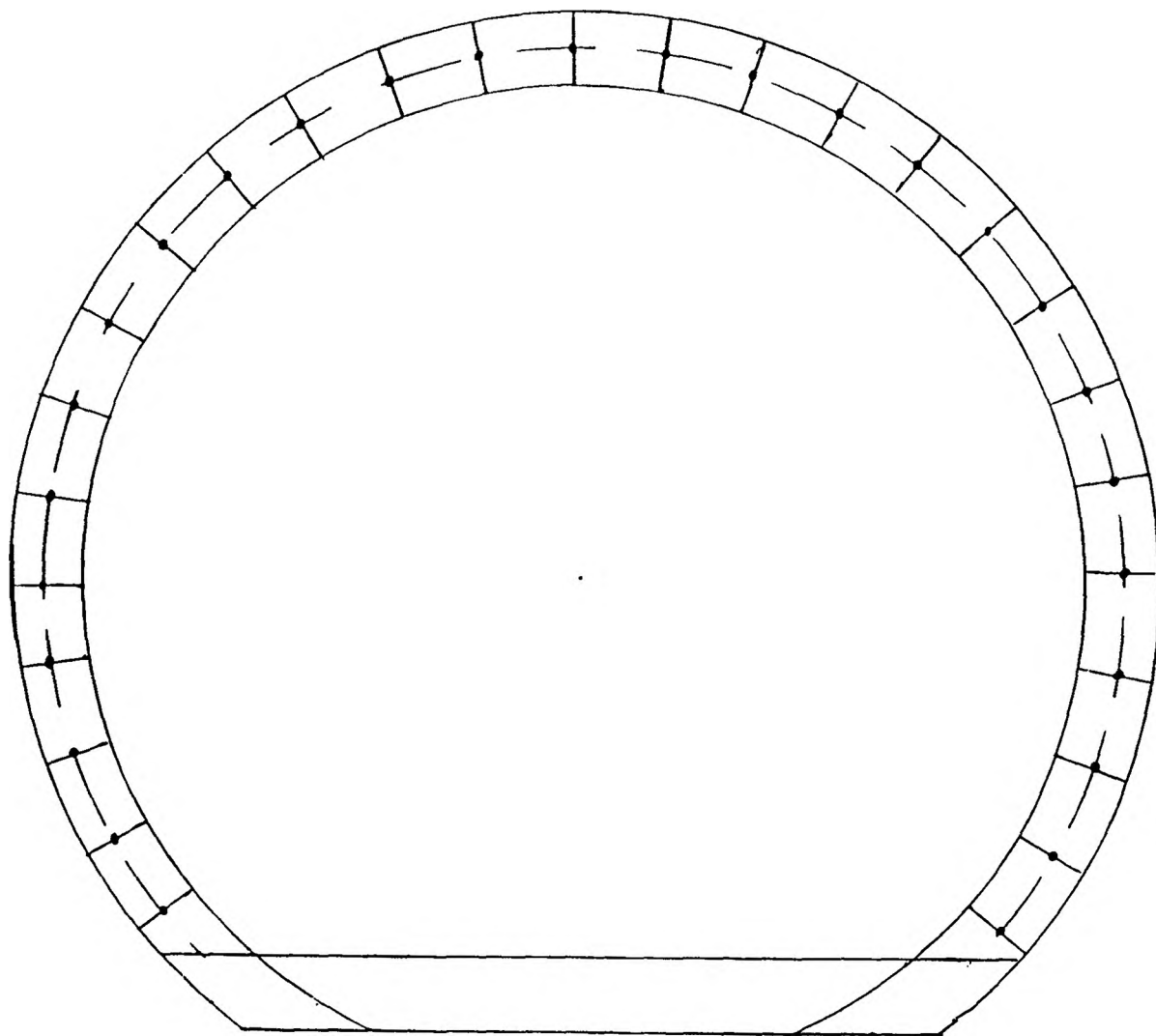must be modified to provide blending. Video memory has a
one-to-one correspondence to raster space. Some method

must be used to assure evaluation of each element that must be modified. Since realism in background displays requires a minimum resolution of 1024 x 1024 raster elements, individual evaluation of each element would be a time consuming task requiring evaluation of over 1,000,000 elements per display. To simplify this procedure, a method was sought in which a standard procedure could be used to provide the calculations necessary with the least amount of repetition.

To this end, a standardized space was defined. Standard space is the common orientation into which each quadrant can be mapped to expedite the raster line evaluation. This mapping is done by using a unique transformation matrix for each quadrant. The concept of standard space is very powerful and is vital to the blending algorithm.

By defining standard space as shown by the representative quadrant in Figure 10, the number of unique cases is greatly reduced and the relationship of each raster line to the outer and inner boundaries can be defined. Any raster line in standard space may intersect the outer boundary at most once. When the outer boundary is intersected, the raster line may intersect the inner boundary at most once, as shown by raster lines A and B. The case in which the outer boundary is not intersected is trivial, as shown by raster line C.
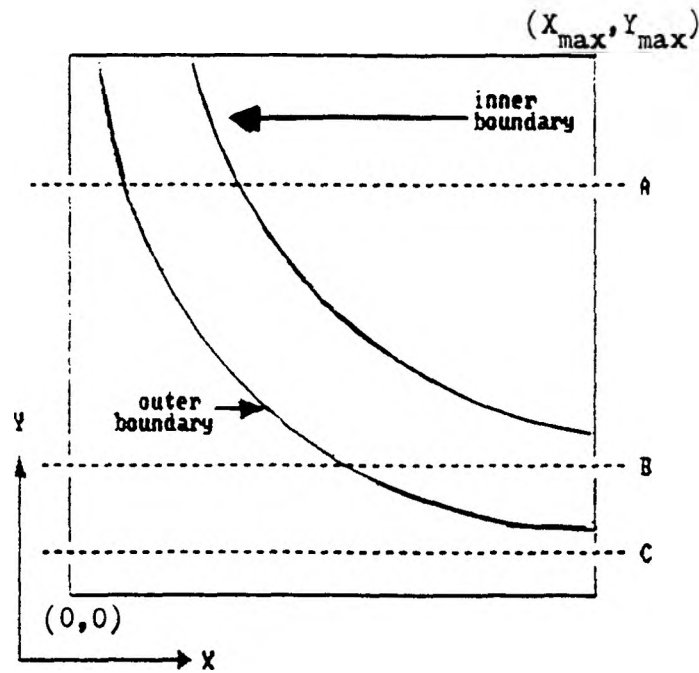
Figure 10. Standard Space Representative Quadrant

By dividing raster space into four quadrants, the algorithm to provide blending can be drastically simplified. Figure 11 shows how raster space is divided. Each element of raster space, (X_R, Y_R), can be mapped into standard space by the equation

$$\begin{bmatrix} X & Y \end{bmatrix} = C_i \begin{bmatrix} X_R \\ Y_R \end{bmatrix}$$

which uses a unique transformation, C_i, for each quadrant as shown below, where i represents the quadrant being referenced.

$$C_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad C_2 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$C_3 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad C_4 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$
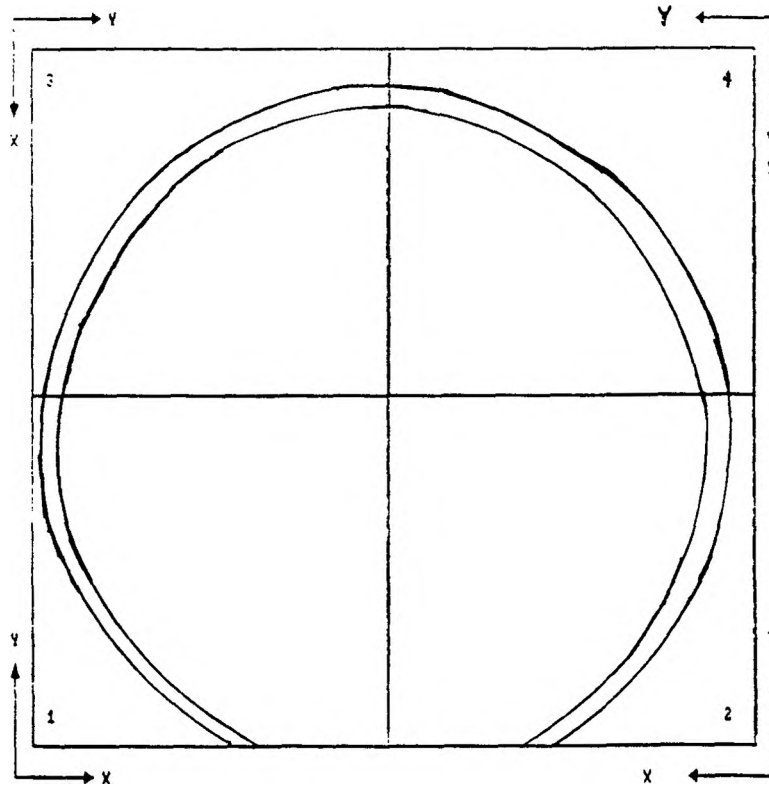
Figure 11. Quadrant Definition

Standard space requires the blending region be
positioned in the quadrant such that a raster line will
intersect the outer boundary before intersecting the inner
boundary, when evaluating it from left to right (see
Figure 10). The area to the left of and/or below the
outer boundary will be termed the masked area, while the
area to the right of and/or above the inner boundary is
the display area. The blending region separates the

masked area from the display area. Figure 12 represents
each quadrant and the orientation of the blending region
after it has been mapped into standard space.

Original Orientation

Quadrant 1   Quadrant 2   Quadrant 3   Quadrant 4

In Standard Space

Quadrant 1   Quadrant 2   Quadrant 3   Quadrant 4
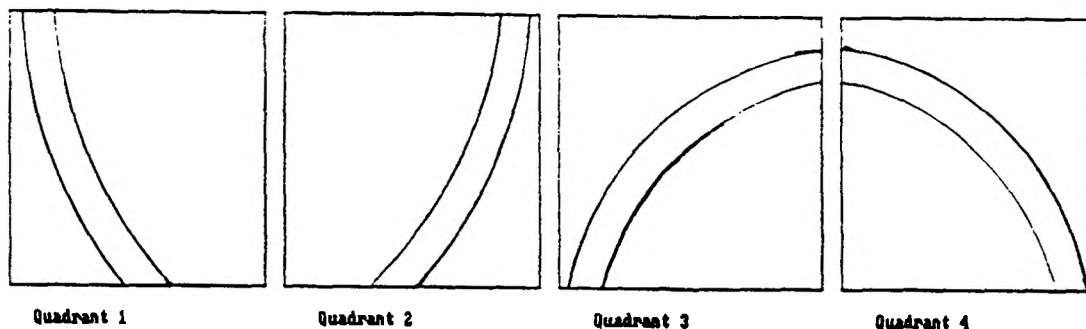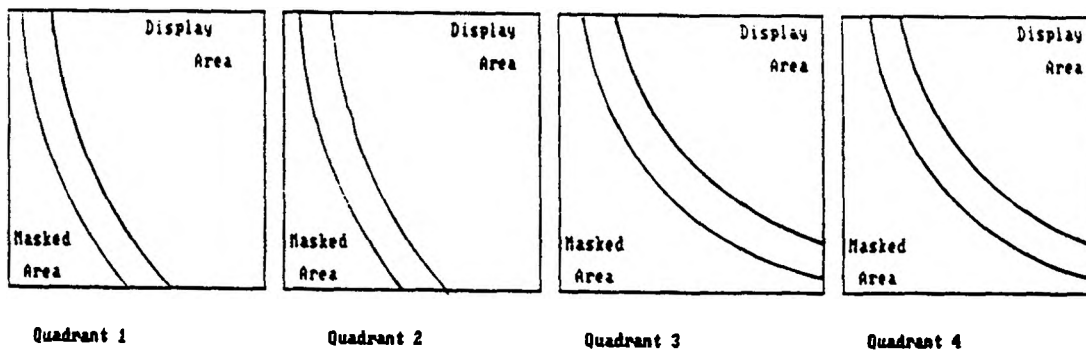
Figure 12. Quadrants Transformed to Standard Space

After blending for the quadrant has been completed,
each element need only be transformed back to raster space
using the appropriate inverse transformation. The result
is then sent to video memory. The blending and raster
fill processes can always take place from left to right,
bottom to top, by mapping each quadrant into standard
space.

RASTER LINE AND BLENDING REGION RELATIONSHIP

Blending will affect all elements on a raster line between the outer and inner boundaries of the blending region. However, each element along a raster line must be evaluated to determine which elements are located within the blending region, and which will not be affected by blending.

To perform blending, video memory is adjusted according to values in a table lookup. The table lookup contains normalized values, where zero represents black, or no intensity, and one represents white, or full intensity. Elements in the masked area are blacked out by assigning them a value of zero. Elements in the display area are to be unmodified and are assigned a value of one. The elements between the outer and inner boundaries of the blending region on each raster line will be ramped between one and zero using a linear ramping function. A linear ramping function was chosen for simplicity and speed of evaluation. However, other ramping functions could be used. Figure 13 shows the ramping function as used on each hemispherical display, and as the two hemispherical displays are combined.
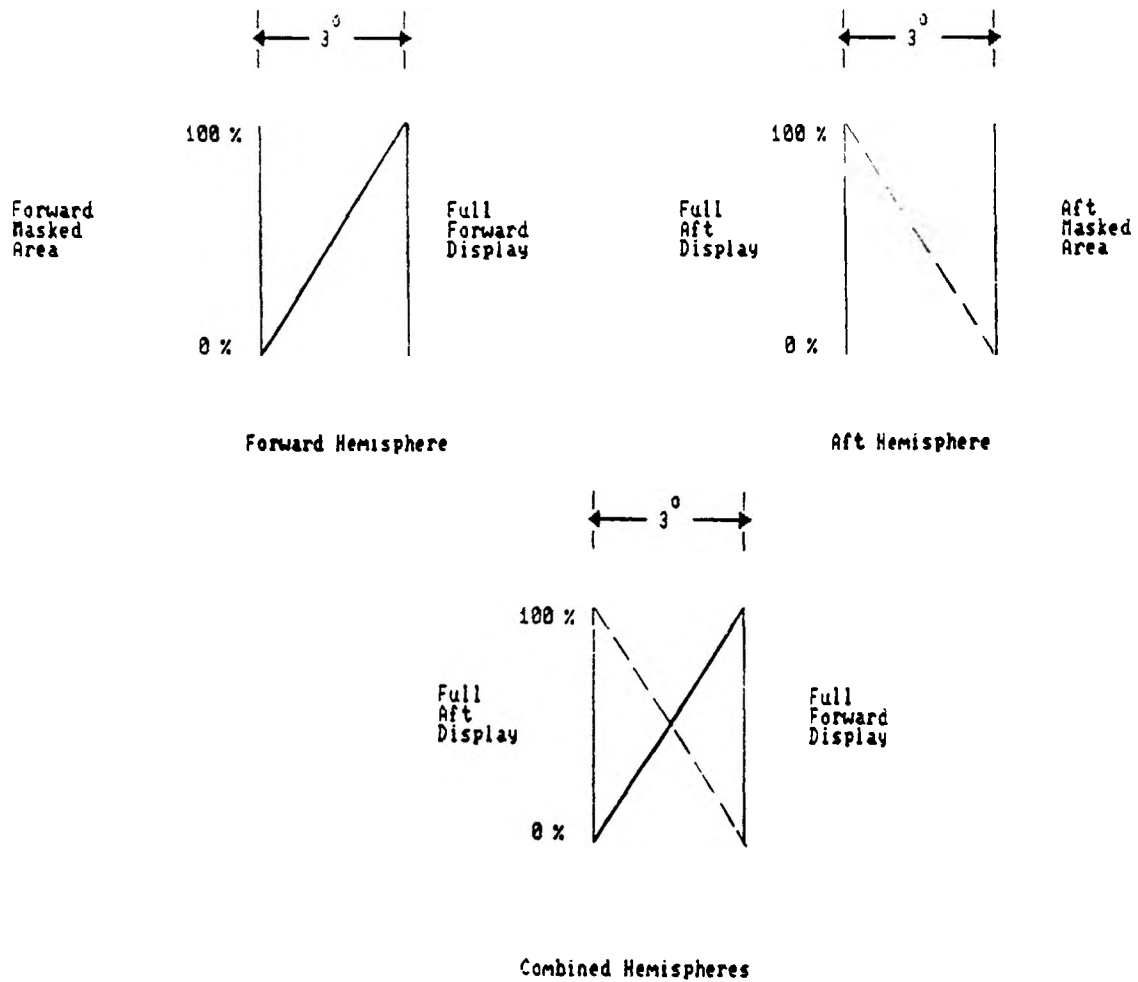
Figure 13. Video Intensity Ramp

The blending values to be inserted in the video memory table lookup are evaluated individually for each raster line. Further, each raster line is evaluated separately, starting at the bottom line and working upward.

First Raster Line Evaluation. It is apparent that one of
two situations occur for the first raster line evaluated:

      1) it intersects the outer blending boundary, or

      2) it does not intersect the outer blending
      boundary.

These two situations are shown in Figure 10.

To determine which of the above situations applies,
the intersection of the outer boundary with the vertical
line $x = X_{max}$ must be determined. If the intersection
falls below the quadrant, the first raster line will
intersect the outer boundary.

To calculate the point of intersection, observe that
the outer and inner boundaries closely approximate
circles. If the equation of the circle defining the outer
boundary can be determined, computing the intersection of
the raster line and the circle is trivial.

One way to approach this is to use all outer boundary
points in determining the equation which would approximate
a circle containing all points. This would be a one-time
effort which would loosely approximate the definition of
the outer boundary. Depending on the location of the outer
boundary points, both perceived and imaginary, this
approach would likely introduce a relatively large error.
Therefore, the equation of the circle to be used will
approximate the outer boundary at the arc defined by the
three closest outer boundary points.

The three points to be used to define the circle are the three outer boundary points closest to the Y value of the raster line. This is shown in Figure 14. These points are labeled P_1, P_2, and P_3, as shown in Figure 14.
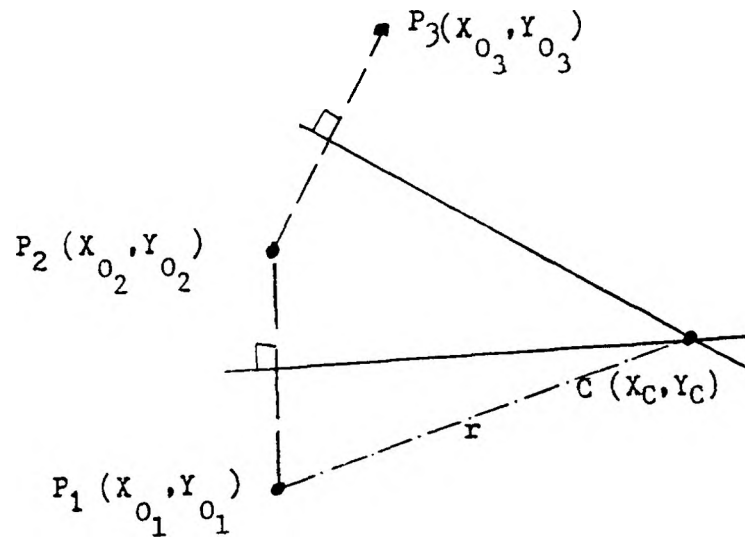

Figure 14. Circle Definition From Three Boundary Points

The circle's center (X_C, Y_C) is the point of intersection of the perpendicular bisectors of the line segments P_1P_2, and P_2P_3. The equation of the circle containing these outer boundary points is

$$(X - X_C)^2 + (Y - Y_C)^2 = r^2 \tag{1}$$

where $r = \sqrt{(X_0 - X_C)^2 + (Y_0 - Y_C)^2}$

Having determined the equation of the circle , it is possible to determine the point (X_{max}, Y) where the circle intersects the vertical line $x = X_{max}$. Since the blending algorithm requires integer values, it may be necessary to truncate the Y value. Let the resulting value to denoted by Y_{min} . The last raster line which does not intersect the outer boundary is $y = Y_{min}$. If Y_{min} is less than zero, all raster lines in the quadrant being evaluated will intersect the outer boundary. Otherwise, each pixel of raster lines with a value of Y_{min} or less will have an associated value of zero, indicating this area is masked out of the display. Because of the definition of standard space, every remaining raster line $y = Y_j$ where $Y_{min} < Y_j < Y_{max}$ and where Y_{max} is the value of the uppermost raster line, will intersect the outer boundary. In many cases it will also intersect the inner boundary.

Case Definitions. Figure 15A shows blending segments formed by connecting the associated inner and outer boundary points. Figure 15B shows the three cases which can result as a raster line crosses the blending band. Case 1 occurs when only one segment is intersected. When more than one segment is intersected, either case 2 or case 3 exists. The slope of the line connecting the two intersected segments determines which case exists; when the slope is positive, case 2 exists; when negative, case 3 .
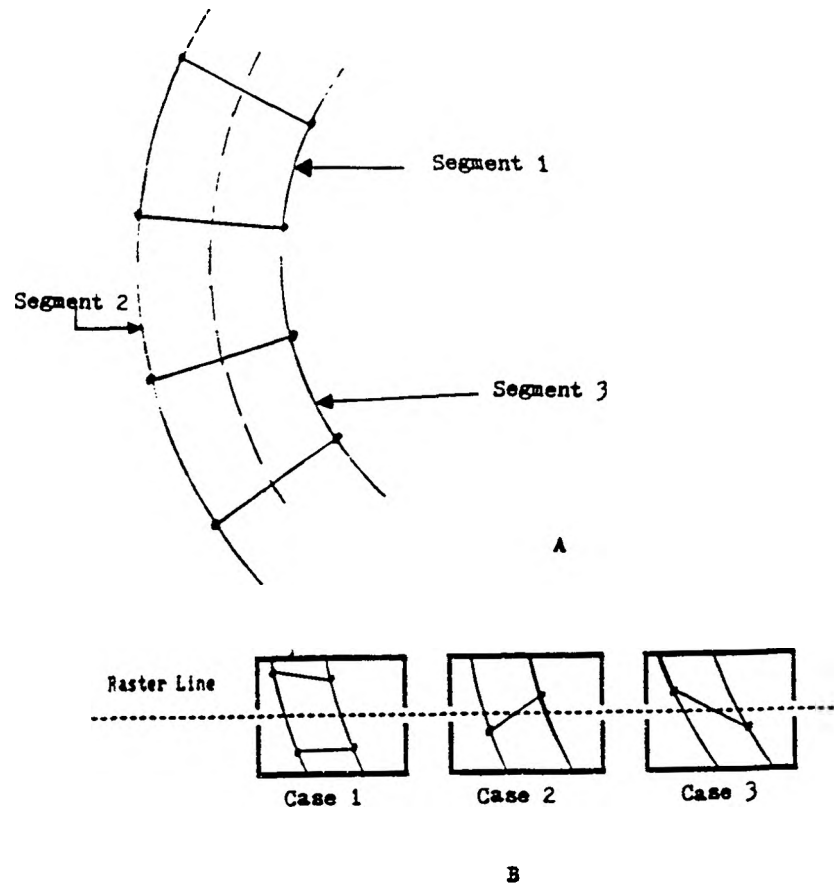
Figure 15. Blending Band Case Definition

Case 1 is the simplest; it covers the situation where a
raster line y = Y_j intersects only one blending segment.
The value of each pixel can be evaluated strictly on its
position relative to the start and end of the blending
region. In this case, the only remaining information to be
determined is the x values of the outer and inner
boundaries. Knowing the beginning and ending points allows
the value of all pixels to be determined during the raster
line fill process.

The Average Circle. To find the intersection of the
raster line $y = Y_j$, $Y_{min} < Y_j < Y_{max}$, with the
outer boundary, the equation of the outer boundary is
determined in a manner similar to that for determining
(X_{max}, Y) from equation (1). The problem varies from the
earlier description in that two cases may occur as shown in
Figure 16. Figure 16A is analogous to the situation in
Figure 14. Where the raster line crosses the arc with
endpoints P_0 and P_1, it is clear that P_2 will be used
as the third point in determining the equation of the
circle containing P_0 and P_1 because it is the only
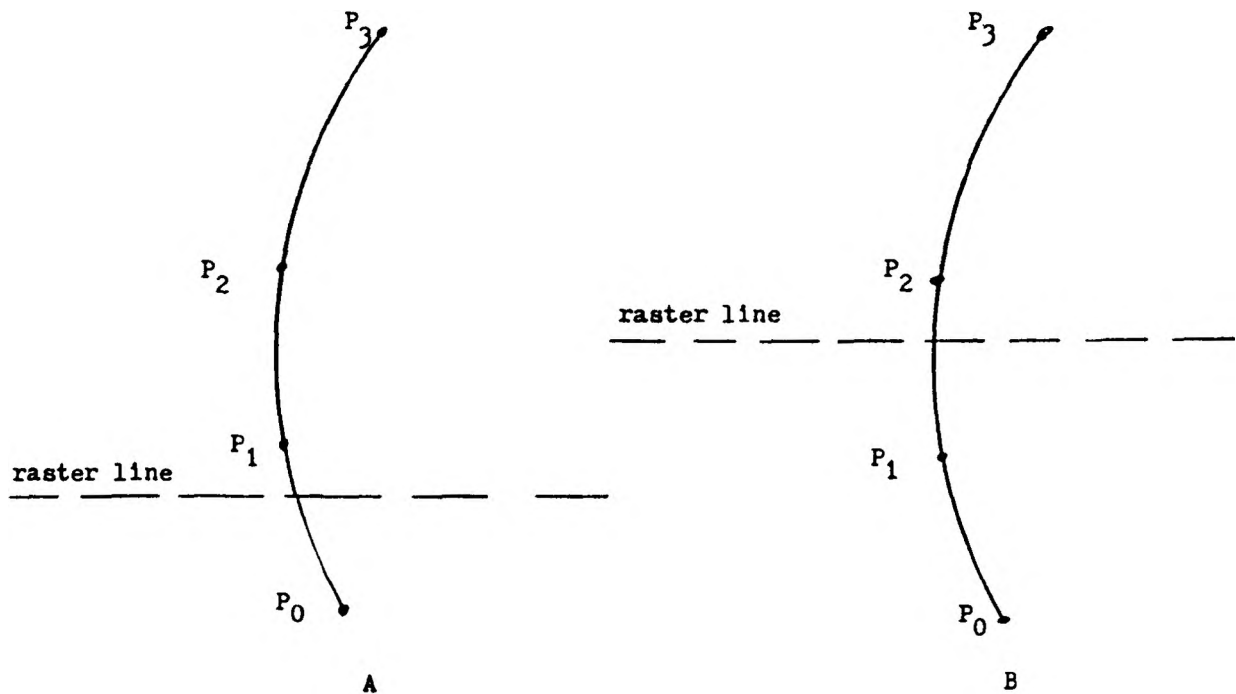point adjacent to the arc.

Figure 16. Point Selection for Circle Definition

The second possibility, shown in Figure 16B, has boundary points existing on either side of the intersected arc. A slightly different method has been developed for determining a more accurate equation of the circle. The method utilizes all four points by constructing the equations of two circles, each using one of the additional boundary points. These circles can then be used to construct an "average" circle (see Figure 17). The radius of the circle is found by averaging the two radii r_1 and r_2. The center is found by averaging the x and y values from the circle centers already found:

$$(X_C, Y_C) = ((X_{C_1} + X_{C_2})/2), (Y_{C_1} + Y_{C_2})/2)$$

The intersection point of the current raster line $y = Y_j$ and the average circle as defined in equation (1) is (X_{0_j}, Y_j) where

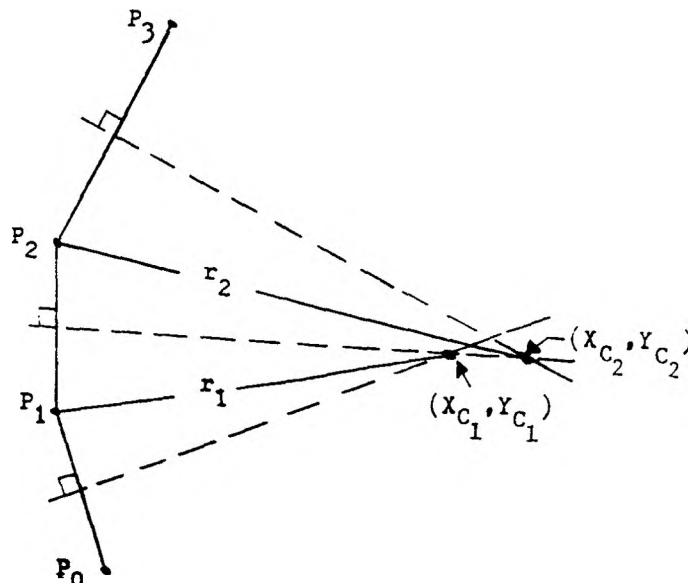$$X_{0_j} = X_C - \sqrt{r^2 - (Y_j - Y_C)^2}$$

Figure 17. Average of the Circles

The same procedure is used to find the intersection of the raster line with the inner boundary, yielding the point of intersection as (X_{I_j}, Y_j). Once (X_{0_j}, Y_j) and (X_{I_j}, Y_j) are known, all pixels on the raster line $y = Y_j$, for $X_{0_j} < X < X_{I_j}$ can be filled using the raster fill scheme to be discussed in the next section.

Crossing Blending Regions. As raster lines are evaluated from bottom to top it is clear that eventually a raster line $y = Y_j$ will intersect more than one blending segment, resulting in either case 2 or case 3 (see Figure 15B). Let the outer and inner boundary points on the segment defining the separation of blending segments be (X^*_0, Y^*_0) and (X^*_I, Y^*_I), respectively. Any raster line $y = Y_j$, such that

$$Y^*_0 > Y_j > Y^*_I \qquad \text{(Case 2)}$$

$$\text{or}$$

$$Y^*_0 < Y_j < Y^*_I \qquad \text{(Case 3)}$$

will pass through more than one segment. In these cases, the point of intersection with the segment connecting the outer and inner boundary point must also be determined.

The intersection of the raster line with the outer and inner boundary is found in the same manner as for case 1. The equation of the line formed by the inner and outer boundary points is:

$$y = Y^*_0 + m * (X - X^*_0) \qquad\qquad (2)$$

where
$$m = \frac{(Y^*_0 - Y^*_I)}{(X^*_0 - X^*_I)}$$

Substituting Y_j for y into equation (2) yields X_{mid},
the X value of the intersection (see Figure 18). It plays
a significant part in determining how the raster fill
process is conducted for case 2 and case 3. Once a raster
line has been evaluated and determined to satisfy the
requirements for case 2 or case 3, each succeeding raster
line, as they are evaluated upward, will be of the same
case type until such time the line again intersects only
one blending region, satisfying the requirements for case 1
again. For example, in Figure 18, case 2 will exist for
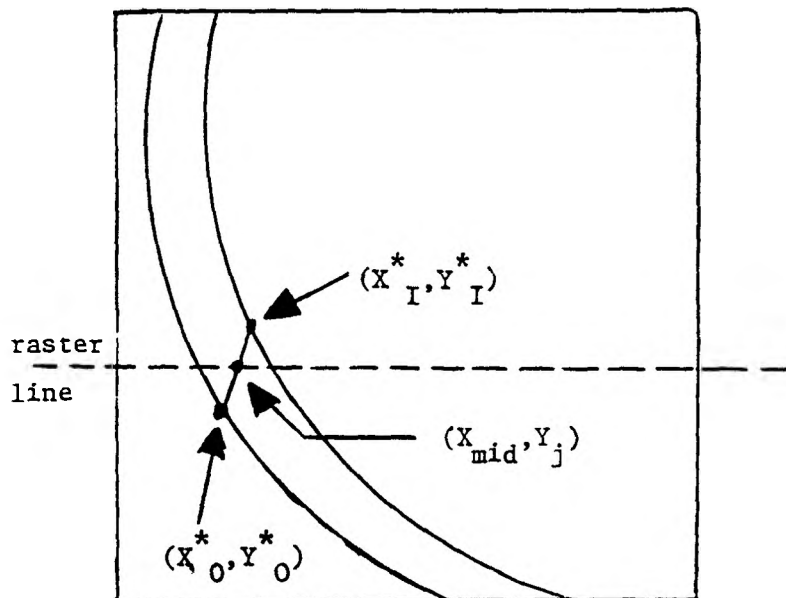each raster line y = Y_j until $Y_j > Y^*_I$, at which
time case 1 will again exist.

Figure 18. Raster Line Intersection With
Segment Boundary

RASTER LINE FILLING

The raster line filling process uses the blending region information described in the previous sections and stored in the lookup table to perform the final step of determining how video memory must be modified to achieve the goal of blending the two hemispherical displays together.

In Figure 19, a typical raster line and blending region segment are shown. In standard space the area to the left and/or below the blending region is its masked region. It will have an intensity represented by zero. All pixels to the right and/or above the blending region will have an intensity represented by one since they are in the display region. Hence, all pixels outside the blending region can immediately be assigned a value of zero or one.
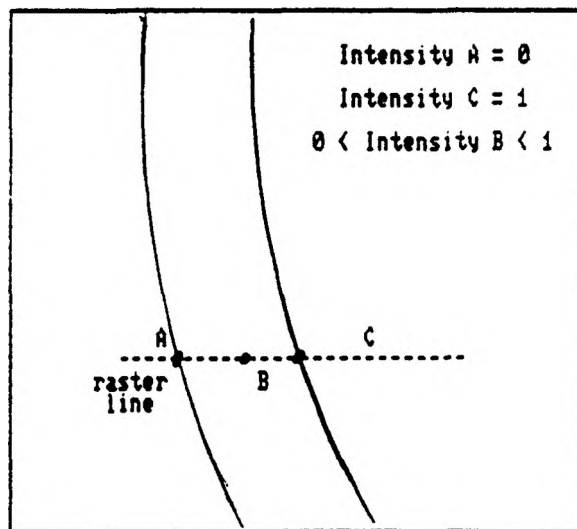

Figure 19. Blending Region Intensity

Raster Filling for Spans. Although writing to video memory is always done on an element by element basis, when it is desired that constant values be written out for pixels located consecutively on a raster line, the raster line filling process requires only the starting and ending locations of the span plus the pixel value.

Before the process of writing to video memory can begin, the span under consideration must be transformed from standard to raster space. For a span in standard space along the line $y = Y_j$ let the starting and ending points defining this span be (X_S, Y_j) and (X_E, Y_j), respectively. The inverse transformation matrix C_i^{-1} must be applied to each of these two points to transform them from standard space to raster space. The result of transforming these points back into raster space yields (X_{SR}, Y_{SR}) and (X_{ER}, Y_{ER}).

When the starting and ending y values of the span in raster space are the same, $Y_{SR} = Y_{ER}$, the span is located on a single raster line and all values of video memory between X_{SR} and X_{ER} on the line $y = Y_{SR}$ are set to a constant using a poke process. In this case there is no need to evaluate each element individually. However, when $Y_{SR} <> Y_{ER}$, the span is not on one raster line and each pixel on the span in standard space must be transformed to raster space individually before a poke process can be performed to modify video memory.

Blending Region Filling. The elements on a raster line within the blending region must be handled individually since no spans exist in the blending region for which it can be assumed that a constant value will be assigned. In the blending region each pixel location along the raster line in standard space is transformed to raster space using the appropriate inverse transformation matrix C_i^{-1} and the value to be assigned is determined. Again the idea of three distinct cases being handled is used in the raster fill process (see Figure 15B).

When case 1 exists, where the raster line intersects one blending segment only, each pixel on the raster line is assigned a value based on its position relative to the outer and inner boundary locations on the raster line. The x value for both the outer and inner boundaries on the raster line being evaluated are known as X_0 and X_I, respectively. The distance, D_0, from any point X, $X_0 < X < X_I$, to the outer boundary, X_0, is found by

$$D_0 = X - X_0$$

while its distance from the inner boundary, D_I, is

$$D_I = X_I - X$$

The value to which the element being evaluated is to be set in video memory, K_p, can now be found by the following:

$$K_p = K_{max} * D$$

where K_{max} is the value for maximum intensity and

$$D = D_0 / (D_0 + D_I) \qquad (3)$$

is a normalization factor. If a system is used which allows eight bits to determine intensity, $K_{max} = 255$.

When cases 2 or 3 exist, the point of intersection (X_{mid}, Y_j) with the raster line $y = Y_j$ (see Figure 18) is also used in the evaluation. This point allows the segment to be divided into two subsegments, the first between X_0 and X_{mid} and the second between X_{mid} and X_I. Each subsegment can be evaluated separately in a manner similar to that described for use when case 1 conditions are satisfied if the intensity associated with the point (X_{mid}, Y_j) is known. This value, known as K_{mid} is found by computing:

$$K_{mid} = K_{max} * (X_{mid} - X_0) / (X_I - X_0)$$

which assigns the intensity at the intersection point as a percentage of the maximum intensity, set according to the location of intersection point with respect to the outer and inner boundaries.

The first subsegment, from X_0 to X_{mid}, is filled in the same manner as for case 1 but substituting X_{mid} for X_I. This would result in

$$D_0 = X - X_0$$
$$D_I = X_{mid} - X$$

where $X_0 < X \leq X_{mid}$. Using D_0 and D_I, the normalization term D is found from equation (3). Similarly, when $X_{mid} < X < X_I$, X_{mid} is substituted for X_0 such that

$$D_O = X - X_{mid}$$

$$D_I = X_I - X$$

The associated intensity, K_p, can be found from the following:

$$K_p = K_{mid} * D \qquad \text{when } X_O < X < X_{mid}$$

$$\text{and}$$

$$K_p = K_{max} * D \qquad \text{when } X_{mid} \leq X < X_I$$

RESULTS AND CONCLUSIONS

The technique described for intensity blending of computer image generation-based displays was successfully implemented on Naval Training Device 2E6, Air Combat Maneuvering Simulator, NAS Oceana, by McDonnell Aircraft Company. This was the first opportunity which allowed this procedure to be demonstrated. Using a General Electric Compu-Scene III computer image generation system, hemispherical background images were generated which were projected using two light valve projectors [14].

The ramping function originally used was a linear ramp which combines display intensities so they add up to 100 percent [15]. This offset ramp may prove to satisfy other applications as satisfactorily as it has this one. However, it is anticipated that given other applications, with other light valves or optics packages, that a new offset ramp would have to be found. After much empirical work, an offset ramp, as in Figure 20, was decided upon.

The offset ramp proved more pleasing because of the response of the light valves. The light valve response showed that, ideally, a smaller offset was needed for brighter scenes, and a larger offset was needed for dimmer scenes. The 160/255 offset shown in Figure 20 was a compromise of the above requirements that proved best for this particular application. Insufficient data is available to determine if this offset ramp will satisfy all similar applications, or if it must be modified for each new application.
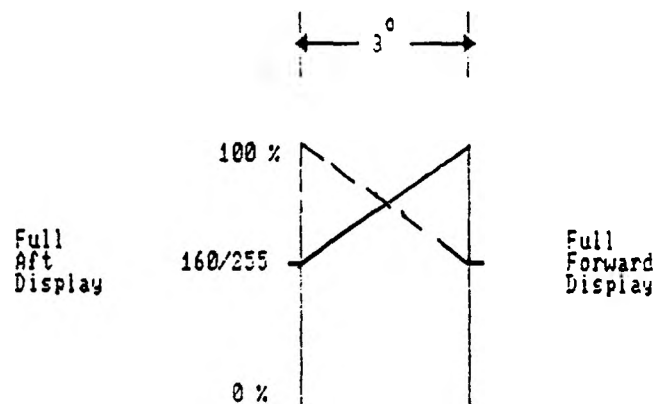
Figure 20. Offset Ramp Function

This application proved the blending algorithm can be used to control intensity when joining hemispherical displays. Some applications require non-hemispherical displays be projected onto the dome. For example, an area of interest display could be inserted as a patch into the forward display area, covering from 30 to 60 degrees. This

provides the capability of displaying a specific area at a higher resolution which would allow much more detail to be apparent. For air to ground work, such as take-offs and landings or bombing runs, this is an important tool. It is desirable to blend the forward display area with the inset such that the eye can perceive no abrupt change in the display.

With modifications to the blending algorithm to account for the different shapes, and a new definition of standard space for this application, it is anticipated that the blending algorithm can be as successful on this and any further applications which may arise as it has proven to be in its initial application.

REFERENCES

1. David B. Coblitz, "Training Techniques Using Computer Generated Imagery (CGI)," National Security Industrial Association Interservice/Industry Training Equipment 2nd Conference and Exhibition, 1980.

2. William B. Scott, "Visual Systems Refine Flight Simulation," <u>Aviation Week & Space Technology</u>, October 17, 1983.

3. "AV-8B Simulator Designed for Combat Missions," <u>Aviation Week & Space Technology</u>, November 1, 1982.

4. MCAIR-2E6, Compu-Scene III, <u>Operations and Maintenance Manual</u>, August 1986, Simulation and Control Systems Department, Defense Systems Division, General Electric Company, Daytona Beach, Florida.

5. Richard C. Hebb, "Visual Display Parameters," Simulation Technology Branch, Naval Training Equipment Center, Technical Report NAVTRAEQUIPCEN IH-356, September, 1984.

6. Kenneth J. Stein, "USAF Evaluates New Simulator Imagery," Aviation Week & Space Technology, June 11, 1984.

7. MDC B0517, Rev. A, Proposal for Modification of F-14A Device 2F112 WAVS, Volume IIA, 28 August 1987.

8. Richard C. Hebb, "Computer Program for Analysis of Spherical Screen Distortion," Technical Report NAVTRAQUIPCEN IH-332, March 1982.

9. Jerome T. Carollo and Nathaniel D. Reynolds, "Distortion Correction in Computer-Image Generation-Based Wide Angle Visual Display Systems", General Electric Company, Daytona Beach, Florida.

10. 2E6 CDC Calibration User's Manual, 25 August 1986, Simulation and Control Systems Department, Defense Systems Division, General Electric Company, Daytona Beach, Florida, Document No. 63A142676.

11. MDC A8426, Proposal for Computer-Generated Dynamic Earth/Sky Visual System for Device 2E6, Volume I, 30 September 1983.

12.   Jack W. Herndon, "A Visual Technology Research Simulator for Vertical Take-off and Landing (VTOL)," Advanced Simulation Concepts Lab, Naval Training Equipment Center, Orlando, Florida.

13.   Jerry L. Bentz, "Joining Techniques for Optically Combined Visual Display Systems," National Security Industrial Association/ Industry Training Equipment 2nd Conference and Exhibition, 1980.

14.   Kenneth J. Stein, "Simulation Techniques Converging to Meet Military/Commercial Needs," Aviation Week & Space Technology, March 18, 1985.

15.   F. D. Eckelmann, "Device 2E6 Blending Circuit Notes," McDonnell Aircraft Company, 1986.

VITA

Elizabeth Scheppler Reidelberger was born on September 7, 1954, in Davenport, Iowa. She received her primary and secondary education in Davenport, Iowa.

She attended at Fontbonne College, St. Louis, Missouri, where she received a Bachelor of Arts degree with majors in Mathematics and English in May, 1976. She has been employed at the McDonnell Douglas Corporation in the Flight Simulation Department and MCAIR Training Systems since June, 1976.

She has been an MS candidate in Computer Science at the University of Missouri-Rolla, Graduate Engineering Center in St. Louis since January, 1981.