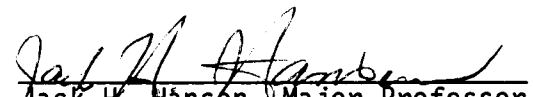
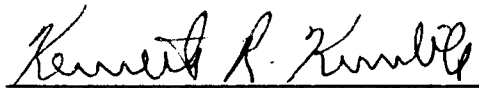


To the Graduate Council:

I am submitting herewith a thesis written by Mitchell J. Hurst entitled "Terrain Simulation of United States Geological Survey Digital Elevation Models." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Science.


Jack H. Hansen, Major Professor

We have read this thesis
and recommend its acceptance:


Kenneth R. Kimble


W.C. Armstrong

Accepted for the Council:


Vice Provost
and Dean of The Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the Library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor, or in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature Mitchell J. Hurst
Date 3/4/85

TERRAIN SIMULATION OF UNITED STATES
GEOLOGICAL SURVEY DIGITAL
ELEVATION MODELS

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Mitchell J. Hurst

March 1985

ACKNOWLEDGMENTS

In the course of thesis research, a number of scholars have assisted me generously. In particular I am indebted to Dr. Jack Hansen, Dr. Kenneth Kimble and Professor Wilbur Armstrong, who served as my advisor and committee, respectively. With their consultation and criticism I was able to achieve valuable results otherwise impossible. I would especially like to thank Dr. Jack Hansen for his unending practical outlook which was essential in formulating the goals of this thesis. I would also like to thank Mr. Bob Clemmons for valuable programming hints and Mr. Marshall Hutto and Ms. Rhonda Dawbarn for their helpful assistance while using the VAX 11/780 at The University of Tennessee Space Institute's Computer Center.

ABSTRACT

This thesis deals with the presentation of digital terrain in a three dimensional surface representation. The data base consists of digital spatial coordinate elevational data for terrain information based on the U.S. Geological Survey (USGS) Digital Elevation Models (DEM). Processing was accomplished on a VAX 11/780 with display on a Vectrix VX384. The analysis approach uses bicubic patch methods (surface equations) to depict aspect, slope, light source and viewer perspective. The points on each patch are projected to the respective pixels on the display device and the z-depth of each pixel is stored in a screen-buffer. Closer points take precedence over more distance points, thus producing a realistic scene from the digital data. Application examples include simulation scenes of selected flood levels and stereo pair generation.

TABLE OF CONTENTS

SECTION	PAGE
I. INTRODUCTION	1
A. Problem Definition :	1
B. Previous Research.	1
C. Thesis Goal.	3
II. HERMITE CUBIC CURVE.	5
A. Curve Point Calculation.	5
B. Point Normal Calculation	9
III. SURFACE PATCH DEFINITION	11
A. Patch Point Calculation.	11
B. Point Normal Calculation	14
IV. TRANSFORMATION OF DIGITAL ELEVATION MODEL TO SURFACE PATCHES.	16
V. HIDDEN SURFACE AND SHADING ALGORITHM	23
VI. DISCUSSION AND RESULTS	28
A. Discussion	28
B. Results.	30
VII. APPLICATIONS	38
A. Scientific Modeling of Natural or Manmade Events . .	38
B. Remote Education and Training.	40
VIII. CONCLUSIONS.	42
LIST OF REFERENCES	43
APPENDIXES	45
APPENDIX A. Digital Elevation Data Tapes.	46
APPENDIX B. Perspective Projection Matrix	49
VITA	59

LIST OF FIGURES

FIGURE	PAGE
II.1. Hermite Cubic Curve.	5
III.1. Surface Patch.	12
III.2. Calculation of da/dt and db/dt	13
IV.1. Southwest Region of an Example DEM	17
IV.2. SURFACE Array Containing Example DEM	18
IV.3. Surface Patch A of Example DEM	21
V.1. SCREEN BUFFER Array Employed to Keep Track of Closer Scene Points	24
V.2. Point Normal Vector and Sun Vector	26
VI.1. Contour Map Illustration of Thief Neck Island with Water Level Equal to 741-ft (225.9-m).	32
VI.2. Contour Map Illustration of Thief Neck Island with Water Level Equal to 800-ft (293.8-m).	32
VI.3. Computer Graphic Image of Thief Neck Island with Water Level Set at 741-ft (225.9-m).	33
VI.4. Computer Graphic Image of Thief Neck Island with Water Level Set at 800-ft (293.8-m).	33
VI.5. Photographic Stereogram of Thief Neck Island	35
VI.6. Computer Generated Stereogram of Thief Neck Island	35
VI.7. Photographic Ground Point Origin of Thief Neck Island.	36
VI.8. Computer Generated Ground Point Origin of Thief Neck Island	36
B.1. Perspective Projection	51
B.2. Line Segment AB.	51
B.3. Perspective Transformation T_1 of Segment AB.	52

FIGURE	PAGE
B.4. Perspective Transformation T_2 of Segment AB.	54
B.5. Perspective Transformation T_3 of Segment AB.	55
B.6. Perspective Transformation T_4 of Segment AB.	56
B.7. Device Coordinate, XD and YD, Calculation from Scene Coordinate, (X, Y, Z).	57

SECTION I

INTRODUCTION

A. Problem Definition

The computer simulation of topological features has a wide range of applications, including water level simulation and stereo pair generation. To produce an image, the first step is to acquire a database of the surface to be plotted. The U.S. Geological Survey is presently preparing a digital data base of the United States. This digital elevation data consists of arrays of elevations at regularly spaced intervals with each data tape being equivalent in area to a conventional 7.5-minute quadrangle map. This data base forms the spatial and elevational information for this thesis on the simulation of topography. The simulation process entails the transformation of the Digital Elevation Model (DEM) data into surface equations (data base primitives) in order to approximate elevations and surface tangents at locations not contained in the data tape. A hidden surface and shading algorithm is then applied to the data base primitives in order to present the desired perspective scene when viewed from specified positions.

B. Previous Research

A classical tradeoff exists between computer efficiency and data accuracy. Data accuracy as used in this thesis refers to the visual realism of the image. The critical factor in the

overall computer efficiency is the selection of the data base primitive. Popular primitives used in scene creation include points, planar surfaces, quadric surfaces, and curved surfaces. A difference in these primitives is the computation time required to calculate an arbitrary point (x, y, z) on the primitive.

Points require no calculation since they are the data that is to be plotted. To shade each point with the proper intensity, the corresponding normal of each point must be stored. The number of points and corresponding normals that must be saved to create a realistic image is extremely large. To date, no attempt has been made to simulate terrain by using the point method. However, other applications using the point method have been successfully demonstrated. For example, Reeves [7] used the point method to model grass and fire. The immense size of the data base needed to maintain the scene primitive, makes the point method inappropriate for the present application.

Planar faces offer a realistic means to store the characteristic of a topological area. In this application, each planar surface could be stored as a set of four points, easily extractable from the DEM. An arbitrary scene coordinate on a surface could be quickly calculated by applying simple linear geometry. Many flight simulator packages use this data primitive because the linear approach allows real time perspective variations. The disadvantage is the lack of realism associated with the scene, since most planar faces are clearly discernible from neighboring faces.

Curve surfaces, are by their nature, the most accurate data base primitives. However, the employment of these primitives results in increased computational time to solve for the arbitrary points on the surface. Curve surfaces used to describe three-dimensional shapes can be formulated in many different ways. Marshall [5], Brown [2], Pratt [6], Barsky [1] and Foley [3]. G. Gardener [4] used quadric surfaces combined with planar surfaces to reproduce topological forms. The advantage of the quadric is the minimal time needed to solve for unknown points. However, the ability to transform a DEM into a set of quadrics and surfaces has yet to be proven.

Other surface models include the Bezier and B-spline models. However, due to the great number of DEM points which exist for an arbitrary topological scene, large computational times are required to employ either the Bezier or B-spline Method.

C. Thesis Goal

The primary goal of this thesis is to achieve visual realism of the DEM data base. Therefore, a curved surface data base primitive will be employed. To maximize the efficiency in calculating the necessary parameters, the Hermite curve will be used to construct the curved surface or patch. The details describing the mathematical definition of a Hermite curve are presented in section II. Section III describes the interpolation of a "source" Hermite curve, along two "bounding" Hermite curves, into a "destination"

Hermite curve, thereby constructing a single surface patch. Section IV explains how the DEM data base is transformed into these patches while section V deals with the description of the hidden surface and shading algorithm. The results are compared to actual photographs in section VI, and section VII outlines some possible applications.

SECTION II

HERMITE CUBIC CURVE

A. Curve Point Calculation

Parametric equations have advantages in curve and surface design. A parametric cubic curve is one for which x , y , and z are each represented as a third order polynomial of some parameter t . Hence,

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x \quad (\text{II.1})$$

$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y$$

$$z(t) = a_z t^3 + b_z t^2 + c_z t + d_z$$

As t increases from 0 to 1, coordinates defining the curve are calculated. However, 12 coefficients first need to be obtained in order to calculate these curve coordinates. (Derivation of equations II.2-II.18, follow Foley [3] with a slight change in nomenclature introduced.)

The Hermite form of a parametric cubic curve derives these 12 coefficients by utilizing the positions and tangents at the curve's endpoints (Figure II.1).

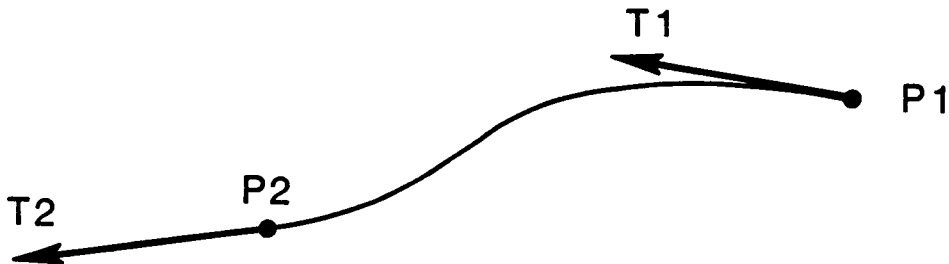


Figure II.1. Hermite Cubic Curve.

Due to the symmetry in the cubic equations of $x(t)$, $y(t)$, and $z(t)$ only the derivation of $x(t)$ will be presented. Therefore, it is necessary to solve for a_x , b_x , c_x , and d_x from equation II.1 subject to the conditions:

$$x(0) = P1_x, x(1) = P2_x, x'(0) = T1_x, x'(1) = T2_x \quad (II.2)$$

The subscript x refers to the x -component of the point and tangent vectors.

Rewriting $x(t)$, we have:

$$x(t) = [t^3 \ t^2 \ t \ 1] \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}_x \quad (II.3)$$

$$x(t) = [t^3 \ t^2 \ t \ 1] C_x \quad (II.4)$$

$$x(t) = T C_x \quad (II.5)$$

Here T is the row vector of powers of t , while C_x is the column vector of coefficients of $x(t)$.

Now express the conditions of equation II.2 by using equation II.4:

$$x(0) = P1_x = [0 \ 0 \ 0 \ 1] C_x \quad (II.6)$$

$$x(1) = P2_x = [1 \ 1 \ 1 \ 1] C_x \quad (II.7)$$

To continue with the tangent vector conditions, first differentiate equation II.4 with respect to t , obtaining

$$\mathbf{x}'(t) = [3t^2 \ 2t \ 1 \ 0] \mathbf{C}_x \quad (\text{II.8})$$

Then

$$\mathbf{x}'(0) = \mathbf{T1}_x = [0 \ 0 \ 1 \ 0] \mathbf{C}_x \quad (\text{II.9})$$

$$\mathbf{x}'(1) = \mathbf{T2}_x = [3 \ 2 \ 1 \ 0] \mathbf{C}_x \quad (\text{II.10})$$

The four conditions (II.6), (II.7), (II.9), and (II.10) can be gathered together into a single matrix equation:

$$\begin{bmatrix} \mathbf{P1} \\ \mathbf{P2} \\ \mathbf{T1} \\ \mathbf{T2} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} \mathbf{C}_x \quad (\text{II.11})$$

Inverting the 4×4 matrix achieves the objective of solving for $\mathbf{C}_x$, giving

$$\mathbf{C}_x = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{P1} \\ \mathbf{P2} \\ \mathbf{T1} \\ \mathbf{T2} \end{bmatrix}_x = \mathbf{M}_h \mathbf{G}_{hx} \quad (\text{II.12})$$

Here $\mathbf{M}_h$ is the Hermite matrix and $\mathbf{G}_h$ is the Hermite geometry vector. Apply this result to equation II.5 to obtain,

$$\mathbf{x}(t) = \mathbf{T} \mathbf{M}_h \mathbf{G}_{hx} \quad (\text{II.13})$$

and by symmetry,

$$y(t) = TM_h G_{hy} \quad (II.14)$$

$$z(t) = TM_h G_{hz} \quad (II.15)$$

Given **P1**, **P2**, **T1**, and **T2**, evaluate $x(t)$, $y(t)$ and $z(t)$ for $0 \leq t \leq 1$ and find all points on the cubic curve from **P1** to **P2** with starting tangent vector **T1** and ending tangent vector **T2**.

Taking the product TM_h produces:

(II.16)

$$TM_h = [(2t^3 - 3t^2 + 1) (-2t^3 + 3t^2) (t^3 - 2t^2 + t) (t^3 - t^2)]$$

Postmultiplying this by G_{hx} yields:

(II.17)

$$x(t) = TM_h G_{hx} = P1_x(2t^3 - 3t^2 + 1) + P2_x(-2t^3 + 3t^2) + T1_x(t^3 - 2t^2 + t) + T2_x(t^3 - t^2)$$

Equation II.17 is used to calculate the x-coordinate of the curve as a function of t . Similar equations exist for $y(t)$ and $z(t)$ by substituting the subscript x for y and z , respectively. These equations can be written collectively as

(II.18)

$$P(t) = P1(2t^3 - 3t^2 + 1) + P2(-2t^3 + 3t^2) + T1(t^3 - 2t^2 + t) + T2(t^3 - t^2)$$

B. Point Normal Calculation

In order to determine the shading intensity at points along the curve, an equation must be derived which calculates the curve tangent as a function of t . Thus differentiating equation II.17 with respect to t yields:

$$\begin{aligned} x'(t) = & P1_x(6t^2 - 6t) + P2_x(-6t^2 + 6t) \\ & + T1_x(3t^2 - 4t + 1) + T2_x(3t^2 - 2t) \end{aligned} \quad (II.19)$$

As before, similar equations exist for $y'(t)$ and $z'(t)$ by substituting the subscript x for y and z , respectively. These equations can be written collectively as

$$\begin{aligned} T(t) = & P1(6t^2 - 6t) + P2(-6t^2 + 6t) \\ & + T1(3t^2 - 4t + 1) + T2(3t^2 - 2t) \end{aligned} \quad (II.20)$$

Algorithm Efficiency

The eight functions of t in equation II.17 and II.19 are identical for $y(t)$ and $z(t)$. Therefore, it is logical to build a lookup table for these functions in order to reduce the calculation of the location and tangent of each point along a cubic curve. In other words, the number of mathematical operations can be greatly reduced in solving for points and tangents along the hermite cubic curve if a lookup table containing the functions is utilized. Table II.1 illustrates an example lookup table with a stepsize of 0.1.

TABLE II.1

EXAMPLE LOOKUP TABLE

Function of t	0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1
$2t^3 - 3t^2 + 1$	1	0.97	0.89	0.78	0.65	0.50	0.35	0.22	0.10	0.03	0
$-2t^3 + 3t^2$	0	0.03	0.10	0.22	0.35	0.50	0.65	0.78	0.89	0.97	1
$t^3 - 2t^2 + t$	0	0.08	0.13	0.15	0.14	0.13	0.09	0.06	0.03	0.01	0
$t^3 - t^2$	0	-0.01	-0.03	-0.06	-0.09	-0.13	-0.14	-0.15	-0.13	-0.08	0
$6t^2 - 6t$	0	-0.54	-0.96	-1.26	-1.44	-0.50	-1.44	-1.26	-0.96	-0.54	0
$-6t^2 + 6t$	0	0.54	0.96	1.26	1.44	1.50	1.44	1.26	0.96	0.54	0
$3t^2 - 4t + 1$	1	0.63	0.32	0.07	-0.12	-0.25	-0.32	-0.33	-0.28	-0.17	0
$3t^2 - 2t$	0	-0.17	-0.28	-0.33	-0.32	-0.25	-0.12	0.07	0.32	0.63	1

SECTION III

SURFACE PATCH DEFINITION

A. Patch Point Calculation

The patch is the sole building block of the three-dimensional scene. A surface patch, in this presentation, is constructed by interpolating from a "source" Hermite curve along two "bounding" Hermite curves into a "destination" Hermite curve. That is, to calculate all the scene coordinates of a patch, the first step is to determine the scene coordinates of the "source" curve at $s = 0$ (Figure III.1). Using **P1**, **P2**, **T1**, and **T2** as the conditions for equation II.18, we can evaluate for $0 \leq t \leq 1$ and therefore find all the points on this curve from **P1** to **P2**, with starting tangent vector **T1** and ending tangent vector **T2**. After this, take one s -step along the "bounding" curve at $t = 0$ using **P1**, **P4**, **S1**, and **S4** to evaluate for the second scene coordinate of this curve (Note: the first scene coordinate of the $t = 0$ curve is **P1**) and call it **a** (Figure III.2). Use this same logic to take one step along the other "bounding" curve at $t = 1$ using curve conditions **P2**, **P3**, **S2**, and **S3**, call this scene coordinate **b**. At this time, the two endpoints, **a** and **b**, of the next curve to be evaluated in the t -direction are known. However, the necessary tangent vectors for this curve are not known as they were for the initial $s = 0$ curve. Therefore, it is necessary to interpolate between

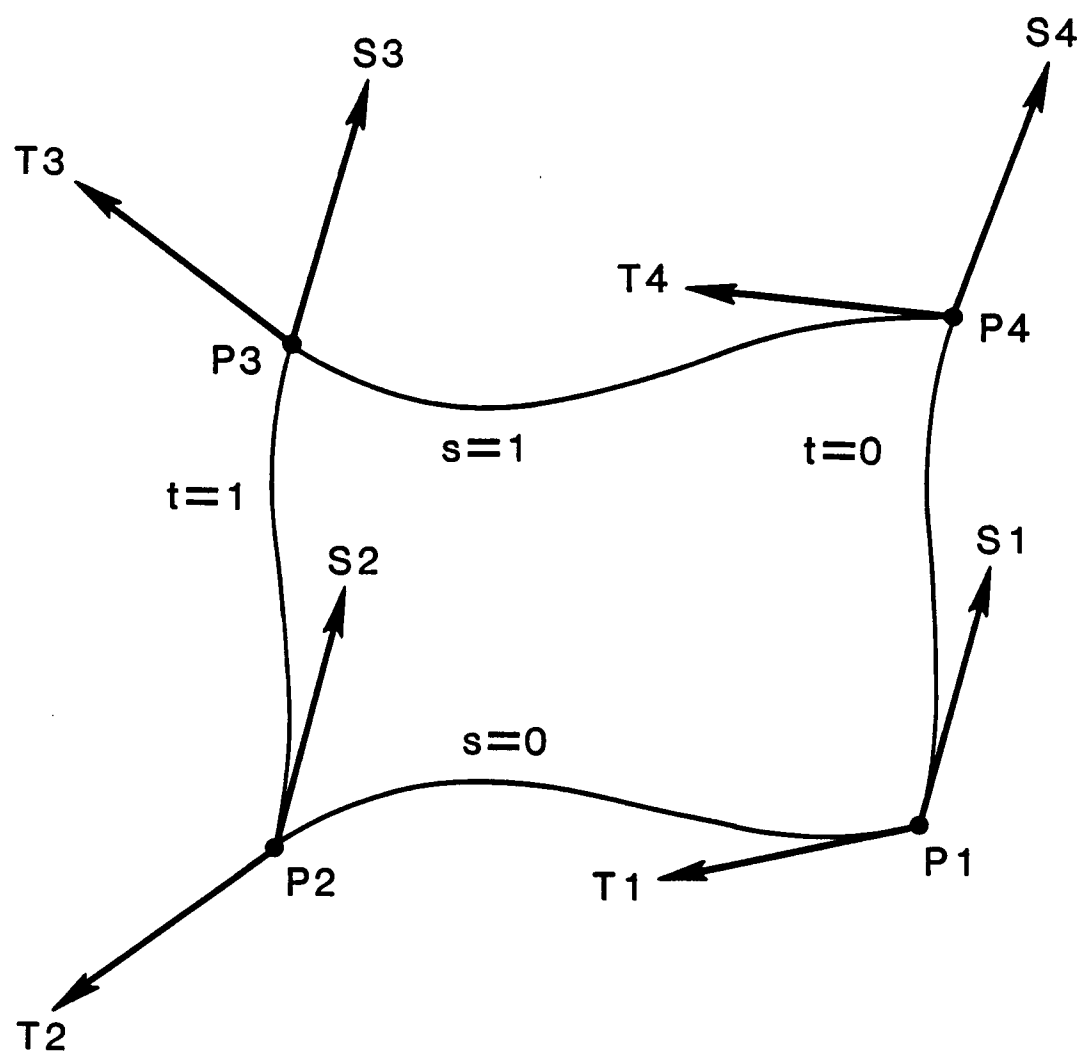


Figure III.1. Surface Patch.

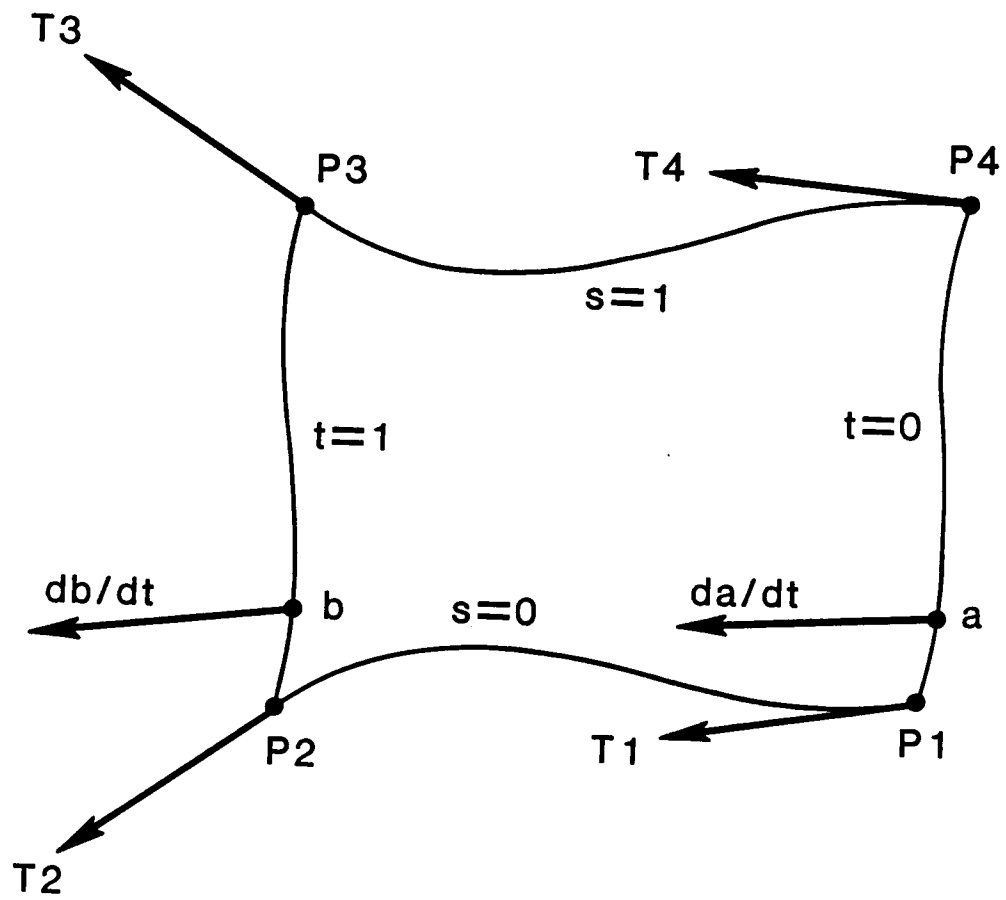


Figure III.2. Calculation of da/dt and db/dt .

the $T1$ and $T4$ tangent vectors to create an approximation of the t -tangent vector at scene coordinate a . The equation for the t -tangent vector, as a function of s , along the $t = 0$ curve is

$$da/dt = (T4 - T1) * s + T1 \quad (III.1)$$

By symmetry, the equation for the t -tangent vector, as a function of s , along the $t = 1$ curve is

$$db/dt = (T3 - T2) * s + T2 \quad (III.2)$$

The necessary information to solve the cubic curve from a to b with starting tangent vector da/dt and ending tangent vector db/dt as the conditions for equation II.18 is now known. This process is repeated for each s -step ($0 \leq s \leq 1$). The end result is the complete evaluation of all scene coordinates originating from this patch.

B. Point Normal Calculation

To determine the shading value at each scene coordinate the normal vector at each coordinate must be determined. To do this, calculate the cross product of dp/dt and dp/ds , where dp/dt and dp/ds are the tangent vectors in the t and s direction, respectively, of an arbitrary point on the patch. Calculating dp/dt is straight forward from equation II.20 using a , da/dt , b and db/dt for the needed conditions. To calculate dp/ds first

determine da/ds and then db/ds by using equation II.20. After this is done, approximate dp/ds by interpolating between da/ds and db/ds . The equation of the tangent vector, in the s -direction, as a function of t for an arbitrary point on the patch is

$$dp/ds = (db/ds - da/ds) * t + da/ds \quad (III.3)$$

At this time, all the equations needed to calculate the necessary attributes of an arbitrary point on a patch, namely, the three-dimensional location of that point and the surface normal at that point, are known. In summary, 36 conditions are needed to describe a surface patch in this presentation, three coordinates for each of the $P1$, $P2$, $P3$, $P4$, $T1$, $T2$, $T3$, $T4$, $S1$, $S2$, $S3$, and $S4$ vectors.

SECTION IV
TRANSFORMATION OF DIGITAL ELEVATION MODEL
TO SURFACE PATCHES

The DEM data tapes consist of a regular array of elevations spaced 30-meters apart as referenced in the Universal Transverse Mercator Coordinate system. The elevations are given in meters above sea level. Each data tape consists of I-profiles or x-values, ordered from west to east, and each profile consists of J-data elements or y-values ordered from south to north. The elevation at a DEM point on the surface can thus be defined as $x_i y_j$ where i and j correspond to the respective profile and data element in the DEM. (The information contained in a DEM is outlined in Appendix A.) Because the data is regularly spaced, it seems appropriate to transform it into a set of surface patches with each patch being 30-meters square. In order to calculate the 36 conditions associated with each patch, as described in section III, it is first necessary to read the elevation data into a two-dimensional array called SURFACE with dimensions $I \times J$. Another step, that will become apparent later, is to multiply all data values by $1/30$ as each is stored in the array SURFACE. Figures IV.1 and IV.2 illustrate the southwest region of an example DEM and its corresponding SURFACE array. In Figure IV.1 we define a three-dimensional scene coordinate space (x, y, z) , with the

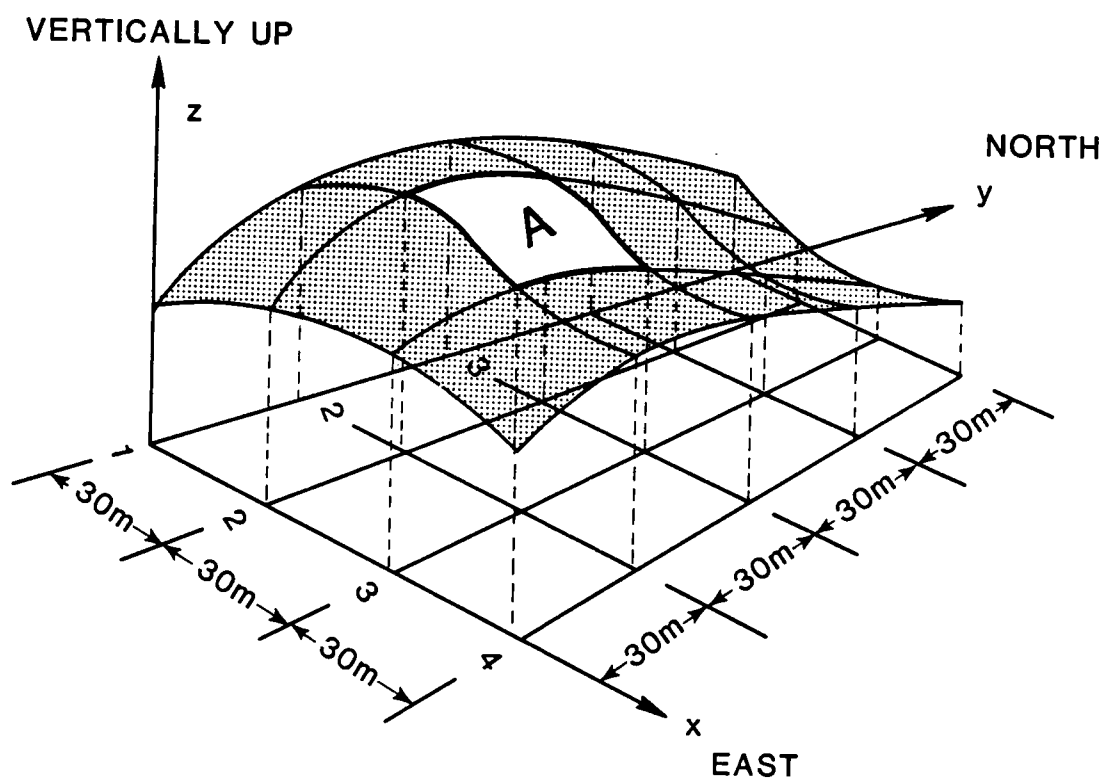


Figure IV.1. Southwest Region of an Example DEM.

NORTH

y

5	$\frac{X1Y5}{30}$	$\frac{X2Y5}{30}$	$\frac{X3Y5}{30}$	$\frac{X4Y5}{30}$	
4	$\frac{X1Y4}{30}$	$\frac{X2Y4}{30}$	$\frac{X3Y4}{30}$	$\frac{X4Y4}{30}$	-----
3	$\frac{X1Y3}{30}$	$\frac{X2Y3}{30}$	$\frac{X3Y3}{30}$	$\frac{X4Y3}{30}$	-----
2	$\frac{X1Y2}{30}$	$\frac{X2Y2}{30}$	$\frac{X3Y2}{30}$	$\frac{X4Y2}{30}$	-----
1	$\frac{X1Y1}{30}$	$\frac{X2Y1}{30}$	$\frac{X3Y1}{30}$	$\frac{X4Y1}{30}$	-----
	1	2	3	4	----- x EAST

Figure IV.2. SURFACE Array Containing Example DEM.

x-axis pointing east, the y-axis pointing north and the z-axis pointing vertically up. Because each elevation was multiplied by 1/30, the array SURFACE is actually the three-dimensional data base characterizing the terrain. For example, the elevation at profile-2, data element-3 (i.e., SURFACE (2,3)) directly corresponds to the scene coordinate (x, y, z) where $x = 2$, $y = 3$, and $z = \text{SURFACE (2,3)}$. The overall advantage of this scheme allows for the array SURFACE to be logically processed (i.e., the calculation and saving of the 36 conditions defining each patch.)

Algorithm Efficiency

The following strategy decreases the computational requirements and data storage needs in converting the DEM data to surface patches. Because the data is spatial in nature, there are certain patch conditions of the complete set of 36 that are constant from one surface patch to the next. Table IV.1 contains the conditions defining patch A of Figure IV.1 and Figure IV.3.

Notice that the x-coordinate of all tangent vectors in the t-direction are always 1. These and other constants are shown as follows:

$$\mathbf{T1_x = T2_x = T3_x = T4_x = 1} \quad (\text{IV.1})$$

$$\mathbf{T1_y = T2_y = T3_y = T4_y = 0}$$

$$\mathbf{S1_x = S2_x = S3_x = S4_x = 0}$$

$$\mathbf{S1_y = S2_y = S3_y = S4_y = 1}$$

TABLE IV.1
CONDITIONS DEFINING PATCH A

Vector	X	Y	Z
P1	3	2	SURFACE(3,2)
T1	1	0	$\frac{\text{SURFACE}(2,2) - \text{SURFACE}(4,2)}{2}$
S1	0	1	$\frac{\text{SURFACE}(3,3) - \text{SURFACE}(3,1)}{2}$
P2	2	2	SURFACE(2,2)
T2	1	0	$\frac{\text{SURFACE}(1,2) - \text{SURFACE}(3,2)}{2}$
S2	0	1	$\frac{\text{SURFACE}(2,3) - \text{SURFACE}(2,1)}{2}$
P3	2	3	SURFACE(2,3)
T3	1	0	$\frac{\text{SURFACE}(1,3) - \text{SURFACE}(3,3)}{2}$
S3	0	1	$\frac{\text{SURFACE}(2,4) - \text{SURFACE}(2,2)}{2}$
P4	3	3	SURFACE(3,3)
T4	1	0	$\frac{\text{SURFACE}(2,3) - \text{SURFACE}(4,3)}{2}$
S4	0	1	$\frac{\text{SURFACE}(3,4) - \text{SURFACE}(3,2)}{2}$

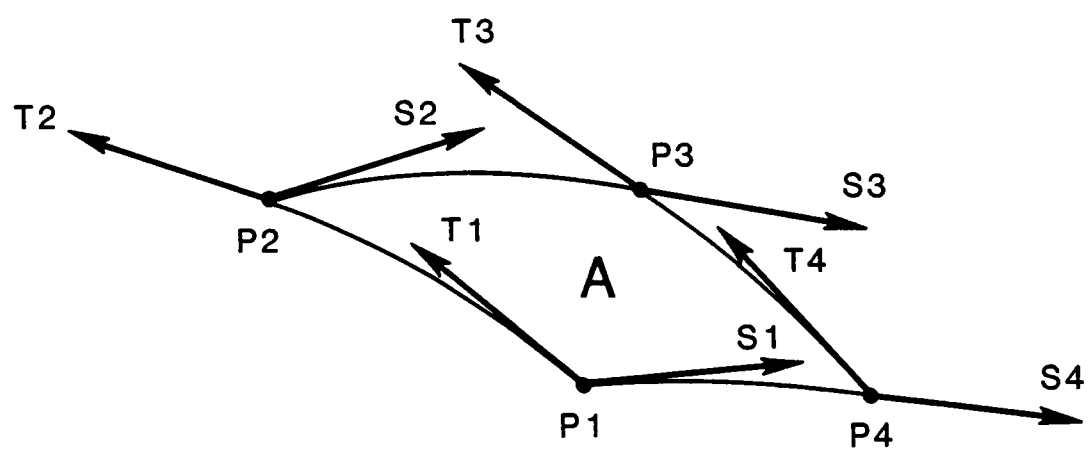


Figure IV.3. Surface Patch A of Example DEM.

Also notice that the x and y coordinate of the initial corner of each patch (**P1** in this case) is always in the same relative position with respect to the x and y coordinates of **P2**, **P3**, and **P4**. This fact leads to the following equalities:

(IV.2)

$$\mathbf{P2_x = P1_x - 1}$$

$$\mathbf{P2_y = P1_y}$$

$$\mathbf{P3_x = P1_x - 1}$$

$$\mathbf{P3_y = P1_y + 1}$$

$$\mathbf{P4_x = P1_x}$$

$$\mathbf{P4_y = P1_y + 1}$$

This information allows the calculation of **P2_x**, **P2_y**, **P3_x**, **P3_y**, **P4_x**, and **P4_y** in terms of **P1_x** and **P1_y**. In summary, the equations of IV.1 and IV.2 result in the storing of only 14 of the 36 conditions needed to define a surface patch. A very important point to mention in closing is that the processing time needed for the data transformation (from DEM to surface patches) is not associated with the time it takes to perform perspective variations on a set of these surface patches. The algorithm to do this is described in the next section.

SECTION V

HIDDEN SURFACE AND SHADING ALGORITHM

This algorithm produces a realistic presentation of surface patches. Each surface patch is separately processed by this algorithm. The "processing" refers to the set of sequential steps that each surface patch undergoes. The processing steps are as follows:

- (1) A coordinate location point P, on a patch is calculated as described in section III.
- (2) The three-dimensional point P is projected to the two-dimensional view plane using the transformation matrix explained in Appendix B. This transformation will give the X_DEVICE and Y_DEVICE coordinates of point P. This matrix also gives the distance D which P is away from the view plane.
- (3) To keep track of the closer points of the scene, an array called SCREEN_BUFFER is implemented to take care of the bookkeeping (Figure V.1). After the X_DEVICE, Y_DEVICE and D are calculated, two possibilities exist.

Case (1): If D was larger than SCREEN_BUFFER (X_DEVICE, Y_DEVICE, DEPTH) then the point that was previously mapped into X_DEVICE, Y_DEVICE is actually closer than P,

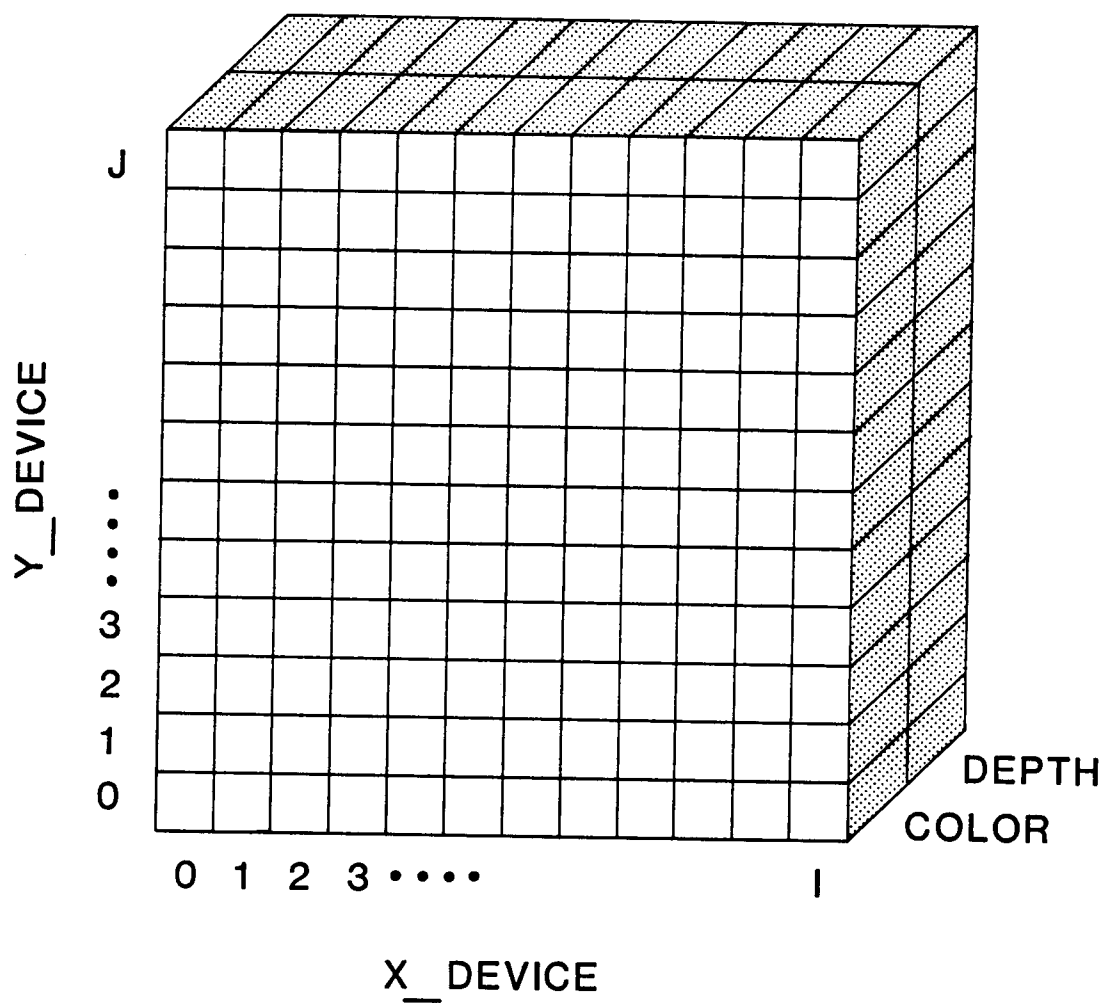


Figure V.1. SCREEN_BUFFER Array Employed to Keep Track of Closer Scene Points.

therefore no further processing of point P is necessary and step (1) can be initiated again, that is, find another point on the patch and process as before.

Case (2): If D was smaller than SCREEN_BUFFER (X_DEVICE, Y_DEVICE, DEPTH), then point P is visible and the previous intensity-value and D-value of SCREEN_BUFFER at X_DEVICE, Y_DEVICE is no longer valid. That is, it is necessary to update the intensity of pixel X_DEVICE, Y_DEVICE and store the new value of D to reflect the closeness of point P to the view plane. To calculate the intensity of point P we use equation II.20 to first determine the tangents in both the s and t directions, as discussed in section III. After these two vectors have been calculated cross the s-vector into the t-vector to determine the surface normal at point P. To calculate the shading intensity or tone of P the sun vector must be defined. This vector and the normal of point P are illustrated in Figure V.2. In nature, there are many factors that determine the final appearance of an object. (For example, the properties of the surface of the object, the atmospheric conditions, the relative position of the sun, and the reflection of light off other objects.) This presentation will calculate the amount of light as a function of the angle, θ , between the normal of

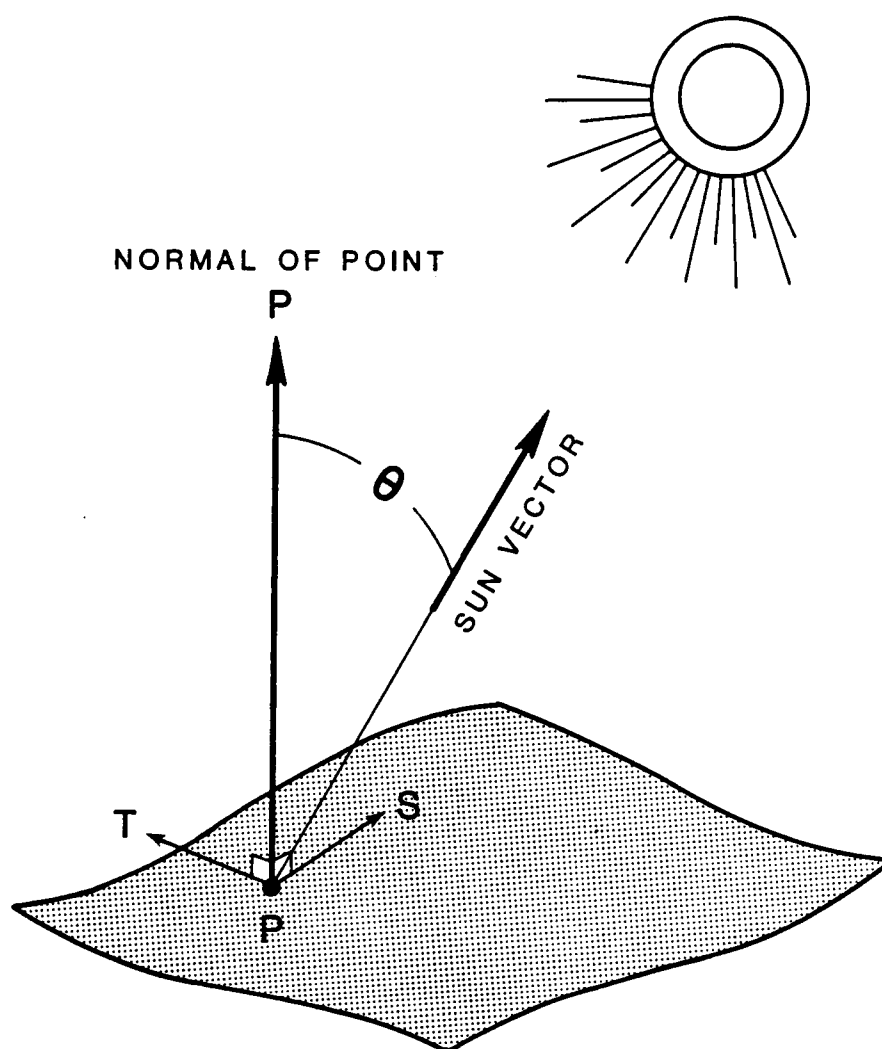


Figure V.2. Point Normal Vector and Sun Vector.

point P and the sun vector. This angle is shown in Figure V.2. Lambert's cosine law relates the amount of reflected light to the cosine of the angle between the normal to the surface and the direction of the point light source (sun vector). Also, an ambient light component will be added to all surfaces. This component is produced from the scattering of light in the atmosphere and the reflection of light off other objects in the scene. The equation to calculate the intensity of a given point P is thus:

$$I_p = I_L \cos \theta + I_a$$

Where I_p refers to the intensity of point p, I_L refers to the intensity of light source and I_a refers to the ambient light intensity component. If θ is greater than 90° , only the ambient term is used. (In other words, if the angle between the surface normal and sun vector is greater than 90° then $I_p = I_a$.) This value is placed in the SCREEN_BUFFER (X_DEVICE, Y_DEVICE, COLOR) and the processing of point P is concluded. Step (1) is now initiated again for another point on the patch. When all the points on this patch are calculated, the next patch is processed, and so on until the entire scene has been processed. The overall effect of this scheme forces closer points to remain in the SCREEN_BUFFER array.

SECTION VI

DISCUSSION AND RESULTS

A. Discussion

As outlined in section I there are numerous ways of creating a three-dimensional topological scene. The trade-off is visual realism (data accuracy) versus computational efficiency. This section will discuss, with respect to visual realism and associated efficiency, the characteristics of the planar face, Bezier surface, B-spline surface, and Hermite interpolating technique.

The planar face is the most efficient data base primitive. The normal and therefore intensity of each planar face can be quickly calculated. The points located within each planar face can also be calculated using basic linear geometry. A hidden surface algorithm, used to process each planar face, can be developed based on linear geometry to quickly draw the perspective scene. However, the drawback is in the realism of the scene because the boundary between two faces with slightly different intensities will be visible. This is known as the Mach Band effect. In mathematical terms there is no first derivative continuity across surface boundaries. The first derivative or slope from one planar face instantly changes in crossing into a neighboring face. In other words, there is no blending of the slopes in going from one planar face to another. Therefore, face boundaries are noticeable in

scenes created with planar faces. If enough planar faces describe a given object so that each face maps into no more than one pixel, then this technique produces the continuous shading quality characteristic of a curved surface.

The Bezier surface is computationally too expensive for this application. The reason for this is that each initial control point, or DEM elevation, affects all points that are calculated on a Bezier surface. This is known as global control. The advantage of a Bezier cubic surface is that the surface maintains first and second derivative continuity.

The B-spline surface is inaccurate at the DEM data control point. This is because a B-spline surface is not forced to pass through the initial control points that were used to create the surface. In the case of the DEM data, the surface representation would not necessarily pass through the initial DEM elevation values. An advantage of a B-spline surface is that it maintains local control and therefore is computationally less expensive than Bezier surfaces. In other words, the altering of a single control point, or DEM elevation, affects only the surface in the vicinity of that control point. This surface also maintains first and second derivative continuity.

The surface used in this thesis is created by the interpretation of an initial Hermite curve into a resultant Hermite curve along two bounding Hermite curves. The resulting surface patch formed by the Hermite curves has first derivative continuity and

local control. This surface also passes through all control points, or DEM elevations, thus data accuracy is maintained at each DEM elevation. The overall efficiency is better than that of the Bezier and B-spline surfaces which employ a greater number of points in the calculation of the surface equation.

A point to be mentioned about curved surfaces in general is that these data base primitives allow the resulting image to be computed to whatever level of detail the situation demands. This level of detail is controlled by the step size chosen for the calculation of points on the surface.

In this thesis the stepsize was calculated in terms of the distance a given patch was from the view plane. Thus, a three-dimensional patch, if visible, will map into a two-dimensional area on the display device. The proper number of points must be chosen so that all pixels within this two-dimensional area are turned on and shaded properly. If the number of points solved for the patch is too small, pixels within this two-dimensional area will not be updated and assigned the appropriate intensity.

B. Results

To evaluate the quality of the computer generated image and, therefore, the surface equations used to create the topological scene, the software created images were compared with photography and a contour map of the study area. The software images were all

displayed on a Vectrix-VX384 color graphic monitor and photographed with a Nikon-FM 35mm SLR camera.

Water Level Depiction

The Rockwood, TN contour map is not the same data base as the Rockwood TN DEM as they were produced differently from different data bases (see Figures VI.1, 2, 3, 4). Elevations on the contour map are depicted as irregular concentric boundaries. Each contour line depicts a constant elevation. The DEM data base, however, is a spatial array of elevations (Figure IV.1, page 17). Another difference is the date that each data base was generated. The contour map was constructed in 1969 while the DEM data was constructed in 1982. However, comparison with the contour map remains a viable method of measuring overall elevation model accuracy especially at waterland boundaries. To accomplish such a comparison, the simulated model was set to show a 59-foot (18.0-meter) increase in the water level of the Tennessee River in the section of the Rockwood quadrangle containing Thief Neck Island. The contour map was adjusted to also depict this water level increase. Figure VI.1 illustrates the water level at the normal, 741-ft (225.9-m) elevation, while Figure VI.2 illustrates a water level of 800-ft (243.8-m). The illustration was constructed by whiting-out contour lines less than 800-ft (243.8-m). Figure VI.3 illustrates the computer graphic image of the same region with the water-level specified at 741-ft (225.9-m), while Figure VI.4 illustrates the computer graphic image

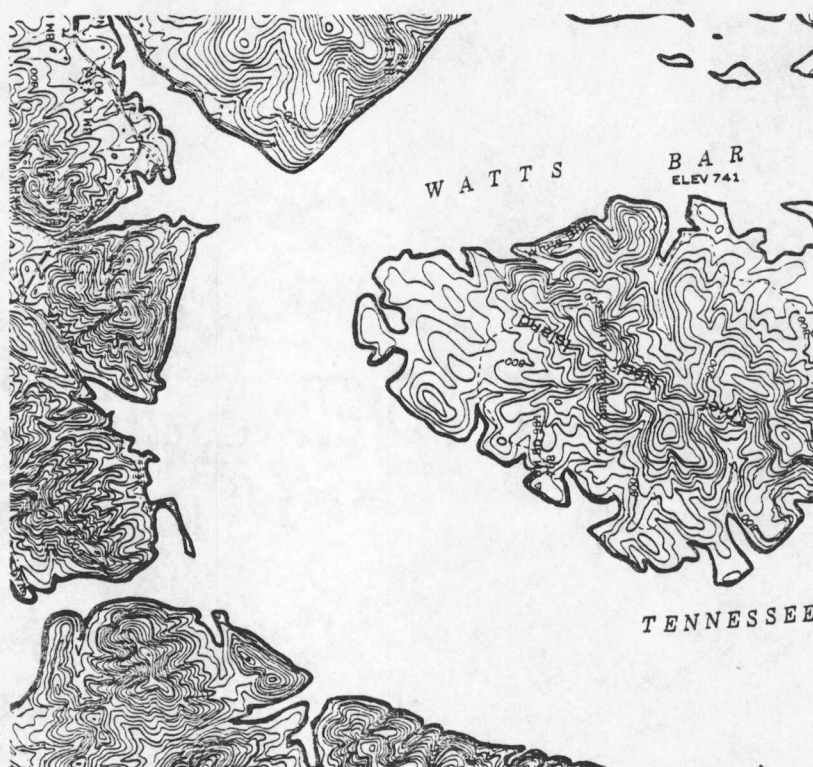


Figure VI.1. Contour Map Illustration of Thief Neck Island with Water Level Equal to 741-ft (225.9-m).

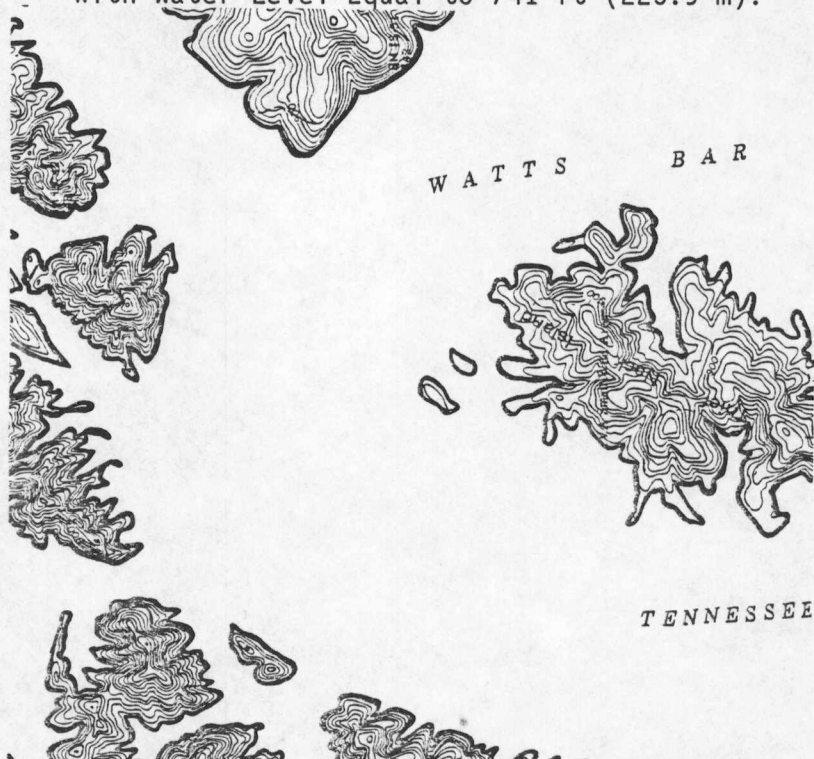


Figure VI.2. Contour Map Illustration of Thief Neck Island with Water Level Equal to 800-ft (293.8-m).

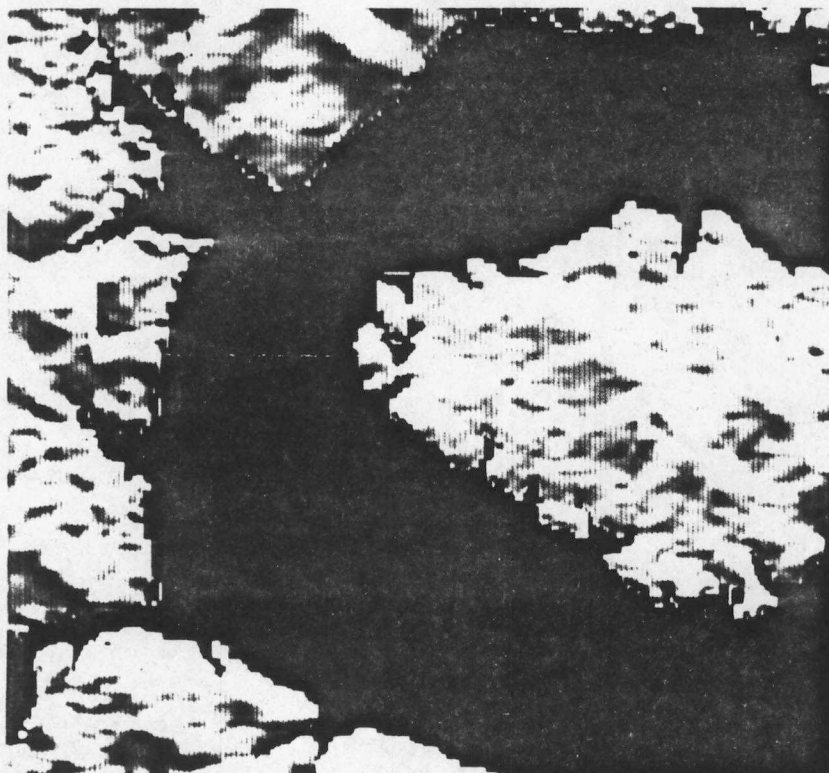


Figure VI.3. Computer Graphic Image of Thief Neck Island with Water Level Set at 741-ft (225.9-m).

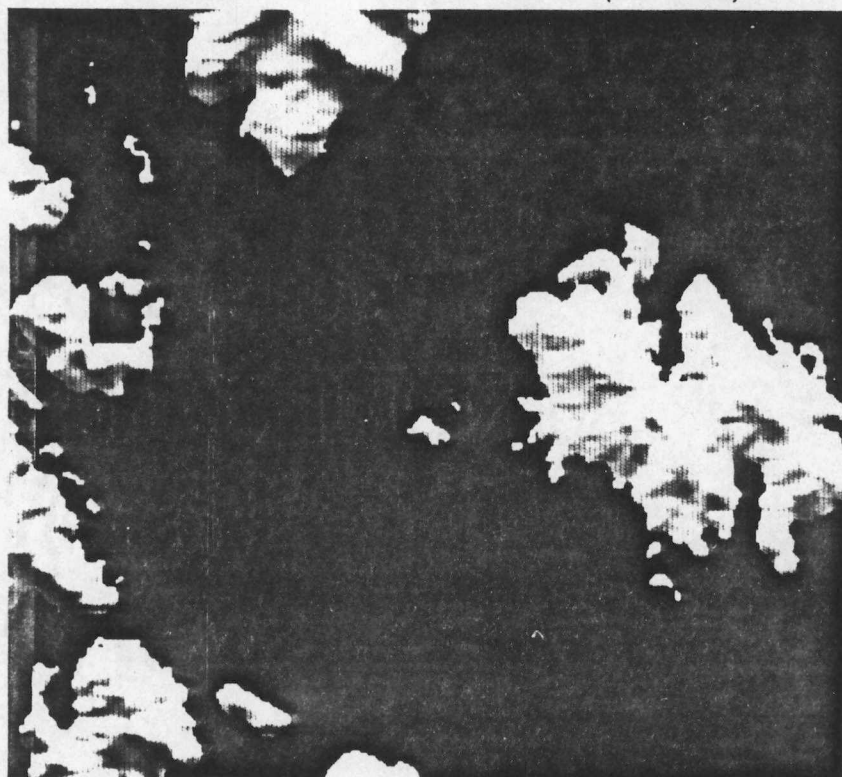


Figure VI.4. Computer Graphic Image of Thief Neck Island with Water Level Set at 800-ft (293.8-m).

with the water-level specified at 800-ft (243.8-m). Comparisons of the contour illustrations, Figures VI.1 and VI.2, with the corresponding graphic images, Figures VI.3 and VI.4, verify the usefulness of this application.

Stereo Pair Generation

In this illustration a stereo pair of photographs of the Thief Neck Island section is compared to a pair of computer generated images. The two generated images each have a slightly different view reference point, therefore, allowing the creation of a stereogram of Thief Neck Island. The original photographs were taken at an elevation of 40,000-ft using infrared film (Figure VI.5). Figure VI.6 illustrates the computer generated stereogram of the same section of Thief Neck Island. The advantage of the actual photographic stereogram is the amount of detail and resolution present. However, a computer generated stereogram utilizing the DEM data can be created for any position in the three-dimensional scene and is not limited to only the viewpoint of the original photography.

Ground Perspective Image

To illustrate ground level potential uses, a computer generated image with a ground origin is compared to a photograph taken from the same origin point. The photograph was taken at a ground point within the Rockwood, TN quadrangle area looking North 20° East. The image is of Thief Neck Island (Figure VI.7). Figure VI.8 is the computer generated image and has the same view reference

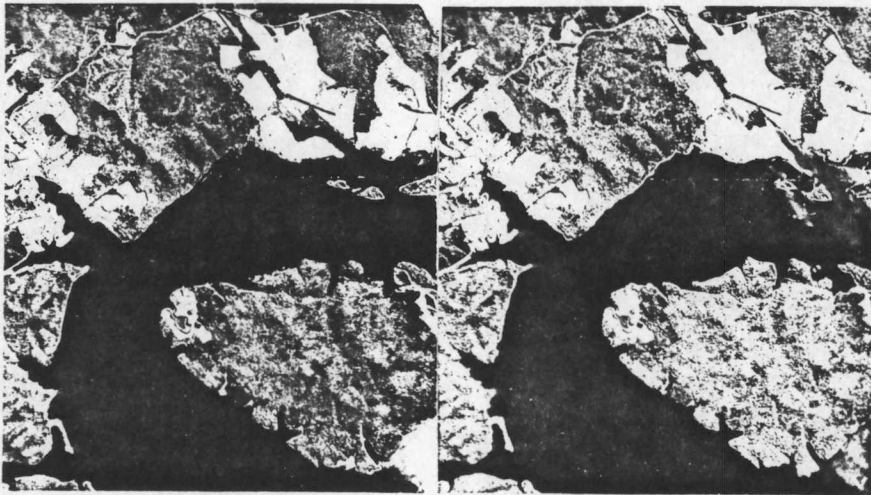


Figure VI.5. Photographic Stereogram of Thief Neck Island.

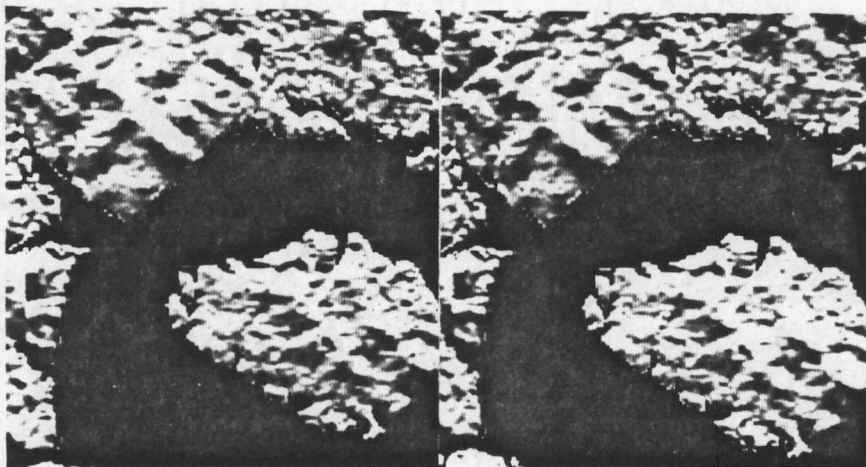


Figure VI.6. Computer Generated Stereogram of Thief Neck Island.



Figure VI.7. Photographic Ground Point Origin
of Thief Neck Island.

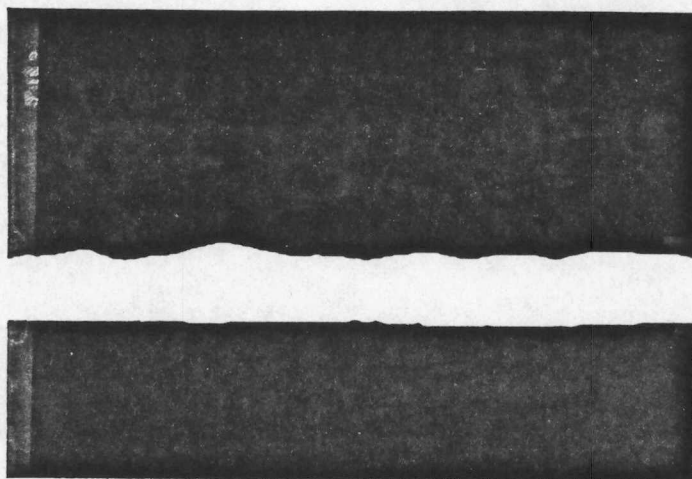


Figure VI.8. Computer Generated Ground Point Origin
of Thief Neck Island.

point and look direction that was used to photograph Figure VI.7. Again, the advantage of the actual photograph is resolution. However, the time to generate such a graphic image will usually be only a fraction of the time it takes to obtain on-the-ground photography. Also, perspective views can be generated from any location in the scene with any look direction. Such a series of corresponding photographs would be impractical, if not impossible, to take.

SECTION VII

APPLICATIONS

The ability to remotely generate three-dimensional scenes of terrain surfaces based on measured elevational data has extensive applications. These applications can be broken down into two categories, (A) scientific modeling of natural or manmade events, (B) education and training. The number of DEM tapes needed for the data base varies from application to application.

A. Scientific Modeling of Natural or Manmade Events

This application gives researchers a valuable tool with which to present their findings. This thesis presented in section VI the simulation of a selected water level as an example application. Possible applications include the following:

- 1) Water level presentations--The ability to realistically present the coverage of a natural or manmade flood level is of assistance to environmental engineers, civil engineers, public health officials, property appraisers, and agency officials responsible for this decision making information. This application could also be used to pictorially inform non-technical individuals on the effects of flooding.

- 2) Snow level presentations--The National Weather Service (NWS) has steadily increased its accuracy of weather forecasting.

The NWS tries to predict the extent of snowfall as a function of elevation and atmospheric conditions. This data could be used to depict snow levels in mountainous regions. The snow level presentations could then be used to predict road access, potential avalanche areas, safe off-trail skiing areas and possible crop damage regions.

3) Increased realism--The overlay of LANDSAT data onto the digital terrain model would greatly increase the realism of the image. This plot combined with the flooding and snow level predictions could not only depict the extent of such an event but the type of vegetation affected.

4) Predictions of natural resource and mineral locations-- Certain types of minerals and natural resources, including coal and oil, are found in geographical environments having characteristic shapes and forms. The ability for searchers to locate such terrain characteristics would be greatly enhanced by the three-dimensional views afforded by this system.

5) Fire spread predictions--This application could potentially be valuable in fire management, thus saving lives and natural resources. A computer program could dynamically calculate, a frame at a time, the affected area as a function of time. The algorithm could treat the fire line as a set of point sources. The propagation of each source would be a function of wind direction, vegetation type(s), atmospheric conditions, shape of terrain, and other factors related to the overall ground speed and direction of a fire. Each frame could present the newly calculated fire front. The fire spread

could be predicted at some specified time in the future. The algorithm could then present a sequential set of frames displaying the fire's speed, giving the fire control personnel an added edge in bringing the fire under control. This computer program could even be extended to aid in directing the drop of fire retardants and the airlifting of fire-fighting machinery to selected locations.

B. Remote Education and Training

This application category gives educators a means of presenting complex ideas associated with topological characteristics.

1) Flight route training--An extremely powerful training tool for military and civilian pilots not familiar with a particular flight route could be to create continuous scenes of the route. Prior to takeoff the pilot could, at a greatly accelerated rate, visually fly the route. The algorithm could highlight features such as communication towers, airports, restricted areas, high population densities, mountainous regions, and other aircraft hazards.

2) Presentation of site and transportation networks--The ability to produce a realistic scene depicting the environmental impact of proposed industrial, transportation, airport, or housing projects could be invaluable as an information base for safety engineers, and land developers.

3) Education tool for geologists and other terrain related disciplines--The instructor could quickly call up and vary the perspective view of any topological feature. Stereograms, as presented

in section VI, could be generated to increase the usefulness of this application.

4) Search and rescue--This application could give search and rescue teams an added insight into the topological characteristic of a given area, thus reducing decision making time. A typical application could be to design the algorithm to determine the starting points of each search and rescue team along with a trail route to gain access to the area. The algorithm could even calculate perspective views at specified intervals along the different search routes, thus reinforcing the proper moment of the search and rescue teams.

5) The recreation of historical or annual events--War strategy and evolution or migration of a specimen can be visually recreated thereby, depicting the event in a more exact manner.

In summary, a potential application exists whenever the topological surface of an area plays a role in an event and there exists a need to visualize this event. A realistic three-dimensional perspective image contains easily interpreted information of value to many individuals.

SECTION VIII

CONCLUSIONS

This thesis has described a means of generating realistic topological images from U.S. Geological Survey Digital Elevation Model (DEM) data. Scenes were constructed using a surface patch database created by interpolating between Hermite curves. A hidden surface and shading algorithm was designed to sequentially process each patch.

The quality or realism of the computer generated images were excellent. The ability to simulate flooding, create stereograms, and arbitrary perspective scenes of the DEM was very effective and computationally efficient.

LIST OF REFERENCES

LIST OF REFERENCES

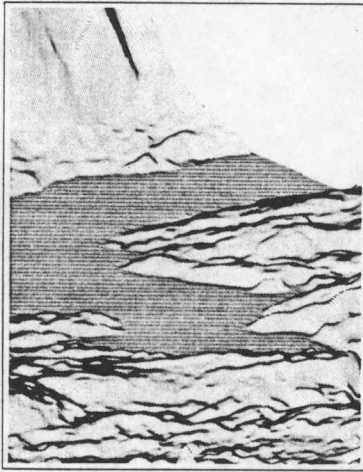
- 1) Barsky, B., "A description and evaluation of various 3-d models," IEEE CG&A (January 1984), 38-52.
- 2) Brown, C. and Ballard, D., **Computer Vision**. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.
- 3) Foley, J., and Van Dam, A., **Fundamentals of Interactive Computer Graphics**. Addison-Wesley, Reading, Massachusetts, 1982.
- 4) Gardner, G., "Simulation of Natural Scenes Using Textured Quadric Surfaces," Computer Graphics 18, 3 (July 1984), 11-20.
- 5) Marshall, R., Wilson, R., and Carlson, W., "Procedure models for generating three-dimensional terrain," Computer Graphics 14, 3 (July 1980), 154-162.
- 6) Pratt, M., and Faux, I., **Computational Geometry for Design and Manufacture**. Ellis-Horwood Limited, Chichester, England, 1979.
- 7) Reeves, W. T., "Particle systems--a technique for modeling a class of fuzzy objects," Computer Graphics 17, 3 (July 1983), 91-108.

APPENDIXES

APPENDIX A

DIGITAL ELEVATION DATA TAPES

APPENDIX A



National Mapping Program

US GeoData

Digital Elevation Models

Department of the Interior
U.S. Geological Survey
National Cartographic
Information Center (NCIC)

US GeoData Tapes Available from the U.S. Geological Survey

US GeoData tapes are computer tapes which contain cartographic data in digital form. US GeoData digital terrain tapes and digital elevation models (DEM's) are digital records of terrain elevations for ground positions at regularly spaced intervals. In digital form, terrain elevations can be valuable for regional terrain modeling, land use studies, and other scientific and engineering applications.

US GeoData tapes for terrain elevations are available in three different formats from the National Cartographic Information Center (NCIC). One format is U.S. Geological Survey 7.5-minute digital elevation model data. These data are available in 7.5- x 7.5-minute blocks which correspond to U.S. Geological Survey topographic quadrangle maps. Each DEM, like each topographic quadrangle, covers approximately 60 square miles. The ground distance between each digitized point is 30 meters.

The second format available from NCIC is Defense Mapping Agency (DMA) digital elevation model data. These data are available in 1° x 1° blocks. Two of these blocks are required to cover each 1° x 2° map (1:250,000-scale quadrangle) from which the data were produced. The ground distance between each digitized point is 3 arc-seconds.

The third format is Defense Mapping Agency digital terrain or planar data, also available in 1° x 1° blocks (produced from 1:250,000-scale maps). The ground distance between each digitized point is approximately 200 feet.

How US GeoData is Produced

U.S. Geological Survey 7.5-minute DEM's are produced from contour plates that have been digitized or from high-altitude photographs taken at an average scale of 1:78,000. The data are processed to produce a DEM with a 30-meter sampling interval. Each DEM you purchase will correspond to a complete 7.5-minute USGS topographic quadrangle map, covering approximately 60 square miles.

The Defense Mapping Agency DEM's are produced from contour plates that have been digitized and resampled at a latitude/longitude interval of 3 arc-seconds. Each 1° x 2° area is usually split into an east half and a west half to accommodate the large volume of data.

Each DEM covers a 1° x 1° block, so you will need two DEM's to cover a 1° x 2° area (1:250,000-scale quadrangle).

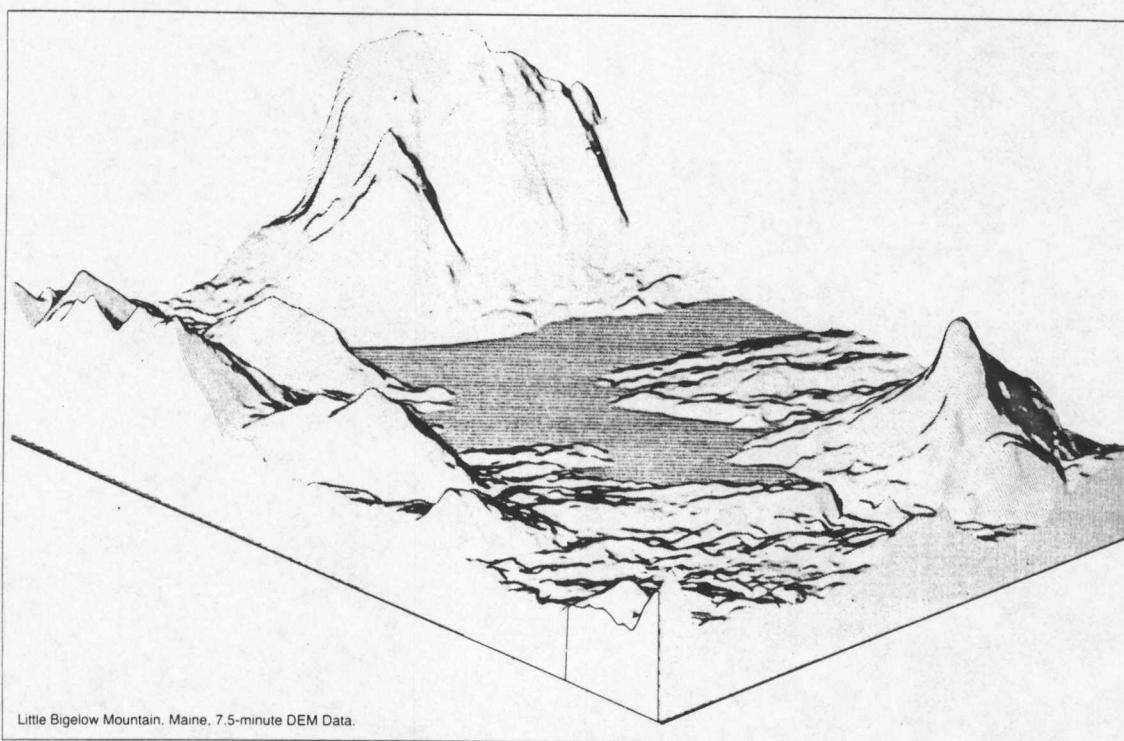
The Defense Mapping Agency planar digital terrain data are also scanned from contour plates but resampled at .01 inch on the map (about 200 feet on the ground). Each 1° x 2° area scanned is usually split into an east half and a west half to accommodate the large volume of information. Each digital terrain file covers a 1° x 1° block, so you will need two files to cover a 1° x 2° area (1:250,000-scale quadrangle).

Accuracy of US GeoData

The accuracy of US GeoData elevation models is dependent on the scale of the source materials from which the data were obtained. For example, the accuracy of DMA DEM's and planar

data is consistent with the accuracy of the contours on the 1:250,000-scale topographic maps used to produce the data. The accuracy of the USGS 7.5-minute DEM's depends on the data source, and they have been assigned to two levels of accuracy: either less than 7-meters vertical root mean square error (RMSE), or 7- to 15-meter vertical RMSE.

DMA DEM and planar data files cover the same geographical area. The main differences are in the sampling interval and the structure of the data file. The DMA DEM and the USGS 7.5-minute DEM files are identical in internal structure, but differ in sampling interval, area covered, and accuracy.



Little Bigelow Mountain, Maine, 7.5-minute DEM Data.

Applications of US GeoData

DEM files may be used in the generation of graphics displaying slope, direction of slope, and terrain profiles between designated points. They may also be used in combination with stream locations and weather data to plan forest fire control. Many non-graphic applications have also been developed, such as modeling terrain and gravity data for use in the search for

energy resources, determining the volume of proposed reservoirs, calculating the amount of material removed during strip mining, and determining landslide probability.

Ordering US GeoData

To order DEM's from NCIC, identify the quadrangles you need by latitude and longitude of the southeast corner or by the quadrangle names. The standard tape NCIC will send you is a

9-track, 1600 bpi tape unless you specifically request an 800 bpi tape. Prepayment is required by check or money order made payable to the Department of the Interior—USGS.

How to Order

For more information and to place orders contact one of the following offices:

National Headquarters

National Cartographic
Information Center
U.S. Geological Survey
507 National Center
Reston, VA 22092
703-860-6045/FTS 928-6045

Regional Offices

**Eastern Mapping Center—
NCIC**
U.S. Geological Survey
536 National Center
Reston, VA 22092
703-860-6336/FTS 928-6336

**Mid-Continent Mapping
Center—NCIC**
U.S. Geological Survey
1400 Independence Road
Rolla, MO 65401
314-341-0851/FTS 277-0851

**National Cartographic
Information Center**
U.S. Geological Survey
National Space Technology
Laboratories, Bldg. 3101
NSTL Station, MS 39529
601-688-3544/FTS 494-3544

**Rocky Mountain Mapping
Center—NCIC**
U.S. Geological Survey
Box 25046, Stop 504,
Federal Center
Denver, CO 80225
303-234-2326/FTS 234-2326

Western Mapping Center— NCIC

U.S. Geological Survey
345 Middlefield Road
Menlo Park, CA 94025
415-323-8111, ext. 2427
FTS 467-2427

**National Cartographic
Information Center**
U.S. Geological Survey
Skyline Building
218 E Street
Anchorage, AK 99501
907-271-4159/FTS 271-4159

EROS Data Center

EROS Data Center
U.S. Geological Survey
Sioux Falls, SD 57198
605-594-6151/FTS 784-7151

APPENDIX B

PERSPECTIVE PROJECTION MATRIX

APPENDIX B

In order to view the scene, each three-dimensional scene coordinate must be transformed into its corresponding two-dimensional device coordinate. The device can be a graphics monitor, addressable dot-matrix printer, bar plotter, etc. That is, any device that offers the ability to address locations on a two-dimensional surface (Figure B.1). The following set of transformations are necessary to perform this perspective projection. (An example is illustrated to aid in understanding. The task is to perform a perspective projection on the line segment from coordinate A, at (0,0,0), to coordinate B, at (1,1,1) (Figure B.2).)

- (1) T_1 Translate scene coordinates so the scene center is at the origin.--The point (XCEN, YCEN, ZCEN) is defined to be the point located in the center of the scene. The effect of this initial translation will force the scene to be "centered" on the display device (Figure B.3). (Note, the subscript n on T_n refers to the n^{th} transformation matrix needed to perform the perspective projection. In closing, a single composite 4 x 4 matrix is given.)

$$P_1 = PT_1, \text{ where } T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -XCEN & -YCEN & -ZCEN & 1 \end{bmatrix}$$

$$A_1 = (-.5, -.5, -.5)$$

$$B_1 = (.5, .5, .5)$$

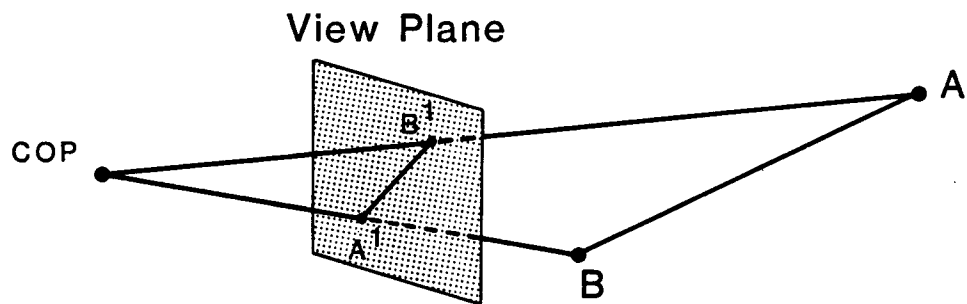


Figure B.1. Perspective Projection.

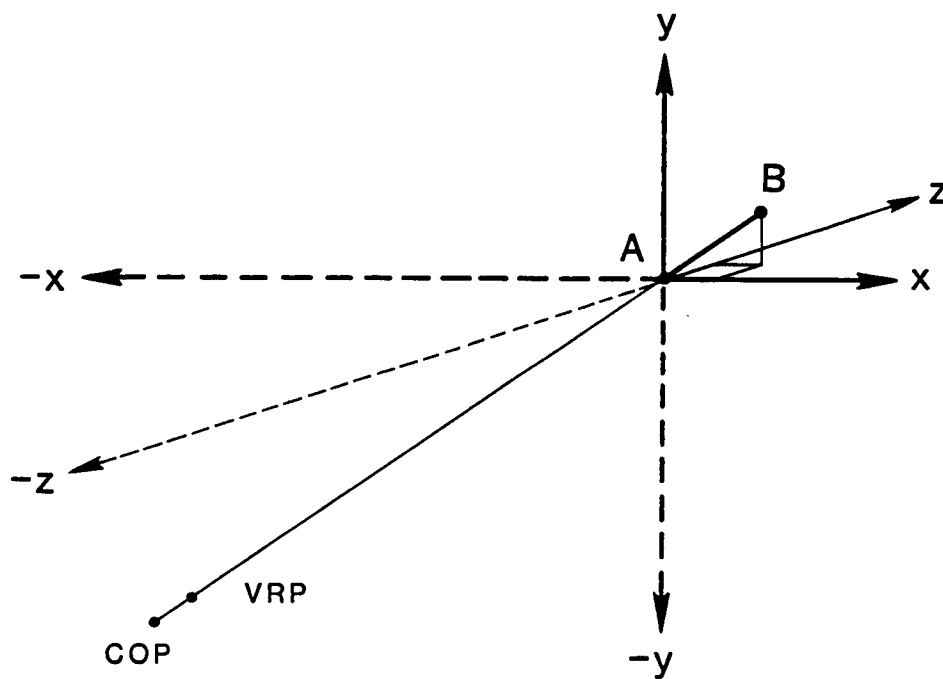


Figure B.2. Line Segment AB.

A three-dimensional image can be viewed from an infinite number of locations. Therefore, a view reference point (VRP) must be defined to view the scene at the desired location. The VRP lies in the view plane. Another point which needs to be defined at this time is the Center of Projection (COP). The COP is specified in scene coordinates relative to the VRP. For further information on perspective projections refer to Foley [3].

The next three transformations are executed to orient the scene coordinates of the object to a position relative to the view plane and COP. These transformations readily accomplish the projection of the object to the view plane. Again, for further information refer to Foley [3].

The VRP will be defined to be $(-5, -5, -5)$ and the COP is defined, in this presentation, as a unit vector starting at the VRP and having a direction equal to that of the VRP. Figure B.3 illustrates the COP and VRP with respect to line segment AB.

(2) T_2 Translate COP to origin (Figure B.4).

$P_2 = P_1 T_2$, where $T_2 =$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -(VRP_x + COP_x) & -(VRP_y + COP_y) & -(VRP_z + COP_z) & 1 \end{bmatrix}$$

$A_2 = (5.07, 5.07, 5.07)$

$B_2 = (6.07, 6.07, 6.07)$

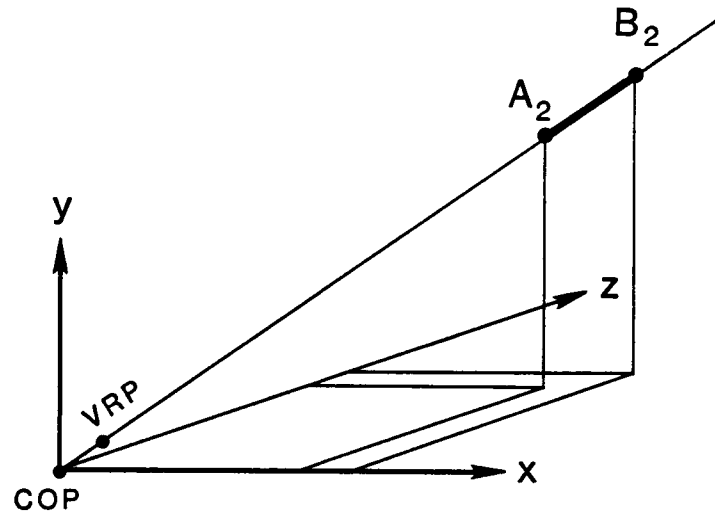


Figure B.4. Perspective Transformation T_2 of Segment AB.

(3) T_3 - Rotate VRP about y-axis into yz-plane. (Figure B.5)

$$P_3 = P_2 T_3, \text{ where } T_3 = \begin{bmatrix} -\frac{VRP_z}{D_{xz}} & 0 & -\frac{VRP_x}{D_{xz}} & 0 \\ 0 & 1 & 0 & 0 \\ \frac{VRP_x}{D_{xz}} & 0 & -\frac{VRP_z}{D_{xz}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $D_{xz} = \sqrt{(VRP_x)^2 + (VRP_z)^2}$

$$A_3 = (0, 5.07, 7.18)$$

$$B_3 = (0, 6.07, 8.59)$$

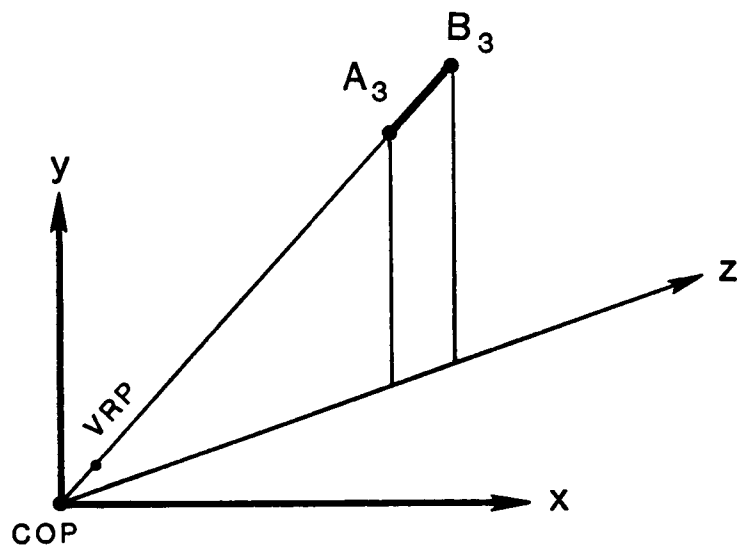


Figure B.5. Perspective Transformation T_3 of Segment AB.

(4) T_4 - Rotate VRP about x-axis into positive z-axis (Figure B.6).

$$T_4 = P_3 T_4, \text{ where } T_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{D_{xz}}{|VRP|} & \frac{-VRP_y}{|VRP|} & 0 \\ 0 & \frac{VRP_y}{|VRP|} & \frac{D_{xz}}{|VRP|} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\text{where } |VRP| = \sqrt{VRP_x^2 + VRP_y^2 + VRP_z^2}$$

$$A_4 = (0, 0, 8.79)$$

$$B_4 = (0, 0, 10.52)$$

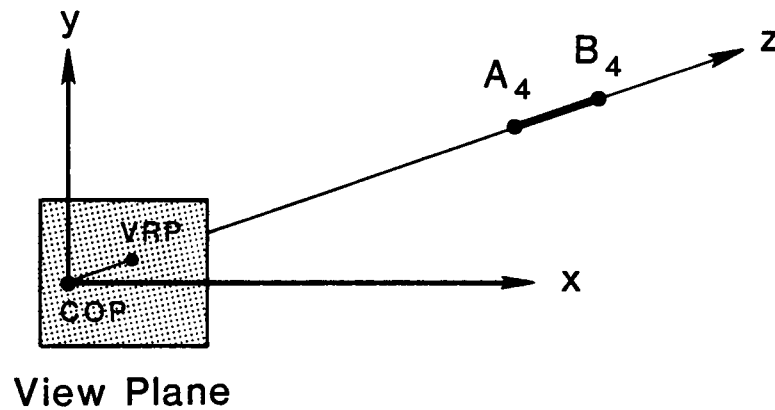


Figure B.6. Perspective Transformation T_4 of Segment AB.

(5) We are now in a position to easily project the three-dimensional scene coordinates, (x, y, z) , onto the two-dimensional view plane. The device coordinate of point P, XD and YD, are calculated as follows, referring to (Figure B.7).

$$XD = \frac{x}{z}$$

$$YD = \frac{y}{z}$$

The transformation matrix is as follows:

$$P_5 = P_4 T_5, \text{ where } T_5 = \begin{bmatrix} \frac{1}{z} & 0 & 0 & 0 \\ 0 & \frac{1}{z} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

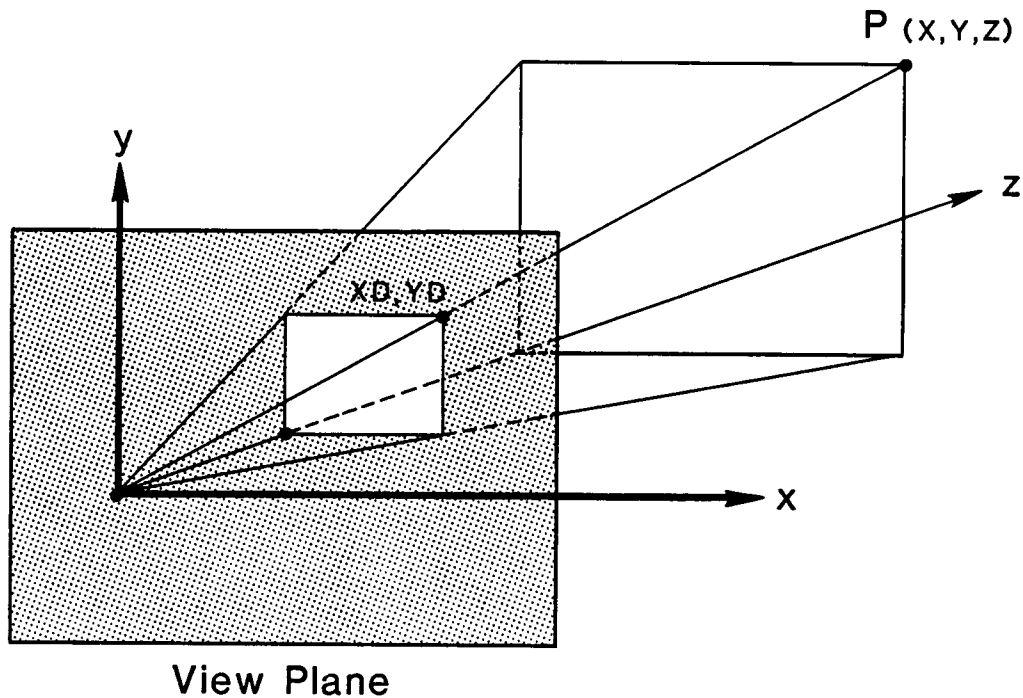


Figure B.7. Device Coordinate, XD and YD, Calculation from Scene Coordinate, (X, Y, Z).

$$A_5 = 0, 0$$

$$B_5 = 0, 0$$

(6) The device coordinate (0, 0) maps into the center of the device screen. Therefore, this next transformation is a function of the particular display device. That is, for the coordinate (0, 0) to appear in the center of the device screen, the number of x and y coordinates and the location of the origin must be defined. For the Vectrix, the origin is defined at the lower left of the screen and the number of pixels in the x and y directions are 672 and 480, respectively. Therefore, the device-dependent transformation for the Vectrix is as follows:

$$T_6 = P_5 T_6, \text{ where } T_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 672/2 & 480/2 & 0 & 1 \end{bmatrix}$$

The Vectrix location of points A and B are, as expected,

$$A_6 = 336, 240$$

$$B_6 = 336, 240$$

The composite matrix T_C , where $P_6 = P T_C = P T_1 T_2 T_3 T_4 T_5 T_6$, can be written as

$$T_C = \begin{bmatrix} R_{11} & R_{12} & R_{13} & R_{14} \\ R_{21} & R_{22} & R_{23} & R_{24} \\ R_{31} & R_{32} & R_{33} & R_{34} \\ R_{41} & R_{42} & R_{43} & R_{44} \end{bmatrix}$$

In summary, the following coordinate equalities map a three-dimensional scene coordinate into a two-dimensional device coordinate.

(Note, however, that T_6 is a device dependent transformation.)

$$X_D = X R_{11} + Y R_{21} + Z R_{31} + R_{41}$$

$$Y_D = X R_{12} + Y R_{22} + Z R_{32} + R_{42}$$

$$Z_D = X R_{13} + Y R_{23} + Z R_{33} + R_{43}$$

VITA

Mitchell J. Hurst was born in West Palm Beach, Florida on November 11, 1958. He attended elementary, junior high, and high school in that city and graduated from Forest Hill High School in June 1976. The following August, he entered Pfeiffer College in Misenheimer, North Carolina and the following school year transferred to the University of Florida where, in July 1981, he received a Bachelor of Science in Engineering. From August 1981 to July 1983 he worked at Arnold Engineering Development Center (AEDC) in Tullahoma, TN. In September 1983 he became a full time graduate student at The University of Tennessee Space Institute in Tullahoma, TN from which he received the Master of Science degree in Computer Science in March 1985.