A DIGITAL TERRAIN MODEL SYSTEM FOR A MICROCOMPUTER

FELGUEIRAS, Carlos Alberto - Software engineer
ERTHAL, Guaraci José - Researcher
DIAS, Luiz Alberto Vieira - Researcher
Instituto de Pesquisas Espaciais - INPE/MCT - DPI
C.P. 515 - 12201, São José dos Campos, SP, Brazil
Commission III

ABSTRACT

The objetive of this work is to describe the development and the implementation of a Digital Terrain Model (DTM), in a 16 bits microcomputer IBM-PC like, under a MS-DOS Operacional System . The model was developed at the Image Processing Department, Institute for Space Research (INPE), Brazil, and is integrated into a Geographical Information System that allows the combination of DTM data with other informations like images and thematic maps.

INTRODUCTION

Digital Terrain Models (DTM's) are mathematical models representing, on digital form, the behavior of a given variable associated to a terrain point (Doyle, 1978). In order to generate the model, a set of samples $(x_i, y_i, z_i ; i=1,2...n)$ are obtained from a real surface. The x and y coordinates are associated to the sample values spacial positions while the z coordinates represent the values of the variable modeled. This variable can be elevation (most common), temperature, wind speed, population distribution, etc....

The DTM process can be divided into three different phases: data sample, model generation and applications developed over the model generated.

A Geographical Information System (GIS) allows the integration, on a single data base, informations from several different origins , such as: thematic maps, DTM data, satellite images, statistical data on tables, etc.... On such a system a good visualization scheme is essential, because the volume of data is enormous.

On the GIS developed at INPE, DTM acquisition, generation and applications modules were included. In this implementation the data sample is gathered by an acquisition function that samples from a digitized tablet the points that will represent the actual terrain to be modeled. The sample points are then organized, and an interpolation process generates the model in a rectangular regular form grid. At the moment, two aplications are allowed : planar geometric projections and contour maps.

This GIS runs on any IBM-PC-XT or AT compatible microcomputer, in which a frame buffer SITIM (developed at INPE) of 1 to 4 planes 1024x1024 pixels was attached. The hardware consists of

a central processing unit, with this frame buffer and a 20 Mbytes hard disk, a keyboard, two monitors (one alphanumeric, one color graphic), and a digitizing tablet, as a minimum configuration. The system was developed on the C language and runs on the MS-DOS Operational System.

DATA ACQUISITION

The data sampling is a critical phase. The final quality of the DTM depends on the input data quality. Even if one uses an excellent interpolator, the results will be as good as the input samples. As an example, for a topographic chart the data must include relevant information about de drainage and crest lines.

The samples are, in general, irregularly spaced on the horizontal plane (x and y coordinates).

In this implementation only the data obtained manually from a digitized tablet were used, but samples from other sources, such as automatic scanner digitizers or stereoscopic equipments, could be used..

The data from a digitized tablet are read on a tablet coordinate system (x,y) and transformed to the user coordinate system (u,v). This makes possible the match of a DTM with another map or satellite image. The transformation is as follows:

$$u = a0 + a1\ x + a2\ y + a3\ x\ y$$

$$v = b0 + b1\ x + b2\ y + b3\ x\ y$$

,and if four points are known the coefficients ai and bi , i=0,1,2,3, can be determined and transformations as scale, rotation, skew and translation are made.

MODEL GENERATION

The model generation module consists of creating a poligonal grid from the samples points. Generally, it has two grid types: triangular and rectangular. The former uses the samples set directly and forms an irregular grid model, while the later is obtained by an interpolation process, i. e., the z values, for each point of the model, is estimated from the sample set. In this case a regular rectangular grid is obtained, i. e., the orthogonal projection of the surface grid points are represented as vertices of rectangles on the x-y plane.

In the present case it was decided to implement a rectangular regular grid because it is easier to generate it computationally and it requires less memory space to store the grid information than the triangular grid.

To obtain the regular rectangular grid the nearest-neighbours interpolation method was used , which uses the n nearest sample points, weighted to the inverse distance to the grid points. This interpolation function is as follows:

$$f(x,y) = \left(\sum_{i=1}^{n} w(x,y)_i * z_i \right) / \left(\sum_{i=1}^{n} w(x,y)_i \right)$$

where z_i are the nearest n neighbours

$$w(x,y)_i = (1/d_i)^m$$

$$d_i = ((x-x_i)^2 + (y-y_i)^2)^{1/2}$$ is the Euclidian distance from the point to the samples.

n = number of neighbours points to be considered within a certain influence radius R.
m = exponent of weighting function.

On this system, the influence radius can be set to accept a pre-determined number of points and the user can also choose the exponent of the weighting function, normally 2 or 4 (Felgueiras et al.,1987).

After the data gathering , and determination of the rectangular grid points, one can get a more dense rectangular grid. This is achieved by refining the grid and interpolating for the z values. The sampled data are on a file that, besides the values xi,yi,zi, contain also a header with useful information as: present number of points, maximum number of points, maximum and minimum values of the variables xi,yi and zi, and others. The control points are also stored, in case one needs to make a transformation.

The refining of the regular rectangular grid can be done by two methods: bilinear and bicubic interpolation. The formulation for the bilinear method is the following:

$$f(x,y) = \sum_{i=0}^{1} \sum_{j=0}^{1} a_{ij} x^i y^j$$

x and y $\in$ (0,1). For a point M(x,y), within the cell ABCD, where A(0,0), B(0,1), C(1,1), D(1,0), by interpolating linearly on BC and AD one determines the point values of z on E(xm,0) and F(xm,1). Then the value for z at the point M is obtained, interpolating on EF.

Then, for x and y $\in$ (0,1):

$$z = y(xz_D + (1-x)z_A) + (1-y)(xz_C + (1-x)z_B)$$

or

$$z(x,y) = \sum_{i=0}^{1} \sum_{j=0}^{1} a_{ij} x^i y^j$$
$$M$$

where

$$a_{ij} = \begin{bmatrix} z_B & z_C - z_B \\ z_A - z_B & z_D - z_A + z_B - z_C \end{bmatrix}$$

This interpolation is very fast on the computer, and for applications in which great accuracy is not essential, acceptable.

For better, but slower results, the Akima (Akima,1974) bicubic interpolator is used:

$$f(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j$$

The function determination can be done directly, if one knows the values of z and its derivatives for every vertice of a generic patch ABCD, i. e., the values of $z(x_i,y_i)$, $\dfrac{\partial z(x_i,y_i)}{\partial x}$, $\dfrac{\partial z(x_i,y_i)}{\partial y}$, and $\dfrac{\partial^2 z(x_i,y_i)}{\partial x \partial y}$; i = 0,1 and j = 0,1 .

Then (Foley and Van Dam,1982): $f(x,y) = X M P M^T Y^T$

where $X = [x^3 \; x^2 \; x^1 \; 1]$, $x \in [0,1]$;

$$M = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

is the Hermite matrix,

M^T = is the transpose matrix of M;

$Y = [y^3 \; y^2 \; y \; 1]$, $y \in [0,1]$;

and

$$P = \begin{bmatrix} z(0,0) & z(0,1) & \dfrac{\partial z}{\partial y}(0,0) & \dfrac{\partial z}{\partial y}(0,1) \\[2mm] z(1,0) & z(1,1) & \dfrac{\partial z}{\partial y}(1,0) & \dfrac{\partial z}{\partial y}(1,1) \\[2mm] \dfrac{\partial z}{\partial x}(0,0) & \dfrac{\partial z}{\partial x}(0,1) & \dfrac{\partial^2 z}{\partial x \partial y}(0,0) & \dfrac{\partial^2 z}{\partial x \partial y}(0,1) \\[2mm] \dfrac{\partial z}{\partial x}(1,0) & \dfrac{\partial z}{\partial x}(1,1) & \dfrac{\partial^2 z}{\partial x \partial y}(1,0) & \dfrac{\partial^2 z}{\partial x \partial y}(1,1) \end{bmatrix}$$

In this Hermite formulation, (Foley and Van Dam, 1982), Akima derived a fast method for determining the necessary derivatives, based on the data points, and its neighbours (Akima, 1974); for such points if the z values are known for the closest twelve points, the derivatives can be estimated. For border points a simple and fast extrapolation is used (Akima, 1970).

APPLICATIONS

Two applications are presented: the contour maps generation and planar geometric projections. For contour maps generation, the method used was based on data organized as a retangular horizontal grid. There are two parts: contour generation and contour plot.

The contour generation function is divided into four modules: inicialization, search for each z value, separation and compression of curves of same z values, and storage (Felgueiras, Dias and Erthal, 1987).

The plotting function, based on a GKS standard, allows the user to select several options: output device (plotter or CRT), normalized output window(0 to 1 in x or y on the screen or plotter), color (its possible to choose 255 colors). The z values may or may not be automatically writen on the master lines, the master lines are wider than the others and the spacing on z values are determined by the user. The interpolation, between two consecutives points in each contour line, may be linear or cubic (using Akima on two dimensions) (Akima, 1970)). Using the cubic interpolation one can gets a high degree of smoothness for the curves appearance. Figure 1 presents an example of linear interpolation with z values automatically written on the master lines. Figure 2 presents the same region (Agulhas Negras, RJ, Brazil), but with Akima interpolation between the points that form contour curves.

Geometric projection is possible from the regular grid points. The portion of the observer can be altered, and the projection can be or perspective or paralel projection. Figure 3 shows the Agulhas Negras region as a geometric paralel projection. Its also possible to generate a stereo pair, which allows an observer to view the region under study in three dimensions, using filter glasses. Figure 4 presents a plotter generated stereo pair of Agulhas Negras region.
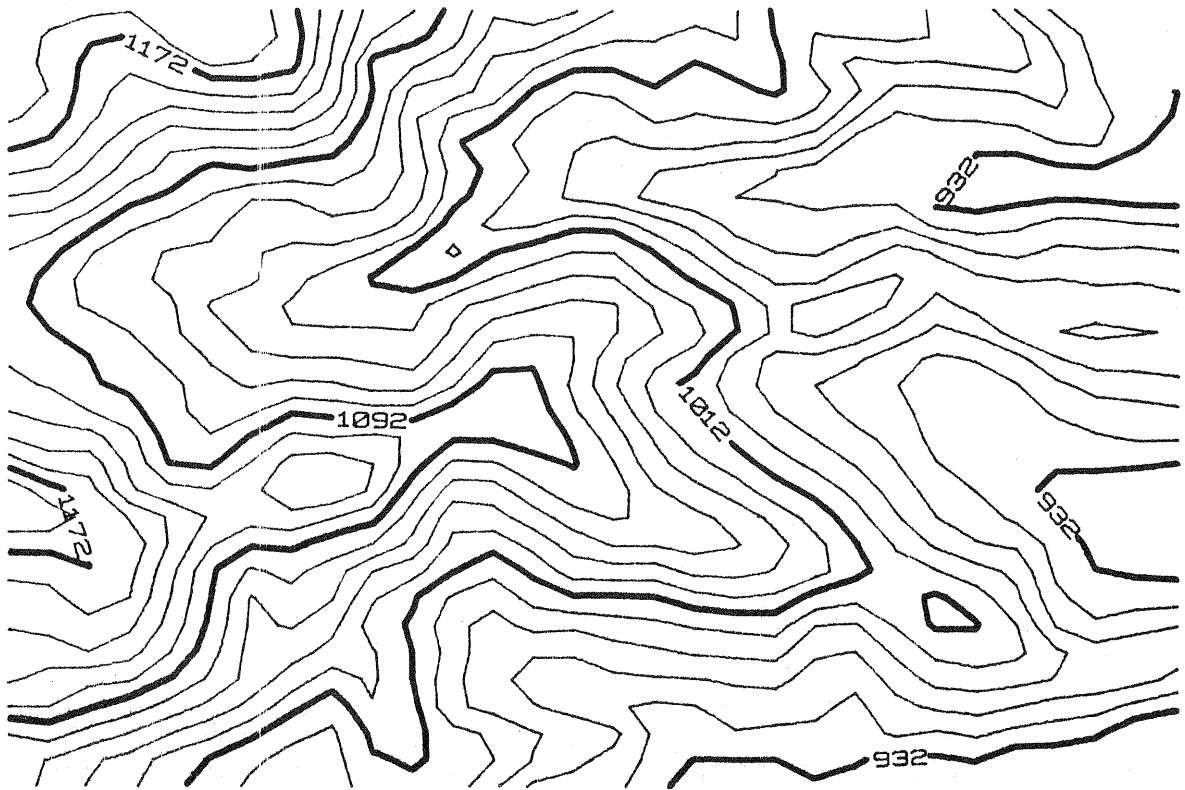
Figura 1 - Contour map of Agulhas Negras, RJ, Brazil, showing
automatically written z-values for master lines, linearly
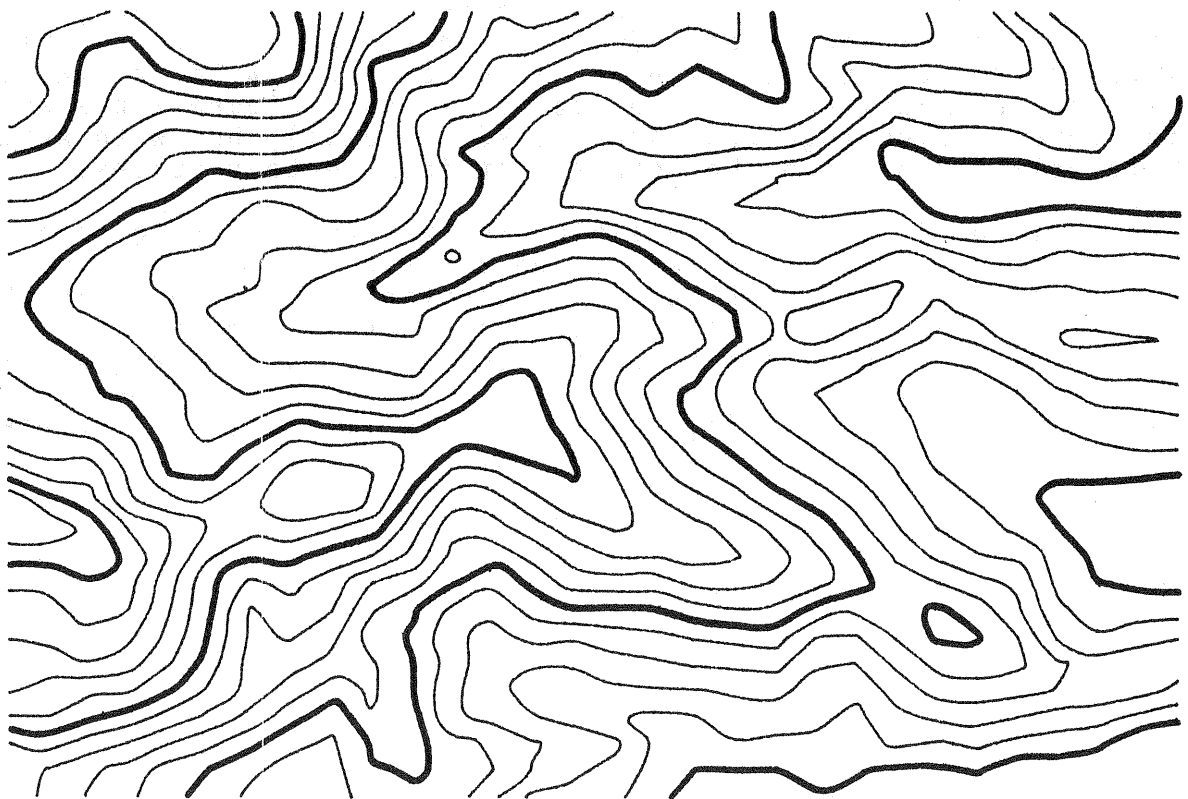interpolated between grid points.

Figure 2 - Contour map of Agulhas Negras, RJ, Brazil, with
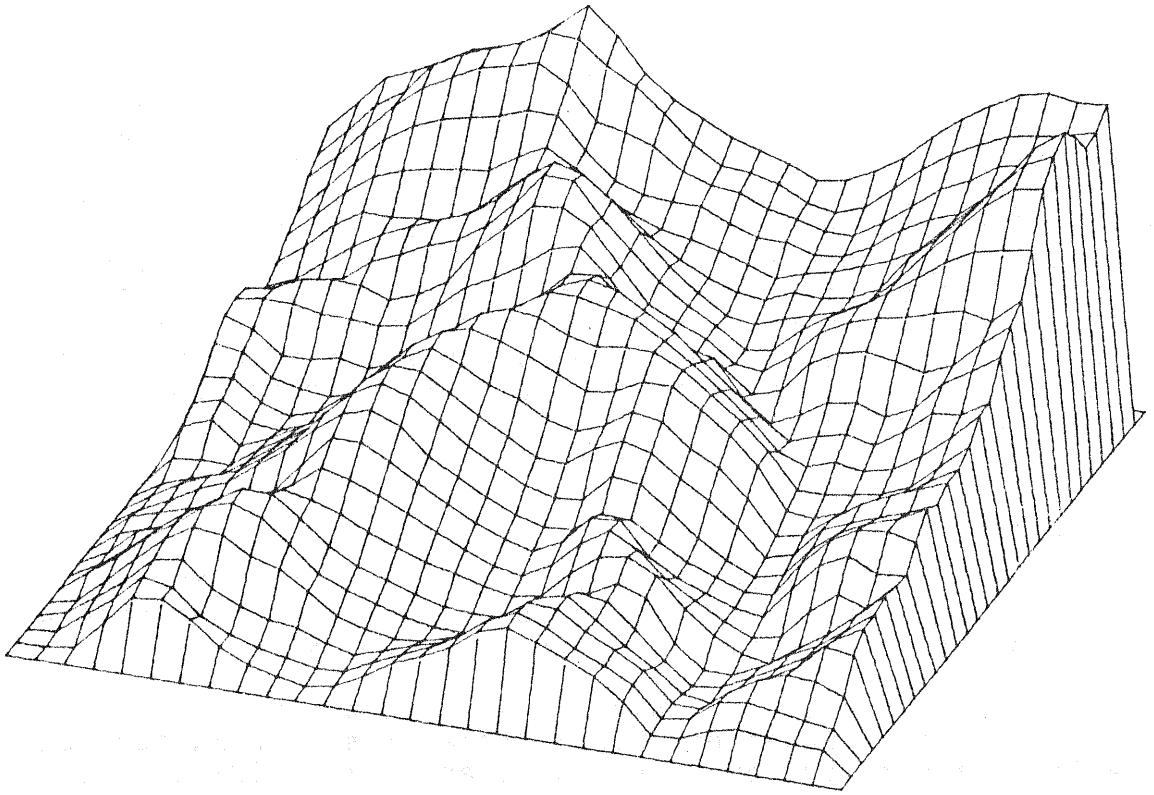Akima interpolation betwenn grid points.

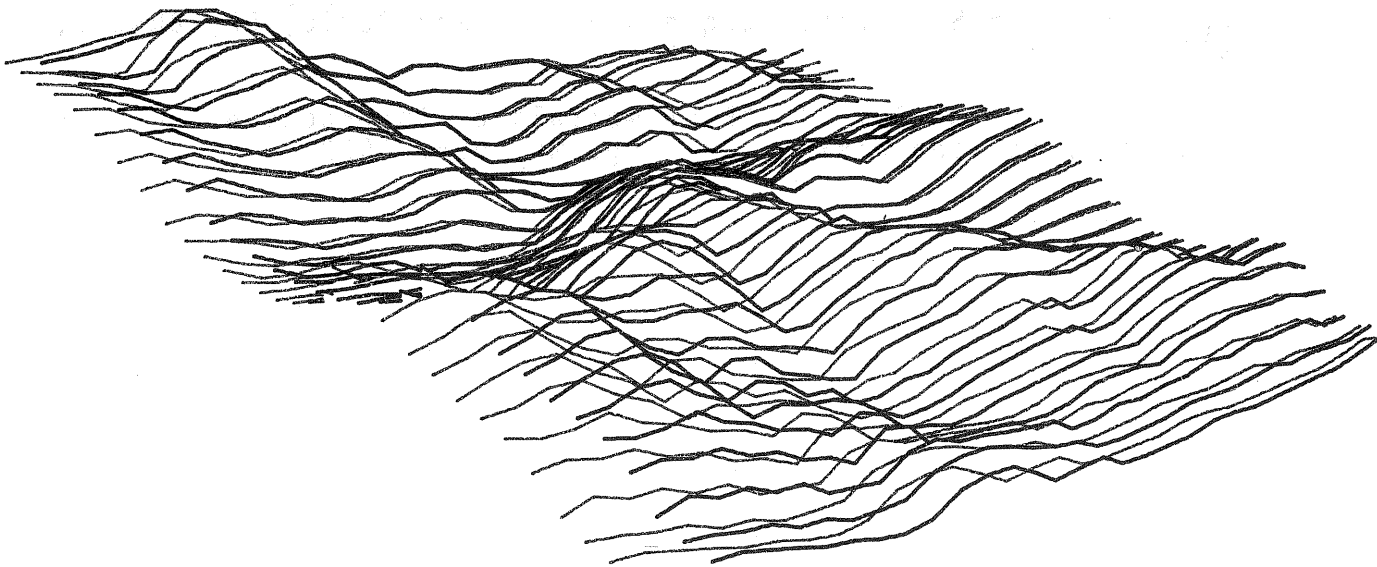Figure 3 - Paralel perspective view of Agulhas Negras, RJ, Brazil.

Figure 4 - Perspective stereo pair view of Agulhas Negras, RJ, Brazil.

CONCLUSION

The DTM here presented is integrated to INPE's Geographic Information System, and has been of great value for visualization purposes. It can be also utilized independently. The development on C computer language tooks about two years by roughly two people full time. Some improvements are possible, as the inclusion of non-convex border for interest regions, and the use of triangular grids. As far as performance, the system works satisfactorily for regions of up to 20000 samples, which is the size for which it was designed. Still there are room for improvement, by use of better algorithms and accelerator cards. As a final conclusion it is shown that microcomputers are capable of run DTM's.

REFERENCES

Akima, H. "A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures". Journal of the Association for Computing Machinery, 17(4):589-602,Oct,1970.

Akima, H. "A Method of Bivariate Interpolation and Smooth Surface Fitting Based on Local Procedures". Communications of the Association for Computing Machinery, 17(1):18-20,Jan, 1974.

Doyle, F. J. "A Digital Terrain Models: An overview". Photogrametric Engineering and Remote Sensing, 44(12):1481-1485,Dec,1978.

Felgueiras, C. A.; Dias, L. A. V.; Erthal, G. J. "Geração Automática de Mapas de Isolinhas Utilizando Microcomputador" Proceedings of the XX National Informatics Congress (XX Congresso Nacional de Informática). SUCESU, pages 425-430, August 31st to Sept 6th, Parque Anhembi, São Paulo, SP, BRAZIL, 1987.

Foley, J. B.; Van Dam, A. "Fundamentals of Interactive Computer Graphics". Addison-Wesley, Reading, MA, USA, 1982.