Geo-Processing, 1 (1979) 53—70

DIGITAL MODEL FOR THREE-DIMENSIONAL SURFACE REPRESENTATION

DORA GOMEZ and ADOLFO GUZMAN

Computer Science Department, IIMAS, National University of Mexico, Mexico 20 D.F. (Mexico)

ABSTRACT

Gómez, D. and Guzmán, A., 1979. Digital model for three-dimensional representation. Geo-Processing, 1:53-70

A tree of planar or spherical triangles is used to represent a 3-d surface; rather flat regions will be represented by large triangles, while abrupt zones will require further subdivision of the model into smaller triangles. Their vertices are not placed on a regular grid; they are allowed to fall at (or near) places such as ridges and peaks, where the change in slope is significant.

Starting from a collection, not necessarily good or complete, of "significant" points, the model selects five of them to form four triangles. Each triangle either matches the surface within a prespecified error tolerance, or else is further subdivided, by selecting appropriate "significant" points, into four triangular sons, which then receive in turn the same treatment. The tree stops growing when all the surface is represented within the specified tolerance. The model consists of the vertex points arranged into a table suitable for quick retrieval and interpolation.

Thus, the model guides its own construction; its components (points) are taken from the set of "significant" points, not in an arbitrary fashion but only where and when needed. Since the model proposes the approximate location of the next point to be included in it, the set of "significant" points may be small or non-existant.

A constant signal to noise ratio and a representation thrifty in storage are achieved in this manner.

The model is being tested for use in digital representation of terrain elevation. Large savings in memory are expected, when compared to contour lines storage, for instance.

The paper concludes with some comments in favor of the use of this model to describe gray level pictures.

INTRODUCTION

The digital representation of three-dimensional surfaces plays important roles in photogrammetry and cartography (drainage patterns, contour lines (Gómez, 1978), valleys formation, stereoscopy (Gómez), dimensions of the human body); scene analysi: (oclusion of bodies, explanation of regions (Guzmán, 1971) and objects, range finding, shape from shading (Horn, 1970); computer graphics (hidden lines, shadows, coloring, specular reflexions); image processing; remote sensing (thickness of ice (Jensen, 1976), underground geology), and other disciplines.

Artículo 35

This is not a problem for 3-d surface representation, as used for instance in applications to cartography and computer graphics. For surface comparison it is

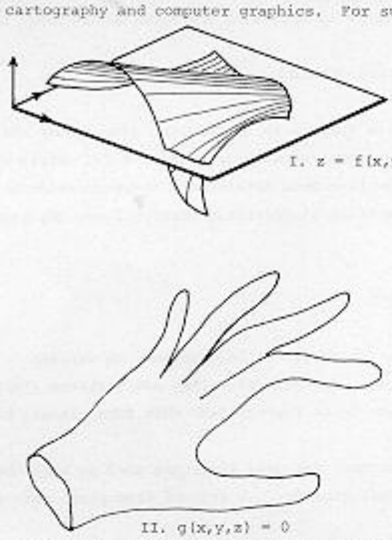


Fig. 1. TYPES OF SURFACES. The model described in this paper is able to represent either (I) single height surfaces, or (II) more general surfaces.

much better to have a unique (canonical) model, perhaps through a normalization procedure.

The set of significant points. Using a method external to the model, for instance stereoscopy (Gómez), gradient extraction (Signor and Nadler,1978), river following, or others, an initial set of "significant" points is chosen on the 3-d surface that we want to represent. A point is called "significant" if in its neighborhood the change in slope is large.

The model begins by using some of these points; if it later finds necessary to grow, it indicates the approximate place (x,y coordinates) where a new "significant" point should be added to the model.

The model thus consists of a subset of "significant" points, defining a triangular irregular mesh; if the original set of "significant" points is too small, the model will suggest where to add one; if too many, most of them will be ignored (not included in the model); if the procedure that implements "significance" is noisy or unreliable, the model still guarantees the ε tolerance, but storage economy suffers.

Therefore, in a computer implementation, it is not necessary to obtain first the set of significant points and then to pick the model from them; instead, the model can begin to grow as soon as five or six are found, and the procedure that extracts significant points is called by the model as it deems necessary.

Obtention of the three-dimensional surface

It is assumed that the surface to be modelled already was obtained and exists available in some suitable representation, v.gr., a 2-d matrix containing height values. This data could have been obtained by stereocorrelation (Gómez) of a pair of pintures, by interpolation of digitized contour lines (Bribiesca and Avilés, 1974) or by other means.

CONSTRUCTION OF THE MODEL

In order to describe the model, it is necessary to explain

(1) its constituent parts. In this case, they are vertices ("significant" points from the 3-d surface to be represented) that form planar, but tilted, triangles.

(2) how the model is sotred; the data structure used to keep the model in memory (primary or secondary storage). A tree of triangles, each with none or four sons, is used.

(3) the use of the model: the procedure to follow for reconstruction of the 3-d surface from the model; the way to obtain the coordinates of a point in the surface from the cover of triangles. Here a directed access is used to the correct triangle starting from the top of the tree of (2), and falling down the appropriate chain of triangle sons, using little search and no backtracking.

(4) the construction of the model, i.e., the obtention of its parts from the 3-d surface. A recursive procedure will be presented, where the model guides its own construction, by suggesting places (x,y coordinates) where to incorporate into itself points from the 3-d surface that are also " significant" with respect to changes in slope.

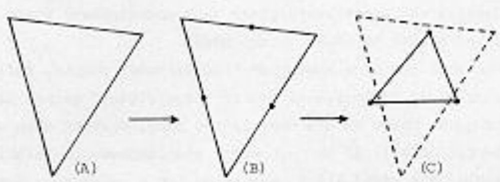


Fig. 2. TRIANGLE REFINEMENT. If it is necessary to refine triangle (A), three new vertices are proposed at the mid-points (B) of the sides; "significant" points are located near those mid-points; once they are found (C), four new triangles stand instead of the original (A).

model can begin to grow as soon as five or six are found, and the procedure that extracts significant points is called by the model as it deems necessary.

Obtention of the three-dimensional surface

It is assumed that the surface to be modelled already was obtained and exists available in some suitable representation, v.gr., a 2-d matrix containing height values. This data could have been obtained by stereocorrelation (Gómez) of a pair of pintures, by interpolation of digitized contour lines (Bribiesca and Avilés, 1974) or by other means.

CONSTRUCTION OF THE MODEL

In order to describe the model, it is necessary to explain

(1) its constituent parts. In this case, they are vertices ("significant" points from the 3-d surface to be represented) that form planar, but tilted, triangles.

(2) how the model is sotred; the data structure used to keep the model in memory (primary or secondary storage). A tree of triangles, each with none or four sons, is used.

(3) the use of the model: the procedure to follow for reconstruction of the 3-d surface from the model; the way to obtain the coordinates of a point in the surface from the cover of triangles. Here a directed access is used to the correct triangle starting from the top of the tree of (2), and falling down the appropriate chain of triangle sons, using little search and no backtracking.

(4) the construction of the model, i.e., the obtention of its parts from the 3-d surface. A recursive procedure will be presented, where the model guides its own construction, by suggesting places (x,y coordinates) where to incorporate into itself points from the 3-d surface that are also " significant" with respect to changes in slope.

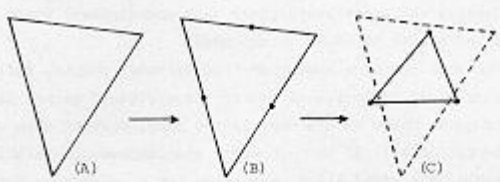


Fig. 2. TRIANGLE REFINEMENT. If it is necessary to refine triangle (A), three new vertices are proposed at the mid-points (B) of the sides; "significant" points are located near those mid-points; once they are found (C), four new triangles stand instead of the original(A).

The parts of the model

To represent a surface $z = f(x,y)$, the model uses a collection of planar tilted triangles; each of them is defined by its three vertices, chosen to lie on the surface $z = f(x,y)$ to be represented.

All the points inside the triangle are interpolated linearly: the surface inside the triangle is considered flat (but not horizontal, in general). Since the real surface $z = f(x,y)$ is not flat, an error is introduced by this assumption. If everywhere in the triangle this error (height difference) does not exceed a tolerance ε, the planar triangle is considered to be a good (and final or "terminal") representative for that region of the surface, and it is included in the model. If the error is larger, the triangle is discarded by dividing it into four smaller triangles, each of which in turn undergoes the same treatment.

Initially the surface is divided into a small set of arbitrarily chosen large triangles; if the surface is bound by a rectangle (as it is frequently the case in maps), four triangles are chosen as shown in part C of Figure 3.

The final model contains triangles (of different sizes) that represent the surface $z = f(x,y)$ with a tolerance ε. Each of the vertices of these triangles was proposed by the model by dividing a triangle in four through inclusion of new vertices near the middle points of the sides (Figure 2).

Once every triangle is refined, the vertices (E in Fig. 3) are stored in an appropriate way, suitable for quick data retrieval for surface reconstruction.

When to stop refining

A triangle such as in Figure 2A is refined further, unless

1) the difference between the real height $z = f(\bar{x},\bar{y})$ and the computed height $\bar{z}$ at the center of mass $(\bar{x},\bar{y},\bar{z})$ of the triangle is smaller than ε, and
2) every point in a grid of points spaced at most K units apart and inside A is within ε of the real point on the surface $z = f(x,y)$.

Test (1) is a quick test; test (2) is applied only if (1) does not find a difference exceeding ε.

K, the distance between two points in the grid of (2), is a function of ε; normally, $K = \min(\varepsilon/m, K_0)$, where m is the mean slope of the surface at the triangle (A), and K_0 is the diameter of the smallest topographic feature (hill, ravine) that it is necessary to represent in the model. Generally K_0 is given by the user of the model: "be sure to check the model every 500 horizontal meters for accuracy"; then $K_0 = 500$.

Flow diagram. The procedure for construction of the model could be summarized as:

- Let T be the set of triangles that are candidates to be included in the model. Initialize T with the four triangles of (C), Fig. 3.

- Mark every triangle of T as "terminal" if it passes tests (1) and (2) of the Section "When to stop refining". If these tests fail for a triangle, mark it "non-terminal", divide it into four sons (cf. Fig. 2) and add them to T.
- Exit when all triangles of T (including all the additions to T) are marked (either "terminal" or "non-terminal"). Then T is the model.

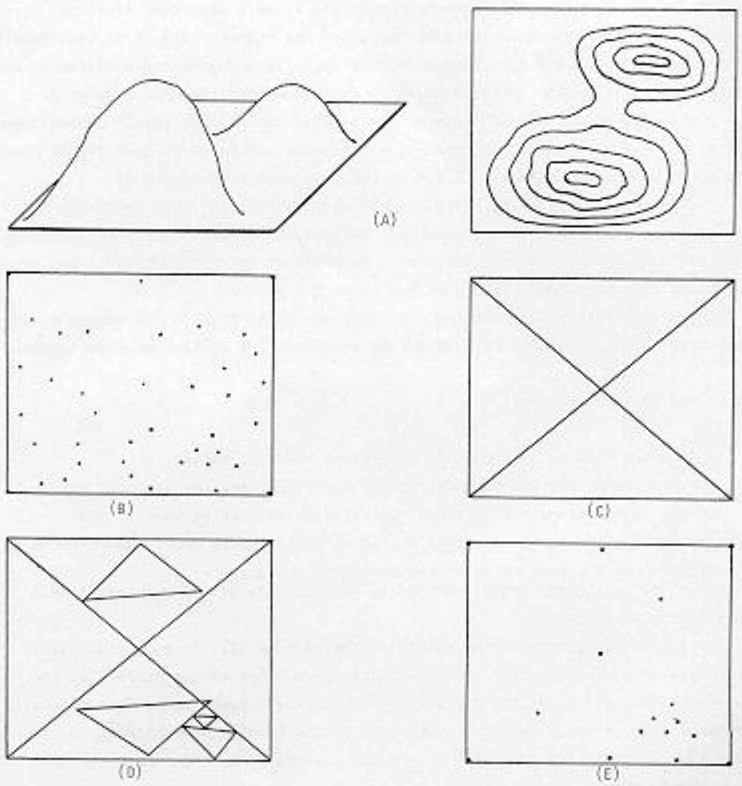


Fig. 3. MODEL BUILDING. (a) the surface, (b) the "significant" points, (c) the four initial triangles, (d) the final triangles, (e) the points of the model, usually a subset of (b). Triangles (d) are in the space (they project out of the paper); similarly, points (e) have three coordinates.

TABLE I. FLOW DIAGRAM FOR MODEL BUILDING.

This simple program constructs surface models such as that shown in fig. 6.

```
BEGIN
     T  the four initial triangles of fig. 3C;
     For every triangle in T
          if it passes tests (1) and (2) of section "When to stop refining"
          then mark it 'terminal'
          else mark it non terminal   and
               add its four sons to T;
END.
```

A non-terminal triangle is not needed in the model, since

(1) its accuracy is worse than ϵ, and

(2) some of its descendants are *a fortiori* terminal triangles, hence suitable for modelling.

Thus, the model could be just the collection of terminal triangles.

This is advisable when the cover is made of similar triangles (q.v.), where it is easy to pick up the correct triangle for surface reconstruction. If the triangles are not similar, it is preferable to retain the non-terminal triangles into the model. This facilitates the addressing of the correct terminal triangle that gives the height $\bar{z}$ of a point (x,y) (i.e., the point $(x,y,\bar{z})$ that represents the point (x,y,z) of the 3-d surface). More of this in the section IV 'Data Retrieval for Surface Reconstruction'.

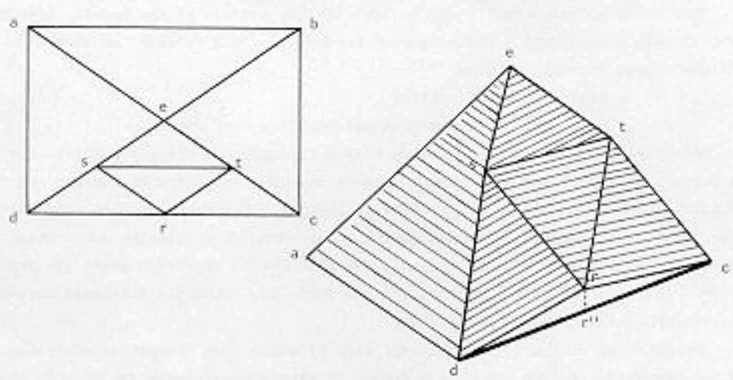


Fig. 4. MODEL VISUALIZATION. We try to give an isometric view of the appearance of the triangular model in 3-d space. Point $S=(s_x,s_y,s_z)$ does not lie on the 3-d line d → e; but point $(s_x,s_y,0)$ does lie on the line $(d_x,d_y,0) \rightarrow (e_x,e_y,0)$. Triangles such as $d\,\hat{e}\,s$ or $\hat{d}\,r\,\hat{c}$ are not part of the model; they represent no part of the real 3-d surface because they lie vertical. They are useless.

Cover of similar triangles

Two polygons are similar if the corresponding angles are equal, the sides parallel and their length proportional.

If in Figure 2 we stop the refinement at (B), choosing the midpoints as new vertices to include in the model, the final cover of the model is composed of two families of similar triangles, because a line joining the middle points of two sides is parallel to the third side.

A word of caution: the triangles are not similar as they lie in the 3-d space. Their projections on the plane x,y do form a family of similar 2-d triangles (for triangles *a b e* and *c d e* of Fig. 4, and all their descendants) another family of similar 2-d triangles for triangles *d a e* and *b c e*, and all their descendants.

The advantages of the cover of similar triangles are:

(a) storage of these triangles is easy. (Klinger and Nikitas) stores a hierarchy of squares.

(b) reconstruction of the surface from the model becomes simplified.

(c) a set of "significant" points (B in Fig. 3) is not needed.

The disadvantage comes from (c):

(d) the model might contain more points, since they are not special or significant: they are not the best to choose for interpolation of planes.

DATA STRUCTURE FOR MODEL STORAGE

This section describes the way to organize the storage of the model. Essentially, the storage consists of a collection of triangles. Each triangle is stored in a "frame"; each of them contains

- three internal vertices
- a "terminal" or non-terminal mark for each son.

The terminal mark (zero) indicates that a triangle son already fulfills the ϵ accuracy, hence it (the son) has no sons of its own --need not be further subdivided.-- The non-terminal mark, an integer different from zero, indicates the location (frame) in the model matrix occupied by this triangle son. Thus, when a node is marked as non-terminal, the mark itself also says where (in what frame) that son is stored. See Fig. 5 and Table II. Slightly different conventions were used in IIMAS-UNAM (Gómez, 1978).

The model is stored in a matrix (C, Fig. 5) which is a collection of frames. A non-terminal triangle occupies a frame; it stores clockwisely (B, Fig. 5) its three central vertices and a mark specifying for each son whether it is terminal or not. A terminal triangle does not use a frame, since it has no sons. But a non-terminal triangle could very well have four terminal sons. That is the case of frames 4 to 9 of Fig. 6.

The initial frame, frame 1, is stored in a slightly different manner (part A of Fig. 5, because it describes a rectangle.

A more complicated example is given in Fig. 6.

<u>Storage of vertices.</u> When describing a non-terminal triangle (v.gr., triangle 1 2 3 of B, Fig. 5, oncly vertices 4, 5 and 6 are stored in the frame belonging to that triangle 1 2 3, since vertices 1, 2 and 3 were undoubtely stored in the ancestors of triangle 1 2 3. This avoids multiple storage of vertices, and exploits the fact that in order to examine whether a point (x,y) fallsinside the

TABLE II. NAMING CONVENTIONS.

These conventions are important for correct storage of vertices (such as 4 of the internal triangle P), and its subsequent appropriate retrieval for reconstruction of the 3-d surface. For a use, see definition of procedure 'altitude' in section "Data retrieval for surface reconstruction".

CONVENTIONS I. *Refer to part (A) of Fig. 5.*

Vertices of triangles which are sons of the rectangle are named as shown. The correct names for (A) are:

rectangle: a b c d

M = triangle a b e

N = triangle b c e

O = triangle c d e

P = triangle d a e

CONVENTIONS II. *Refer to part (B) of Fig. 5.*

Vertices of triangles that are sons of triangles are named clockwisely, starting with the vertex that also belongs to the father.

If the triangle to be named is the internal triangle (P), then start with the vertex that falls near the middle point of line 1 → 2, where 1 is the first of the vertices that belong to the father, and 2 is the second of them.

The correct names for triangles of (B) are:

triangle 1 2 3 (first vertex is 1)

M = triangle 1 4 6

N = triangle 2 5 4

O = triangle 3 6 5

P = triangle 4 5 6

2-d[2] triangle *1 2 3* or not, we already asked a similar question to the ancestors of *1 2 3*. In this way the coordinates fo vertices 1, 2 and 3 are already known when triangle *1 2 3* is accessed (cf. Section 'Data Retrieval for Surface Reconstruction').

A vertex is stored by storing its three coordinates x, y, z. Some duplication (not triplication or multiplication) occurs when a vertex such as 5, 20 or 15 in Fig. 6 gets stored by two non-terminal brother triangles. For instance, vertex 5 is stored at frame 4 that describes triangle *3 1 9*, and also at frame 3 that

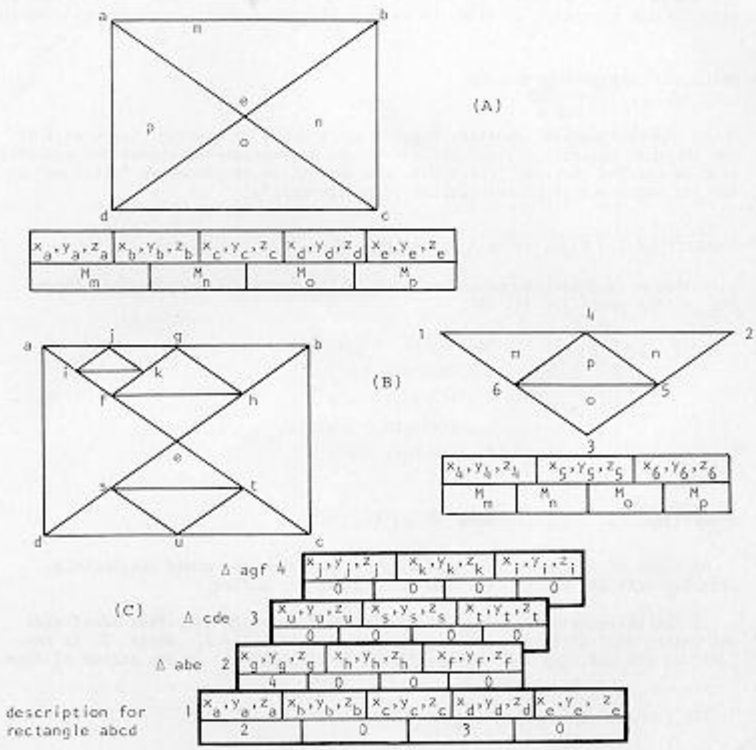


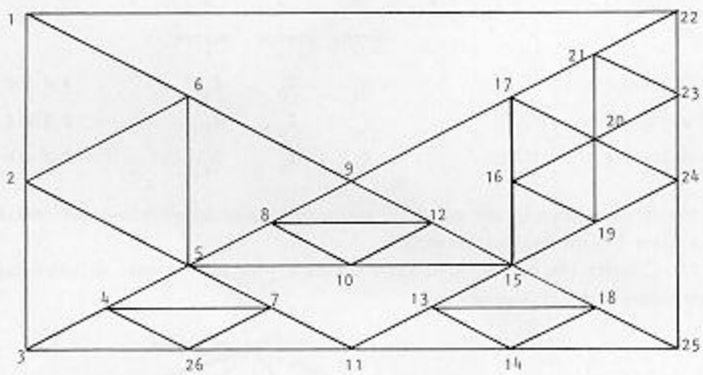
Fig. 5. DATA STRUCTURE. (A) Storage conventions for the initial rectangle. (B) Storage conventions for a non-terminal triangle *1 2 3*. Its sons are *M*, *N*, *O*, *P*. (C) Example of a model and its data structure. Only a non-terminal triangle uses up a frame. The 2 0 3 0 marks of frame 1 mean that son M is non-terminal and it is described in frame 2, son N is terminal, son O is non-terminal and it is described in frame 3 and son P is terminal (mark = 0 means terminal). The model is stored in a matrix (C) which is a collection of frames.

describes triangle 2 5 3 9. The trivial cure will be to keep a table of vertices, and to store in the frame pointers to the table, instead of the three coordinates x,y,z.

This table of vertices is not used in our model because it saves little storage:

(1) if both a pointer and a vertex coordinate occupy a word of memory, then to use the table requires 2 pointers + 3 coordinates = 5 words; not to use the table requires 3 coordinates + the same 3 coordinates = 6 words;

(2) if for some reason triangle 25 3 9 selects vertex 5 as the "significant" point near the mid-point of side 3-9 (Refer to Fig. 6), but triangle 3 1 9 selects vertex 5' (a different vertex, near vertex 5 but not the same) as the "significant" point near the mid-point of side 9-3, then the table wastes memory.



	frame	VERTEX a	b	c	d	e	Δ M	Δ N	Δ O	Δ P
(□ 1 22 25 3)	1	1	22	25	3	9	0	2	3	4

		VERTEX 4	5	6				
(Δ 22 25 9)	2	24	15	17	5	0	0	6
(Δ 25 3 9)	3	11	5	15	7	8	9	0
(Δ 3 1 9)	4	2	6	5	0	0	0	0
(Δ 22 24 17)	5	23	20	21	0	0	0	0
(Δ 24 15 17)	6	19	16	20	0	0	0	0
(Δ 25 11 15)	7	14	13	18	0	0	0	0
(Δ 3 5 11)	8	26	4	7	0	0	0	0
(Δ 9 15 5)	9	8	12	10	0	0	0	0

Fig. 6. MODEL EXAMPLE. This example was constructed using the rules (A) and (B) of Fig. 5 and Table II.
Each frame consists of vertices and pointers to other frames. Only non-terminal triangles occupy a frame of the matrix. This matrix is the model.

Simplified storage for cover of similar triangles

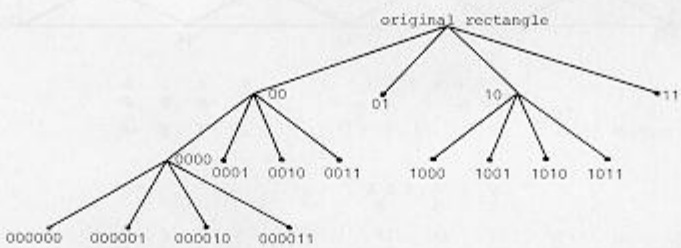
If we assume that the rectangle *a b c d* (Fig. 5) is a square and that the "significant" points are exactly at the mid-points of the sides of the triangles, instead of near them, then all the two-dimensional triangles[2] are similar (in fact, they are isosceles right angled triangles) and the (x,y) coordinates of any vertex need not be stored, since they are the average of the (x,y) coordinates of the vertices of an appropriate side.

The new representation for square *a b c d* of Fig. 5 is:

	Frame #	vertex a	vertex b	vertex c	vertex d	vertex e	Δ Δ Δ Δ M N O P
(*rectangle a b c d*)	1	z_a	z_b	z_c	z_d	z_e	2 0 3 0
			vertex 4	vertex 5	vertex 6		
(*triangle a b c*)	2		z_g	z_h	z_f		4 0 0 0
(*triangle c d e*)	3		z_u	z_s	z_t		0 0 0 0
(*triangle a g f*)	4		z_j	z_k	z_i		0 0 0 0

If the original area is not an square but a rectangle, we will have two families of similar two-dimensional triangles.

If we denote the sons M, N, O and P by 00, 01, 10 and 11, then we could form from Figure 5 the following tree:



First generation	Second generation	Third generation
Δ *a b e* = 00	Δ *a g f* = 00 00	Δ *a j i* = 00 00 00
	Δ *b h g* = 00 01	Δ *g k j* = 00 00 01
	Δ *e f h* = 00 10	Δ *f i k* = 00 00 10
Δ *b c e* = 01	Δ *g h f* = 00 11	Δ *j i k* = 00 00 11
Δ *c d e* = 10	Δ *c u t* = 10 00	
	Δ *d s a* = 10 01	
	Δ *e t s* = 10 10	
	Δ *u s t* = 10 11	
Δ *d a e* = 11		

These codes could be combined with the z values to render a compact model. We do not pursue this further. In a similar manner, a tree of squares can be represented (Klinger and Nikitas).

DATA RETRIEVAL FOR SURFACE RECONSTRUCTION

In order to recover the 3-d surface, it is sufficient to ask the model what is the z value for any pair x,y. This is realized by the function ALTITUDE.

```
ALTITUDE (x,y)    % returns the height z of the point (x,y) as obtained from
                  % the model. It is defined as:

a := MODEL [1,1];   % first vertex of frame 1. Frame 1 is the rectangle.
b := MODEL [2,1];   % MODEL [*,L] is the frame L, a non-terminal triangle.
c := MODEL [3,1];   % MODEL [*,*] is the matrix containing the whole model.
d := MODEL [4,1];
e := MODEL [5,1];
m := MODEL [6,1];    n := MODEL [7,1];    o := MODEL [8,1];
p := MODEL [9,1];   % retrieving the pointers to the sons.
error := -1;

ALTITUDE := if inside (a,b,e,x,y)
           then       if m=0 then height (a,b,e,x,y)
                      else ZETA (a,b,e,x,y,m)
           else if inside (b,c,e,x,y)
           then       if n=0 then height (b,c,e,x,y)
                      else ZETA (b,c,e,x,y,m)
           else if inside (c,d,e,x,y)
           then       if o=0 then height (c,d,e,x,y)
                      else ZETA (c,d,e,x,y,m)
           else if inside (d,a,e,x,y)
           then       if p=0 then height (d,a,e,x,y)
                      else ZETA (d,a,e,x,y,m)
           else error;
END ALTITUDE.
```

Function INSIDE (a,b,c,x,y) is true if the point (x,y,0) is inside the triangle $(a_x, a_y, 0)$, $(b_x, b_y, 0)$, $(c_x, c_y, 0)$ with sidewalks (see Fig. 7).

A point *p* is inside triangle *a b c* if *p* and *c* fall on the same side of *a b* and *p* and *b* lie on the same side of *a c*, and *p* and *a* rest on the same side of *b c*. A thesis (Gómez) contains listings and results.

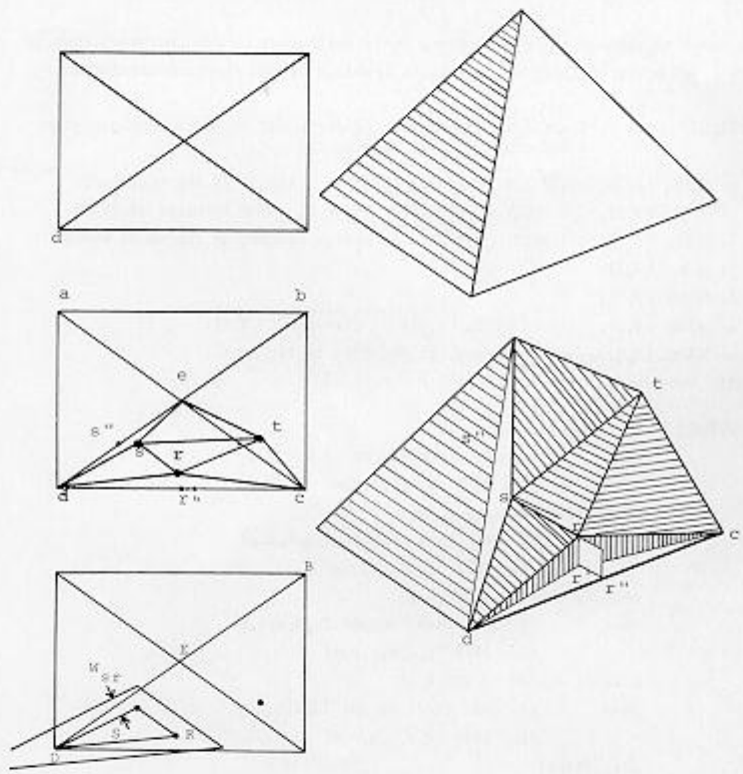


Fig. 7. COPLANAR SIDEWALKS. Compare with Figure 4. If point $r = (r_x, r_y, 0)$ does not fall on line $(c_x, c_y, 0) \rightarrow (d_x, d_y, 0)$, an horizontal area $c\ d\ \lambda^r$ will be without coverage by the triangles; a corresponding part of the 3-d surface will fail to be represented. The cure for this is to give "flaps" to the triangles, so that triangle $\lambda\ d\ \delta$ (and its other three brothers) are enlarged by a coplanar sidewalk that covers up to λ

```
Procedure ZETA (v1, v2, v3, x, y, m) is defined as
v4 := model [1,m];
  v5 := model [2,m];
    v6 := model [3,m];
      mm := model [4,m];
        n := model [5,m];
          o := model [6,m];
            p := model [7,m];
ZETA := if inside (v1,v4,v6,x,y)
        then     if mm=0 then height (v1,v4,v6,x,y)
                 else ZETA (v1,v4,v6,x,y,mm)   § see Table II 'Naming Conventions'
        else if inside (v2,v5,v4,x,y)
        then     if n=0 then height (v2,v5,v4,x,y)
                 else ZETA (v2,v5,v4,x,y,n)
        else if inside (v3,v6,v5,x,y)
        then     if o=0 then height (v3,v6,v5,x,y)
                 else ZETA (v3,v6,v5,x,y,o)
        else if p=0
        then     height (v1,v4,v6,x,y)
        else     zeta (v1,v4,v6,x,y,p);
END ZETA.
```

The search for the correct triangle that represents a point generates no backtracking. At each level of the tree of triangles, we simply go down to the next level through the appropriate son (that son containing the point), until we hit a terminal triangle, where we compute the height by a planar interpolation.

CONCLUDING REMARKS

Since a gray level picture can be seen as a surface in three dimensions, z being the gray level value, it is in principle possible to use the models described here to represent them. This could have use for shape comparison of these surfaces, but the authors have not experimented with this. The idea, anyway, is to use models with large ϵ (large error tolerance, coarse representation) to compare two surfaces; if the models are equal (in some appropriate sense, for instance, the quantized z values agree) then we could afford comparison with a smaller ϵ (more accurate representation). In this way the shape similarity between any two 3-dimensional surfaces (or any two gray level pictures) can be ascertained. A related paper (Bribiesca and Guzmán, 1978a) develops this idea fully for two-dimensional flat regions (binary pictures). The largest problem with this approach is to find a normalization procedure (the basic rectangle of (Bribiesca and Guzmán, 1978b)) that will produce a unique model for the 3-d case: it is easier to compare canonical models.

The method described in this paper is currently being implemented and tested for representation of topographic surfaces formerly described by their contour lines.

Merging of models into a larger model. If four adjacent surfaces *a*, *b*, *c*, *d* are represented by models $\underset{\sim}{a}$, $\underset{\sim}{b}$, $\underset{\sim}{c}$, $\underset{\sim}{d}$, the model of the joint surface (a,b,c,d) is formed by creating a new frame 1 (cf. Fig. 5) which has as non-terminal pointers M_{NW}, M_{NE}, M_{SW} and M_{SE}, pointers to the frames 1 of $\underset{\sim}{a}$, $\underset{\sim}{b}$, $\underset{\sim}{c}$ and $\underset{\sim}{d}$.

Significant points vs. correlation points. The significant points (also called surface-specific points (Peucker, et. al., 1976) are those points of the terrain where slope changes in an important way. The points that a correlation routine finds in an easy manner, based for instance in the two pictures of a stereo pair, are called "correlation points:" they are points that are easy to correlate in the pictures, because the gray levels in their neighborhood are quite different from others, hence they can be identified rapidly and unmistakably. But they will not necessarily fall on top of "significant" points.

The components of the model. The model so far described and its construction can be seen as formed by:

- a tessellation of polygons (Gómez, 1978) (triangles in this case);
- an accuracy criteria, which tells whether a polygon of the model needs further refinement (in our case, comparison of modelled vs. real heights, cf. section "When to stop refining');
- a procedure to refine the model (in our model, select a significant point near the middle point of a side);
- a manner to store the model (as exemplified in Fig. 6);
- a way to access the model (as seen in section 'Data Retrieval for Surface Reconstruction');
- a method to reconstruct the surface from the model (this is given by the procedure *height* (a,b,c,x,y) evaluated at the appropriate triangle *a b c* which contains the point $(x,y,0)$; the appropriate definition of containment is embodied in procedure *inside* (a,b,c,x,y), which takes into account, for instance, the "flaps" of Fig. 7).

Suggestions for further work

1. Refer to Fig. 7. Do not use k_1 = 10% for the width of the sidewalks. Compute instead the maximum distance that $(r_x, r_y, 0)$ can be from r'' for the enlarged

triangle $a\ d\ s$ to meet still the error tolerance ε. This has to do with average slopes of the triangles.

2. Refer to section "Simplified storage for cover of similar triangles". Fully develop the model that uses the representation of each triangle as a string of pairs of binary digits, v.gr., triangle $g\ h\ j$ = 00 00 01 (the son N of the son M of the son M of the rectangle).

3. Do not retrieve the triangles from the root of the tree (cf. section "Data retrieval for surface reconstruction") but store them so as to access them by a double binary search on the coordinates of the vertices (Gómez).

4. Consider the methods of this paper and of (Bribiesca and Avilés, 1974; Bribiesca and Guzmán, 1978a) as similar procedures that address data representation at arbitrary accuracy levels, and use them for shape comparison.

ACKNOWLEDGEMENTS

Andrew Clement and T. Peucker gave the triangular idea; Renato Barrera contributed to the concept of a hierarchy of triangles and other good advices. Abel Carreño and Angel García Amaro, of CETENAL, gave good photogrammetric advice. T. Radhakrishnan kindly revised the manuscript.

Work herein reported was partially done under the Joint Research Agreement (IX-1976) between CETENAL and UNAM.

REFERENCES

Bribiesca, E. and Avilés, R. 1974. Codificación en cadenas y técnicas de reducción de información para mapas y dibujos lineales. IBM Latin American Scientific Center (Mexico City). Informe CCAL-74.

Bribiesca, E. and Guzmán, A. 1978a. Shape description and shape similarity measurement for two-dimensional regions. Submitted to Fourth International Joint Conference on Pattern Recognition. Kyoto, Japan.

Bribiesca, E. and Guzmán, A. 1978b. Shape numbers: a notation to describe pure form and to measure resemblance and difference in shape. Computer Science Dept., IIMAS, National University of Mexico. Report PR-78-18.

Gómez, D. Modelos digitales del terreno de precisión variable. B.S. Thesis, Facultad de Ciencias, Universidad Nacional de México (in preparation).

Gómez, D. 1978. Tessellation of triangles of variable precision as an economical representation for DTM's. Proceedings of the Digital Terrain Models Symposium St Louis, Mo. Available from American Society of Photogrammetry.

Guzmán, A. 1971. Analysis of curved line drawings using context and global information. In Machine Intelligence VI, (D. Michie and B. Meltzer, eds) University of Edinburgh Press. Chapter 20.

Horn, B.K.P. 1970. Shape from shading: a method for obtaining the shape of a smooth opaque object from one view. Ph.D. Thesis, E.E. Dept., M.I.T. Project MAC Technical Report MAC-TR-79.

Jensen, H.H. 1976. Collaboration in Physics within the Nordic countries. Europhysics news 7, 5, pp 1-4.

Klinger, A. and Nikitas, A. Picture decomposition, tree data structures, and identifying directional symmetries as node combinations. Computer Graphics and Image Processing (to appear).

Peucker, T., Fowler, R.J., Little, J.J. and Mark, D.M. 1976. Triangulated irregular networks for representing three-dimensional surfaces. Simon Fraser University, Burnaby, Canada. Technical Report # 10.

Signor, G. and Nadler, M. 1978. Une application de la corrélation numérique d'images: la stéréophotogrammetrie automatique. Congres AFCET/IRIA, Reconnaissance des Formes et Traitement des Images. Paris.