M.E.T.U. Journal of the Faculty of Architecture
Volume 6, Number 1, Spring 1980.

GRID AND POLYGON REPRESENTATIONS IN COMPUTER-AIDED ARCHITECTURAL DESIGN

Bülent ÖZGÜÇ

Received March, 1980

GRID AND POLYGON REPRESENTATIONS

There are various ways to store architectural data graphically in a computer. According to Mitchell[1] these can be classified as:

1. Regular grid
2. Hierarchical array[2]
3. Dimensionless representation of solids and voids[3]
4. Dimensionless representation of spaces[4]
5. Polygon[5]
6. Dual graph[6]
7. Smith diagram[7].

Even though the above could be used for three-dimensional representation, computer graphics and computer vision scientists prefer to use the following classification for dynamic and static representation[8]:

1. 3-D surface points
2. Polygonal network
3. Curved surface patches
4. Quadric patches
5. Cellular space
6. Polyhedra
7. Geometric form
   Ellipsoids
   Cylinders
   Spheres

According to the shape of an object that will be stored in a computer and the geometric transformations that might be required for it, some representation techniques may be more suitable than the others. Certain approximations of shape can also be introduced provided that they are justified within accuracy versus transformations relation. The cost

1. W.J.MITCHELL, *Computer-Aided Architectural Design*, New York: Petrocelli/Charter, 1977, pp.155-221.

2. C.M.EASTMAN, Representations for Space Planning, *Communications of the ACM*, vol.13, no.4, April 1970, pp.242-250.

3. L.A.MARCH, Boolean Description of a Class of Built Forms, *The Architecture of Form*, ed. L.March, Cambridge, England: Cambridge University Press, 1976.

4. P.STEADMAN, The Automatic Generation of Minimum Standard House Plans, *EDRA 2*. eds. J.Archea, C.Eastman, Pittsburgh, Pennsylvania: Carnegie Mellon University, 1970.

5. W.M.NEWMAN and, R.F. SPROULL, *Principles of Interactive Computer Graphics*. New York: McGraw-Hill, 1973.

6. P.STEADMAN, Graph Theoretic Representation of Architectural Arrangements, *The Architecture of Form*. ed. L.March, Cambridge, England: Cambridge University Press, 1976.

7. R.L.BROOKS, *et al.* The Dissection of Rectangles into Squares. *Duke Mathematical Journal*, vol.7, 1940, pp.312-341.

8. N.BADLER and, R.BAJCSY, Three-Dimensional Representations for Computer Graphics and Computer Vision, *Computer Graphics*, ed.R.L.Philips, SIGGRAPH '78 Proceedings, Atlanta, Georgia: The Association for Computing Machinery, August 23-25, 1978.

of representation, storage, and transformation operations is also a decisive factor on the choice of the technique to be used.

The decision for a particular representation technique is more complicated than it looks. In interactive graphics procedures that evolve in time or that have more than one level of accuracy or set of operation requirements, a single technique of representation might not be satisfactory. This introduces a complication that representation techniques are not necessarily compatible with each other. Some can be easily converted, some require tedious procedures, some need an intermittent conversion, and some simply cannot be converted. Computer graphics and computer vision scientists currently give much attention to this aspect[9].

9. N.BADLER and, R.BAJCSY, There Dimensional Representations for Computer Graphics and Computer Vision, *Computer Graphics*, ed. R.L.Philips SIGGRAPH '78 Proceedings, Atlanta, Georgia: The Association for Computing Machinery, August 23-25, 1978.

One of the most popular representation techniques in architecture, especially for Mies followers, is the regular grid. It also happens that, the easiest graphics technique both in storage and manipulation by a computer is coincidential with the Miesian one. The grid is stored as a 2 or 3-D matrix or sometimes as a string that can be chopped up in a given pattern to obtain rows and columns.

There are numerous types of grids which are discussed in detail by Mitchell[10]. The reason why so many types of grids have been developed is basically to be found in the graphical restrictions that grids impose. Each complex figure is expected to be a multiple of a basic unit which is sometimes a rather distasteful limitation for the architect. On the other hand, it imposes a discipline and an ease of operation. To combine the benefits of grid with the 'beauty of artistic lines', architects have thus come up with a fleet of grid types.

10. W.J.MITCHELL, *Computer-Aided Architectural Design*. New York: Petrocelli/Charter, 1977, pp.161-183.

We shall limit our discussions in this paper to regular grids in which each cell is the same shape and size throughout. Hierarchical arrays in which a cell could be divided into smaller cells, perhaps can be included in this discussion but their application is dependent to the availability of programming languages such as EULER[11]. We can derive a rather straight forward hypothesis that each object in space and its two-dimensional projection on a plane can be expressed in terms of a regular grid if the size of the cells are accordingly adjusted. The limiting case of this hypothesis will be when each cell is reduced infinitecimally in size so that every point on the object is addressable. Even if this can be proved to be theoretically correct, its practical application is impossible. Since no known computer has 'infinite memory', the grid approach is bound to be approximate to a certain extend depending on the topology of the subject matter.

11. N.WIRTH and, H.WEBER, EULER: A Generalization of Algol and its Formal Definition. *Communications of the ACM*, vol. 9, nos.1 and 2, 1966, pp.13-25 (no.1) and 89-99 (no.2).

Another widely used representation technique is the polygon. Even if not directly analogical, polygon representation is an opposite to grids. Instead of spatial domains (cells), boundary elements (lines or vectors) are defined in this representation. There is no rule as to the storage technique of vectors composing a shape and much attention has been given to this matter[12]. Polygon representation gets considerably complicated if spatial domains need to be explicitly defined.

To make the best use of the two techniques discussed above, a method has been developed[13] which will be discussed in the remaining part of this paper. Two software systems, MAP and DESINT[14] will be mentioned. The former one uses grid representation and can perform conversions from polygon to grid. It is written for a Digital Equipment Corporation PDP/8 based mini-computer system. DESINT, on the other hand, uses polygon representation and was developed for IBM and UNIVAC mainframe computers. The two, however, can share the same input data that can be generated by COMGRAPH.MENU[15] and input utility routines of MAP. COMGRAPH.MENU is not an integral part of the other two systems but it is an interactive graphics utility package that generates sharable files compatible with both MAP and DESINT. An external computer-aided design algorithm, DYLAM II[16] will also be referred to. MAP system is capable of generating input data for DYLAM II as well as graphically produce the results on a storage tube. A final graphics utility package known as ISOPLOT[17] will be mentioned. MAP can also generate input files for this routine which produces isometric plots with hidden lines removed on grid layouts with respect to the z-coordinates of cells. The discussion of the systems will be limited to their syntax.

12. R.WILLIAMS, Survey of Data Structures for Graphics. *Computing Surveys*, vol.3, no.1, March 1971, pp.1-21.

13. B.ÖZGÜÇ,"An Integrated Systems Approach to Computer-Aided Architectural Design and Graphics Communications." Unpublished Ph.D. Thesis, Philadelphia, Pennsylvania: University of Pennsylvania, 1979.

14. These systems are discussed in detail in B.ÖZGÜÇ, "An Integrated Systems Approach to Computer-Aided Architectural Design and Graphics Communications," Unpublished Ph. D.Thesis, Philadelphia, Pennsylvania: University of Pennsylvania, 1979.

15. C.TINAZTEPE, COMGRAPH.MENU: A Set of FORTRAN Callable Graphics Routines, Computer-Aided Design Laboratory, Moore School of Engineering, Philadelphia, Pennsylvania: University of Pennsylvania, 1978.

16. D.SEADER and, S.GRAVA, A Demonstration of the Computer-Aided Land-Use Modelling for Regional Service System Design. Proceedings of the 6th Annual Urban Symposium, New York: Association for Computing Machinery, 1971.

17. C.HALL, ISOPLOT. The NPL Report (NAC 75). Sections 2 and 4. New York: NPL, 1975.

INTERACTIVE INPUT

The concepts of grid and polygon representations are best discussed on an example. The definition of an arbitrary site is input to the computer through COMGRAPH.MENU, utilizing the Tektronix 4014-1 tube in the Computer-Aided Design Laboratory, Moore School of Engineering, University of Pennsylvania. Figure 1 is the result of this operation. The command menu on the right side of the screen is activated by the cursors of 4014-1, and after the attached electronic tablet is initiated, the plan is simply traced in. The data generated from this interaction is in polygon form where curves are composed of short vectors. Figure 2 illustrates the way this data is generated and stored. The structure is a simple sequential list and for many applications, it might not even be necessary to store vector numbers (v) and node numbers (N1 and N2). Furthermore, two coordinates for each vector might not be necessary. P2's can be replaced by 0 or 1 standing for pen-up or pen-down respectively between the first coordinates of two sequential vectors. However, these reductions will damage interaction capabilities, especially for design applications where vectors (and sometimes even nodes) might be assigned characteristics, as will be explained soon.

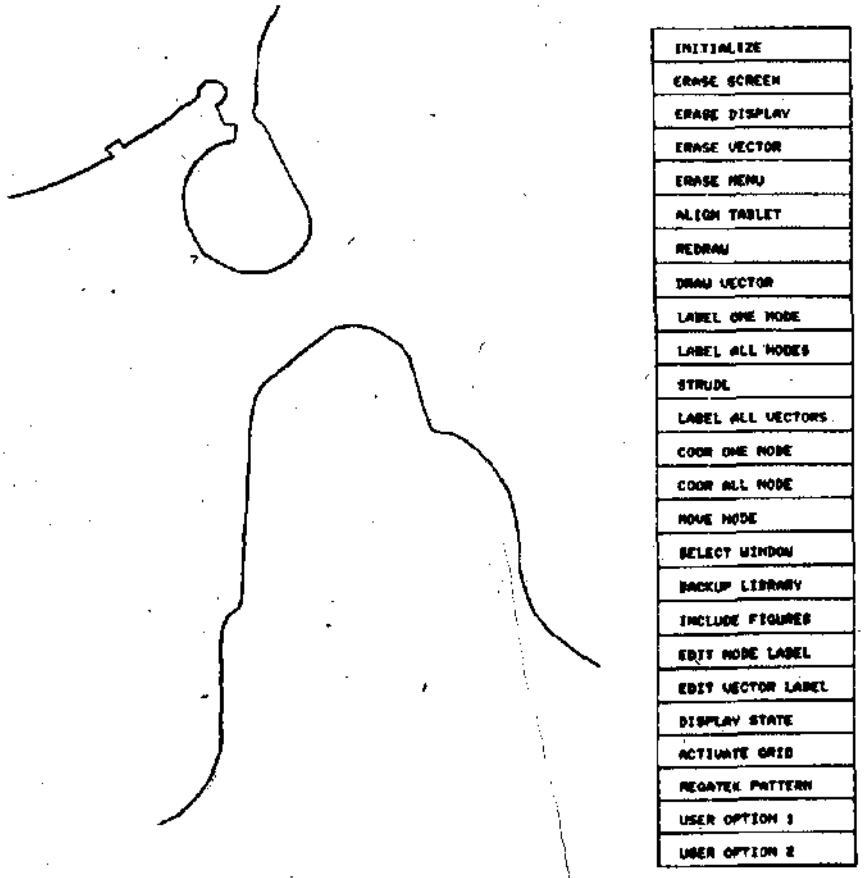
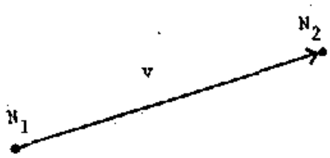


Figure 1. Input to COMGRAPH.MENU using a digitizer tablet and a storage tube. The table on far right is the graphics menu for action selection.



v = vector number
N_1 = node number (starting)
N_2 = node number (ending)
P_1 = coordinates of node 1
P_2 = coordinates of node 2

vector	N_1	N_2	P_1	P_2

Figure 2. A data structure for storing vectors.

Figure 3 is the same map, but this time it is regenerated by MAP sharing the already digitized data. The display is transferred to a Tektronix 4010-1 terminal with the menu removed, and the data is stored on the RK05 cartridge disk of the DEC PDP/8e[18].

Using graphics utility routines of MAP, the input is expanded. Figure 4 is generated by inserting four more nodes to the existing data and Figure 5 is the completed input. The reason for framing the map is in order to be able to assign characteristics to the enclosed areas. An improvement on the earlier data structure (Figure 2) is given in Figure 6. In this case, the vectors can also specify spatial domains, as the case of differentiating land from sea in the example. In order to obtain a grid approximation of this study area, the enclosed map is

18. DEC and PDP are registered trade marks of Digital Equipment Corporation.

Figure 3. Regeneration of input plan on MAP system through the files generated by COMGRAPH.MENU.

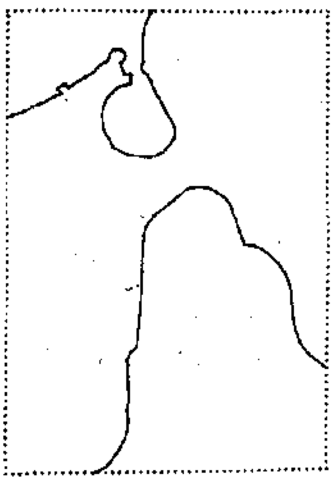


Figure 4. Insertion of four more nodes to the input map in order to enclose the study area.

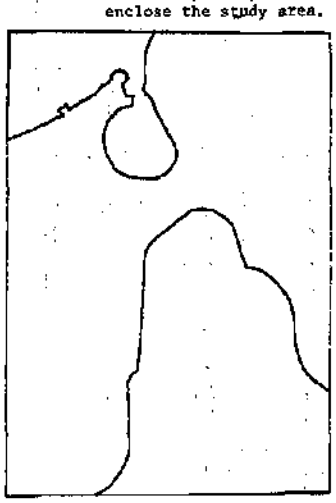


Figure 5. The completed input.

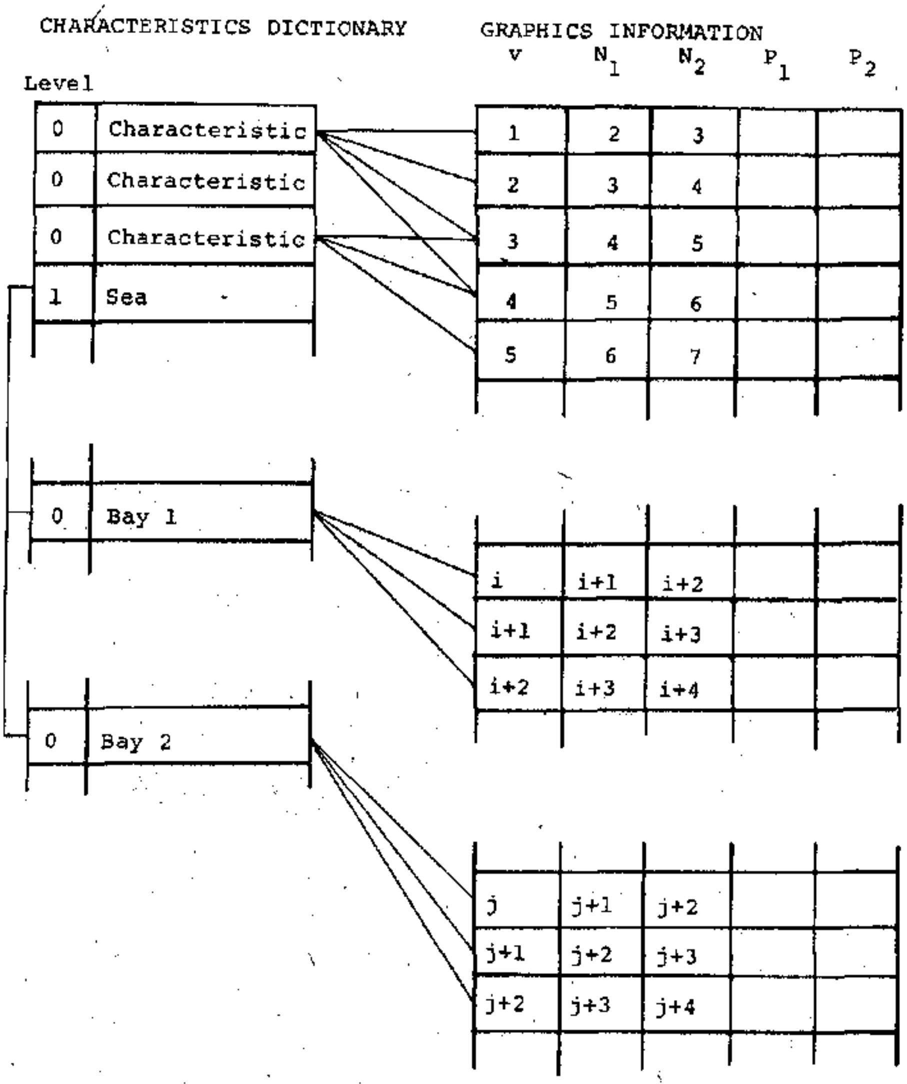


Figure 6. Data structure for assigning characteristics to vector areas.

interactively superimposed by square grids. A satisfactory superimposition is given in Figure 7. The reasons for grid conversion will be explained later. Once the map is satisfactorily superimposed by a grid, vectors are traced and their intersections with the cells are calculated by interpolation. Cells that have more area in land are given the value 1, the rest being assigned 0. The removal of 0 valued cells is illustrated in Figure 8. By tracing the grid first in x-direction, then in y-direction and only plotting the boundaries where there is a change from 1 to 0 (or 0 to 1), Figure 9 is obtained, which is an orthogonal approximation of the original input.

One of the main reasons for this grid conversion was to prepare data for an automated space allocation program known as DYLAM II, but at this time we shall limit the discussions to graphics. Another benefit of grid conversion is in the ease of generating contour maps. Instead of assigning 1 to those cells that are charecterized as land, we shall code the assignments so that the altitudes of the cells will be included.

Transferring the new data to the graphics program ISOPLOT, Figures 10 and 11 are generated which have their hidden lines removed with respect to x and y-axes. The isometric drawings have been generated on a Versatec 1200A electrostatic printer-plotter. Further improvements on the land coding make it possible to store various other characteristics such as land value, plantation, etc. Figures 12 and 13 are perspective drawings of one of the characteristics of the site without the hidden lines removed. This kind of representation is particularly useful in techniques employing overlays. The cells that have the highest number of 'desirable characteristics' will

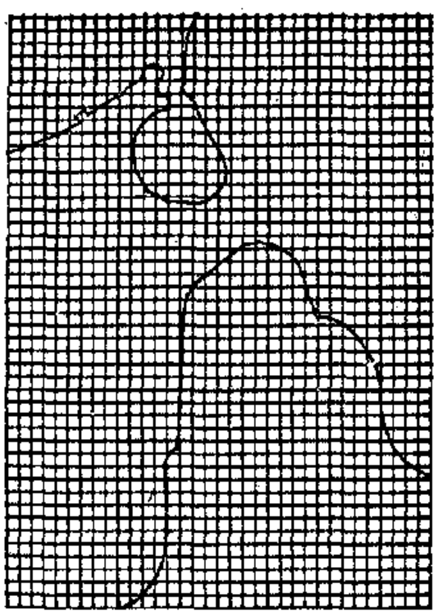


Figure 7. Interactive superimposition of square grid on the study area.

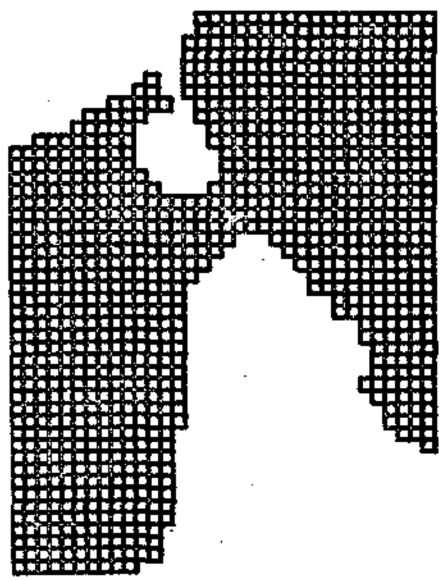


Figure 8. Orthogonal approximation of the input.

be displayed as the tallest locations. The original input is never destroyed through the process and can always be superimposed on the grid approximation for better visualization. However, to obtain isometric plots from discrete geometric data is considerably more difficult and not necessarily more accurate. Especially if the procedures require intersections, unions, differences, etc. of spatial domains for various overlays, it is a very tedious procedure to do such calculations using polygon data.

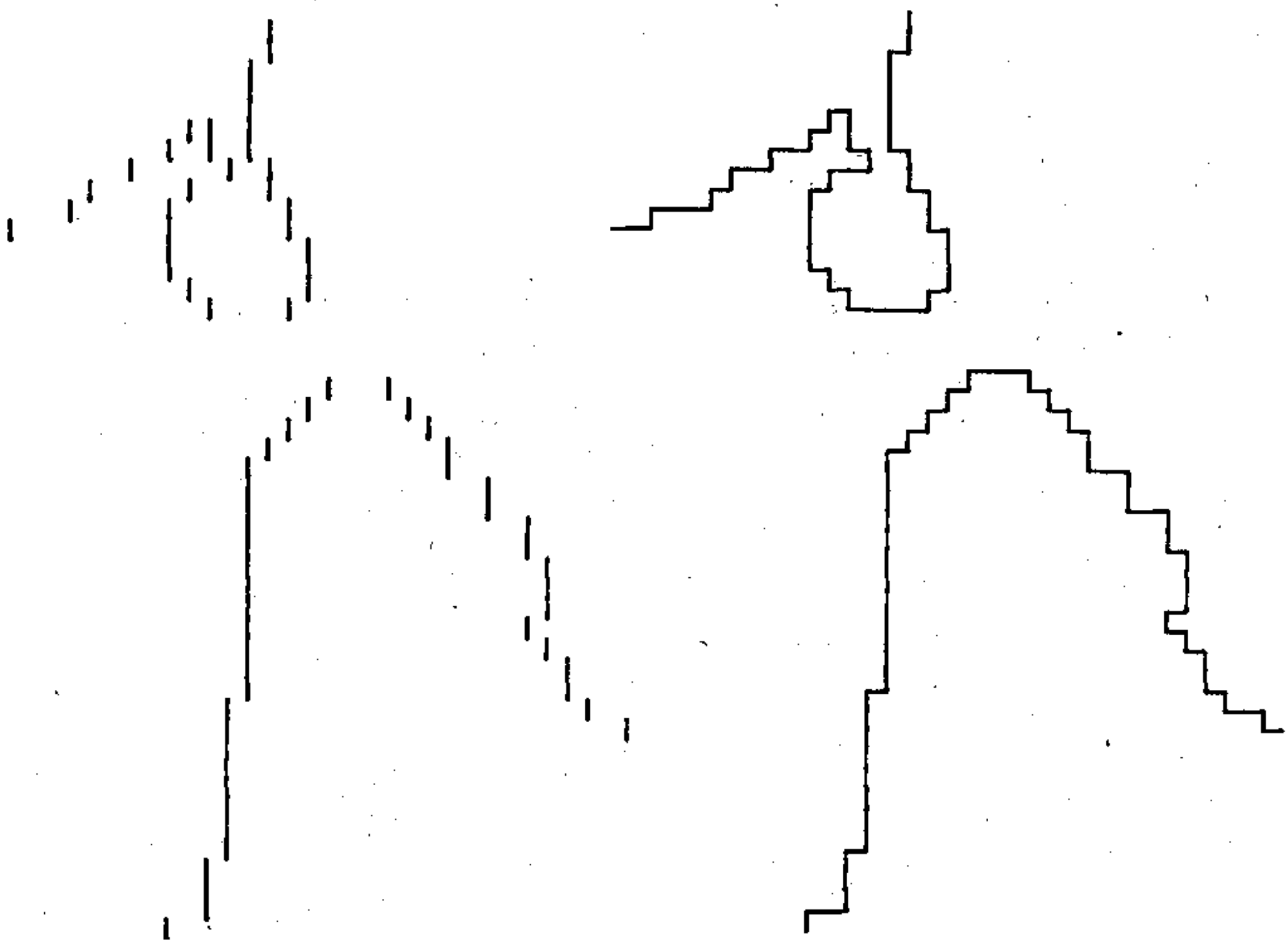


Figure 9. Removal of grids and leaving the orthogonal boundaries.

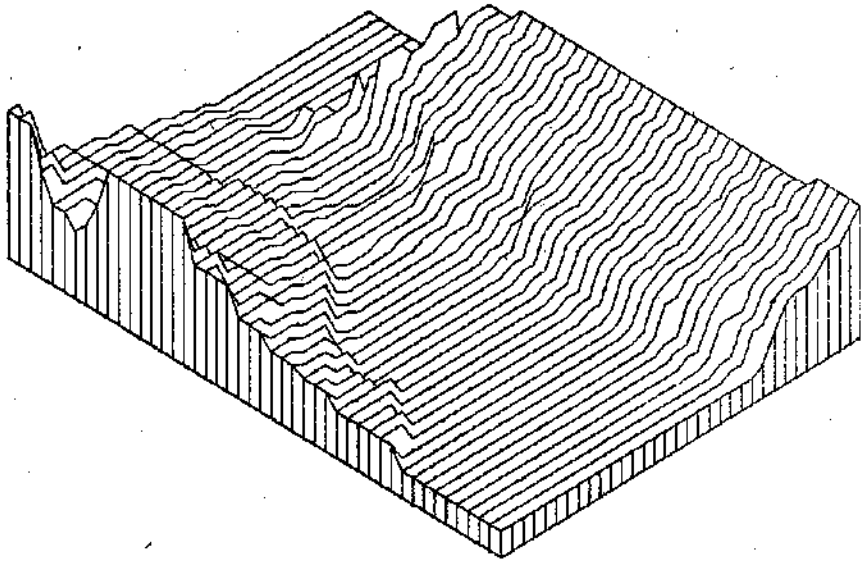


Figure 10. Isometric plot of the study area, traced with respect to x-axis.

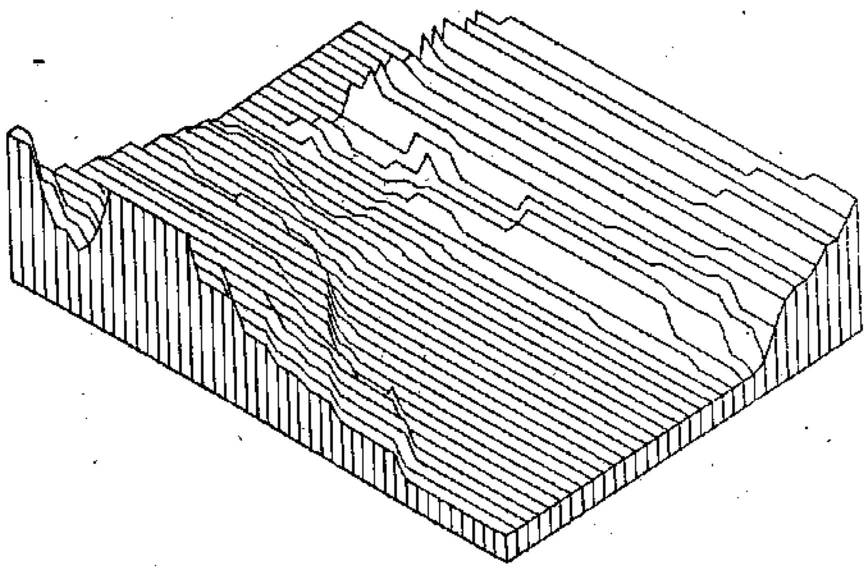


Figure 11. Isometric plot of the study area traced with respect to y-axis.

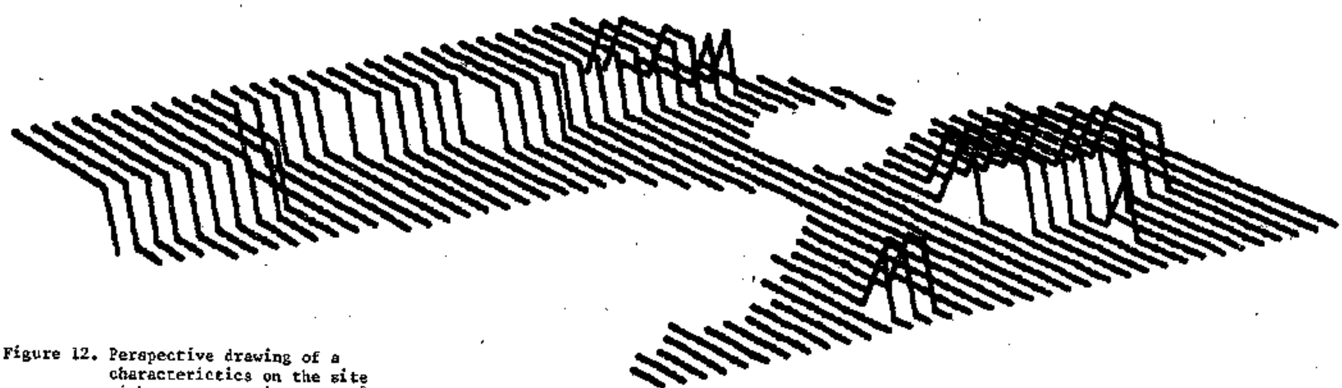


Figure 12. Perspective drawing of a characterictics on the site with x,y,z rotations of 45^0, -50^0 and 240^0respectively.

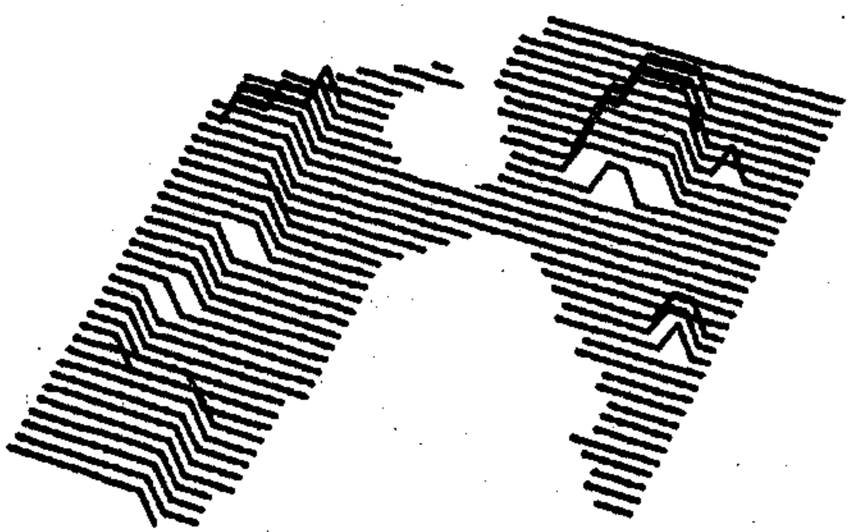


Figure 13. Perspective drawing of the same characteristic with 25^0, -15^0 and 245^0 rotations. (see Figure 12).

ITEMS, GROUPS, AND FUNCTIONS

During computer-aided design, shapes or lines are not used to generate impressive pictures, but for a synthesis of the components to serve a purpose. As soon as the word *component* is mentioned, a meaning is assigned to a particular graphics item. Characteristics of land lots as have been mentioned earlier, is an example for this. We shall define item as the smallest complete component of the system. Groups will be composed of items. A group can also be individually addressed like the items composing it.

In MAP, the concept of an item is fairly simple: each cell is an item. Cells that have similar properties form a group, such as land lots having similar characteristics. MAP can perform various functions on items (or groups). The most obvious of these is the input/output that can be done at various levels. The second one is the graphical transformations that are either simple in nature (eg., scaling, translation) or more complex such as the perspective or isometric map generations as have been previously illustrated. A final group of functions relate to numerical analysis, statistical procedures, topological operations, and set theoretical operations that are used in various design functions. These will not be explained here in detail, but Figure 14 illustrates the flow of the system in general. It is important to note here that the item, characteristic, and operation specifications, together with how they are specified bring in the issue of a meta-language and man-machine communication. This subject will be discussed at a later date, under seperate cover.

In DESINT, that uses polygon representation, definition of an item is more arbitrary. Elements are represented by vectors, therefore any one-coordinate vector (point), 2-D or 3-D vectors, or their combination could form an item. These explicit graphics definitions are handled in three modes by DESINT:

1. Digitized standard items in data blocks.
2. Procedural definitions for generating otherwise difficult-to-digitize items.
3. Interactive input for user-defined items.

The additions to Figure 14 are not in functional sense, but with respect to technical differences. First of all, characteristics as well as item names point to discrete graphical definitions as opposed to cell locations. Since some of the items might be rather complex in nature, they are identified by a frame around them termed a box. This simplifies set operations (such as intersections) considerably. Secondly, unlike grid cells, the individual items can be modified graphically. Therefore, in addition to characteristic changes, graphical changes are also introduced to the input and modification routines. The functions mentioned for MAP also apply to DESINT with the added disadvantage of complexity. Set theoretical operations on polygon representation have been discussed by Eastman and Yessios[19]. Numerical operations for fitting irregular polygons to fixed areas have been discussed by Weinzapfel and Negroponte[20]. Graphical operations are similar but the transformation procedures are technically different.

19. *C.M.EASTMAN and, C.YESSIOS*, An Efficient Algorithm for Finding the Various Intersections and Differences of Spatial Domains. Pittsburgh, Pennsylvania: Carnegie-Mellon University, Institute for Physical Planning, Research Report, 1972.

20. G.E.WEINZAPFEL and, N.NEGROPONTE, Architecture-by-Yourself, in *Computer Graphics*; ed. U.W.Pooch, Proceedings of SIGGRAPH '76, Philadelphia, Pennsylvania: Association for Computing Machinery, 1976, pp.74-78.

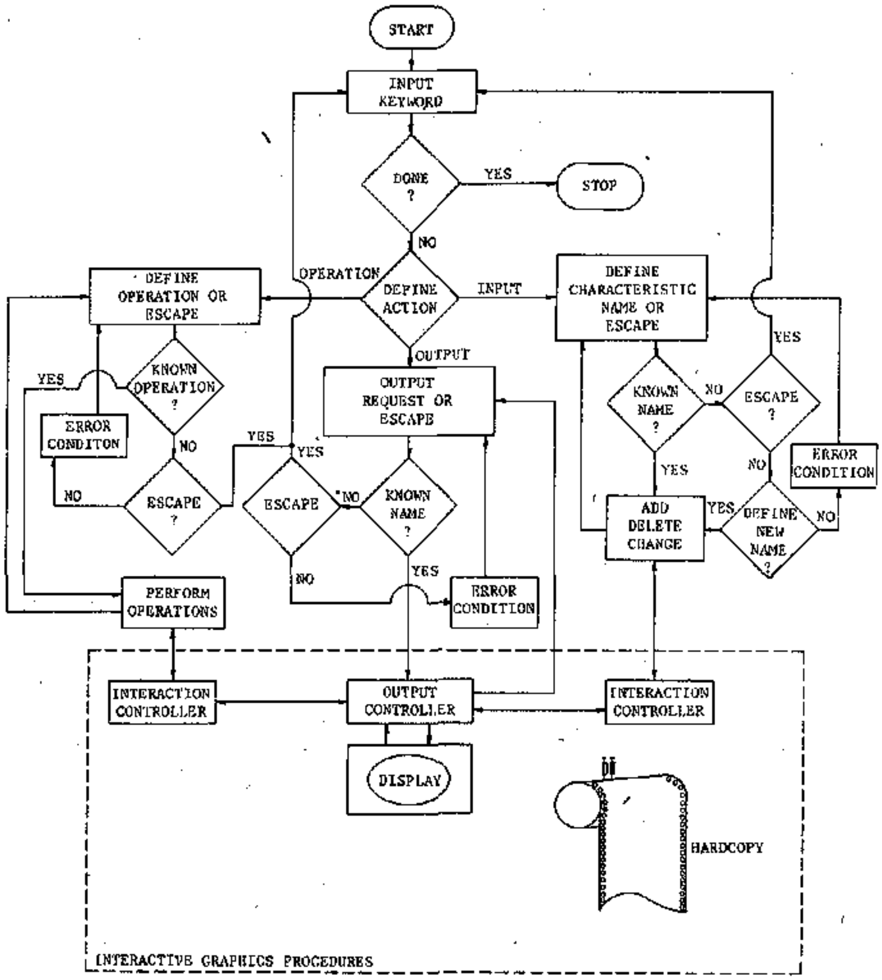


Figure 14. Flow diagram of the system.

SYSTEM INTEGRATION

As have been mentioned a few times, there are advantages and disadvantages of grid and polygon representation techniques. An attempt has been made to combine MAP and DESINT in such a way that one can utilize both techniques wherever needed. The example in the paper in fact is an illustration for the integration of the two. Basic definitions are kept in various files on a disc. These files are related to each other by means of various pointers. The seperation of graphics information from the non-graphics one is in order to protect the meanings of items from being misplaced when conversions are performed. These conversions are performed by MAP. DESINT will generate files on polygon representation only, as well as interacting with the existing ones.

DESINT will also 'recognize' various items in order to generate color perspectives with hidden lines removed. This will be discussed in a following paper. The user is assumed to know the keywords related to the two techniques, handle the cursors of the Tektronix scope, use the pen of the associated tablet, and generate his own vocabulary of item definitions and characteristics. The system will then make it possible to generate an evolving graphics data base that has a set of associated attributes (characteristics), modify and perform transformations on it, handle some design oriented activities, and produce input for other computer-aided architectural design (CAAD) systems in the form of graphical and relational data. The results of CAAD procedures (either automated or interactive) in return modify the existing files and relations thus produce new statistical information, graphical output and relation pattern.

CONCLUSIONS

This approach is believed to demonstrate one of the requirements of CAAD in that the user must be given a choice of his graphic representation as well as define his own vocabulary within given limits. Although the procedures described here are not necessarily easy to program and produce some redundancies, they can save enormous amount of computer time. This is especially true for some operations that can be performed easily on particular representation techniques, but with great difficulty on others. Similar procedures can be applied to other compatible techniques.

The integration of MAP and DESINT brings in the idea of an meta-language. An earlier example of such an integrated system is URBAN-5[21].

The difficulties encountered during the development basically relate to a complicated data structure and numerous interrelated files. Thus, there are a number of cumbersome garbage collection routines, but this was justified as a trade off between the generality as opposed to computer time and disk space.

21. URBAN-5 is discussed in various books and articles by N.Negroponte of which we shall quote two:
N.NEGROPONTE, *The Architecture Machine*, Cambridge, Mass.: The MIT Press, 1970, pp.71-93.
N.NEGROPONTE, URBAN-5: An Experimental Urban Design Partner, in M.Murray (ed.), *Computer Graphics in Architecture and Design*, Proceedings of the Yale Conference on Computer Graphics in Architecture, New Haven: Yale University, 1969.

BİLGİSAYAR YARDIMI İLE MİMARLIK TASARIMINDA IZGARA VE ÇOKGEN BETİMLERİ

ÖZET

Mimarlıkta genellikle yedi tür bilgisayar ile çizge tekniği kullanılmaktadır. Üç boyutlu bilgisayar çizgesi ile uğraşan bilim adamları, bu teknikleri daha değişik biçimde, on başlık altında toplamaktadırlar. Bu yazıda, bunlardan iki tanesi, ızgara ve çokgen çizim tekniklerinin bilgisayar kullanarak mimarlıkta uygulamaları tartışılmaktadır.

Izgara yaklaşımı, fazla duyarlık gerektiği zaman aşırı bilgisayar belleği gerektirmekte, ancak kullanılması kolay olup, mekansal özellikleri rahatlıkla tarif edebilmektedir. Çokgenler ise, hemen hemen her şekildeki çizimi duyarlıkla ifade edebilmekte, ancak sadece çeper gösterdiklerinden mekansal işlemlerde zorluklarla karşılaşmaktadırlar.

Mimarlıkta bilgisayar uygulamaları yapan kişiler, çizge özelliklerini araştırıp, kendi gereksinmelerine uygun olan bir veya birkaç tanesini seçme olanaklarını düşünmelidirler. Birden fazla teknik kullanıldığı zaman, bunları birbirlerine dönüştürmekte gerekebilir. Bu bazen kolay, bazen zor, bazen de olanaksızdır. Yazıda, ızgara ve çokgen yaklaşımlarının bu yönde bir incelenmesi yapılmış, çokgen çizimlerin ızgaraya dönüştürülmesi ve ikisinin aynı anda kullanılması örneklenmiştir.

BIBLIOGRAPHY

BADLER, N. and BAJCSY, R. Three-Dimensional Representations for Computer Graphics and Computer Vision. *Computer Graphics*, ed R.L.Philips, SIGGRAPH '78 Proceedings, Atlanta, Georgia: The Association for Computing Machinery, August 23-25, 1978

BROOKS, R.L. *et al.* The Dissection of Rectangles into Squares. *Duke Mathematical Journal*. vol.7, 1940, pp.312-241.

EASTMAN, C.M. Representations for Space Planning. *Communications of the ACM*. vol.13, no 4,April 1970, pp.242-250.

EASTMAN, C.M. and YESSIOS, C. An Efficient Algorithm for Finding the Various Intersections and Differences of Spatial Domains. Pittsburgh, Pennsylvania: Carnegie-Mellon University, Institute for Physical Planning, Research Report, 1972.

HALL, C. ISOPLOT. The NPL Report (NAC 75). Sections 2 and 4, New York: NPL, 1975.

MARCH, L. A Boolean Description of a Class of Built Forms. *The Architecture of Form*, ed. L.March, Cambridge, England: Cambridge University Press, 1976.

MITCHELL, W. *Computer-Aided Architectural Design*. New York: Petrocelli/Charter, 1977.

NEGROPONTE, N. URBAN-5: An Experimental Urban Design Partner. in *Computer Graphics in Architecture and Design*. ed. M.Murray, Proceedings of the Yale Conference on Computer Graphics in Architecture. New Haven: Yale University, 1969.

NEGROPONTE, N. *The Architecture Machine*. Cambridge, Massachusetts: The MIT Press, 1970.

NEGROPONTE, N. and GROISSER, L.B. URBAN-5: A Machine that Discusses Urban Design. *Emerging Methods in Environmental Design and Planning*. ed. G.T.Moore, Cambridge, Massachusetts: The MIT Press, 1970.

NEWMAN, W.M. and SPROULL, R.F. *Principles of Interactive Computer Graphics*. New York: McGraw-Hill, 1973.

ÖZGÜÇ, B. "An Integrated Systems Approach to Computer-Aided Architectural Design and Graphics Communications." Unpublished Ph.D. Dissertation, Philadelphia, Pennsylvania: University of Pennsylvania, 1979.

SEADER, D. and GRAVA, S. A Demonstration of the Computer-Aided Land-Use Modelling for Regional Service System Design. Proceedings of the 6th Annual Urban Symposium, New York: Association for Computing Machinery, 1971.

STEADMAN, P. The Automatic Generation of Minimum Standard House Plans. *EDRA Two*, eds. J.Archea and C.Eastman, Proceedings of the 2nd EDRA Conference. Pittsburgh, Pennsylvania: Carnegie Mellon University, 1970.

TINAZTEPE, C. COMGRAPH.MENU. A set of FORTRAN callable graphics routines written for the Computer-Aided Design Laboratory of Moore School of Engineering, Department of Civil Engineering. Philadelphia, Pennsylvania: University of Pennsylvania, 1978.

WEINZAPFEL, G.E and NEGROPONTE, N. Architecture-by-yourself. *Computer Graphics*, ed. U.W.Pooch, Proceedings of SIGGRAPH '76, Philadelphia, Pennsylvania: University of Pennsylvania, 1976, pp.74-78.

WILLIAMS, R. Survey of Data Structures for Graphics. *Computing Surveys*, vol.3, no.1, March 1971, pp.1-21.

WIRTH, N. and WEBER, H. EULER: A generalization of ALGOL and its Formal Definition. *Communications of the ACM*, vol.9, nos.1 and 2, 1966.