論文
2-3-4

Reconstruction of 3D Topography from Contour Line Data using Artificial Neural Networks
(신경회로망을 이용한 등고선 데이터로부터 3차원 지형 복원)

김 수 선*
(Su-Sun Kim)

ABSTRACT

We propose an algorithm which can reconstruct the 3D information from geographical information. The conventional techniques, the triangular patches and the Random Fractal Midpoint Displacement (RFMD) method, etc., have often been used to reconstruct natural images. While the RFMD method using Gaussian distribution obtains good results for the symmetric images, it is not reliable on asymmetric images immanent in the nature.

Our proposed algorithm employs neural networks for the RFMD method to present the asymmetrical images. By using a neural network for reconstructing the 3D images, we can utilize statistical characteristics of irregular data. We show that our algorithm has a better performance than others by the point of view on the similarity evaluation. And, it seems that our method is more efficient for the mountainous topography which is more rough and irregular.

요 약

본 논문에서는 지리 정보로부터 3차원 지형 정보로 복원하는 알고리즘을 제안한다. 자연스러운 3차원 지형의 표현을 위해 기존의 복원 알고리즘들은 삼각형 패치 기법이나 랜덤 프랙탈 중간점 변위기법들을 이용하여 복원하였다. 이와 같이 가우시안 분포를 사용한 랜덤 프랙탈 중간점 변위 방법으로 복원한 결과는 좌우대칭인 이미지에 대해서는 자연스럽게 표현되지만, 자연에 내재된 비대칭인 지형의 모습을 표현하기에는 어려움이 있었다.

좌우비대칭인 지형을 좀 더 자연스럽게 표현하기 위해 기존의 랜덤 프랙탈 중간점 변위 방법에 신경회로망을 이용하여 복원하는 알고리즘을 제안한다. 신경회로망의 학습결과를 3차원 지형 복원에 이용함으로써 비정형적인 데이터가 갖는 통계적 특성을 활용한다. 제시된 알고리즘의 우수성은 복원된 결과의 근사도 평가로 보인다. 또한, 신경회로망 학습 결과를 이용하여 산악지형과 평탄지형에 대하여 실험하고 실험 결과 산악지형에 대한 적용 예가 더 효과적임을 보인다.

1. Introduction

Traditionally 2D data are used for most of the existing Geographic Information Systems (GIS) and 3D data are used only for some specific regions. It is because 2D data can be easily obtained from either a map or a hand operated work, but difficult for 3D data to be constructed [1].

* 정회원 : 한양여자대학 전산정보계열 부교수

논문접수 : 2001. 3. 8.
심사완료 : 2001. 3. 30.

However, it is very important to construct 3D data because 3D GIS data are necessary in practical fields such as in a missile route plan and a selection of expected new construction area. This paper proposes an improved algorithm that constructs 3D geographic data from the existing map data.

There are three existing methods in reconstructing 3D terrains from contour line information of a map: spline interpolation and linear interpolation by using regular grid data, contour line [2,3,4], spline surface interpolation by using triangulation [5], and interpolation by applying fractal theory [6,7].

These methods can be classified into computational complexity, size of memory used, information loss resulted from mesh composition, and reality of reconstructed topography based on the evaluation of reconstructing algorithm. In order to represent naturally irregular 3D terrains there are articles focusing on the reality by using random fractal without any limitations on computational complexity, and the size of memory used [6, 7, 8, 9, 10].

This study proposes the method of the reconstructing asymmetric terrain because it is closer to the original ones more than the general terrain. The size of memory used and time necessary for reconstructing it is not considered, but it does focus on the approximation.

We implement the algorithm in the following steps. First, we perform a polygonal approximation which extracts the feature points from the contour line data. Second, we obtain the triangular patches from the approximated polygonal contours and rectify them to reflect the characteristics of contour map on an actual mountainous region with complex ridges and valleys. Third, we propose the algorithm to construct 3D shapes from the triangular patches, using the Random Fractal Midpoint Dis acement (RFMD) method and the neural networks method which reflect the statistical characteristics of irregular shapes in data. The previous algorithms [6,7,8,9,10] do not naturally represent the shape of asymmetric nature because they use RFMD by Gaussian distribution. In this paper, Artificial Neural Network (ANN) is implemented into RFMD in order to represent natural asymmetric shapes in the real world [11,12]. The purpose of this work is to make the reconstructed shapes to be close to the real shapes. Statistical characteristics of irregular objects and phenomena can be well applied by using ANN learning and 3D reconstruction. The results of implementation on mountainous terrains and flat terrains using the ANN learning effect show that the proposed method is more effective for mountainous terrains.

Section 2 presents the construction methods of polygonal approximation and triangular patch that belong to the preprocess of this study. An algorithm to obtain 3D data based on interpolation using both fractal modeling and ANN learning is proposed in section 3. Section 4 presents the experimental results of the algorithm and reality evaluation of results. The experimental results also show that the proposed algorithm is more effective in actual mountainous terrains. The summary of this paper is included in section 5.

2. Polygonal Approximation and Triangular Patch

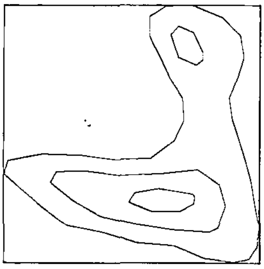
As raw data to reconstruct 3D terrain, a contour line is used. This section summarizes the process of extracting geographical characteristics data and triangular patch.

2.1 Polygonal Approximation

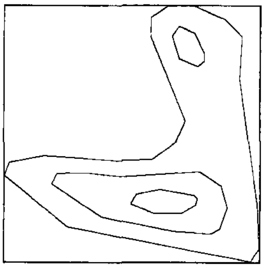
In this paper, coordinate and grid data of a contour line are raw data. The coordinate data are the raw data for reconstruction, and the grid data are ANN input data to estimate reconstructed shapes.

One of the objectives of this study is to condense the amount of data saved on the natural phenomena to be represented. The random fractal can be produced in detail through the process of amplifying the data. Thus, we perform a polygonal approximation that extracts the feature points from the contour line data, using Roberge's algorithm [13]. The input parameter radius of algorithm which is extracted from the feature point differs based on the contour line height. And the radius value resulted in a smaller parameter value for a higher contour line rather than a lower one. Eventhough the rate of condensing the data is different based on threshold value of each feature point extraction, the threshold value is controlled subjectively more for the purpose of reflecting characteristics of the intended reconstruction form than merely for the effect of condensing it.

[Fig. 1]-(a) represents a raw input contour line and (b) shows a polygonal approximated contour line.



(a) Input Contour Line



(b) Polygonal Approximated Contour Line

[Fig. 1] Extracted Feature Points of Contour Line

2.2 Structuring Process of Triangular Patches

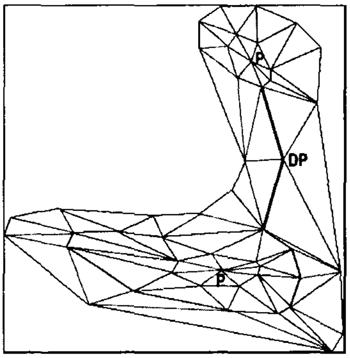
2.2.1 Case I: Structuring Triangular Patches For Two Contour Lines Having No Canyons

First, we start with a peak point of contour. The P in [Fig. 2] is the peak point. Then, a nearest contour line surrounding the peak point is selected. Sample out points from the contour line and connect these points with the peak point. To produce triangular patches, repeat the above steps for all peak points. Second, select another contour line that directly outlines the previously selected contour line. Connect all the sample points of outer line to those of inner line, which are shortest in distance to the sample points of the outer line. This will mostly produce triangular patches, but some other polygons, too. The polygons can be divided into a set of triangles. In polygons, connect all the points diagonally. Then select the shortest diagonal lines. This process will divide polygons into a set of triangles. Repeating the above steps will complete triangular patches.

2.2.2 Case II: Structuring Triangular Patches For Contour Lines Having Canyons

A canyon is defined when a progress angle changes from an acute angle to an obtuse angle. After the canyon is detected, draw the dummy contour line by finding middle points of the canyon. The middle point of the canyon is average of inner products of the points in the neighborhood of the canyon. Connecting the middle point with the closest contour line will form a line and this is defined as "dummy contour line". The DP in [Fig. 2] is the dummy point and middle point of the canyon. Bold line in [Fig. 2] illustrates the example of the dummy contour line. The height of canyon's center point is the average of two neighbor contour lines. A triangle patch is formed by connecting the center point of the canyon and the lower contour line feature points. Complete triangular patches are applied in case I to the dummy contour line.

Triangular patching is completed through case 1 and 2. [Fig. 2] illustrates the example of finial state of triangular patching.



[Fig. 2] Triangular Patches

3. 3D Data Interpolation

3.1 Fractals and ANNs: A Stochastic Terrain Model

Perhaps the most common natural phenomenon which can be represented by computer graphics would be terrain. Since terrain is generally characterized by randomly distributed features that are recognizable by their overall properties as opposed to the specific macroscopic features, its strong stochastic properties make it a good choice for the application of a stochastic model.

A recurrent problem in generating real pictures by computers is to represent natural irregular objects and phenomena. We develop a powerful solution to this computer graphics problem by modeling objects as sample paths of stochastic processes. Of particular interest are those stochastic processes which have been previously found to be useful models of the natural phenomena to be represented. One such model applicable to the representation of terrains, known as "fractional Brownian motion," has been developed by Mandelbrot [14].

The fractal algorithm does not naturally represent the shape of the asymmetric nature because it uses the RFMD by Gaussian distribution. In this paper, ANN is imported to RFMD to naturally represent the asymmetric shapes of the real world.

In order to use a neural network to this application, our study has focused on neural networks for classification with input patterns that are linearly separable. Networks have been able to acquire experiential knowledge during the supervised training process. The experiential knowledge acquisition has been based on the convergent training of single-layer discrete perceptron networks, which can adjust their weights incrementally in order to achieve correct classification of linearly separable sets of patterns.

To this point, we have studied networks that use a linear combination of inputs with weights being proportionality coefficients. Such networks work with the argument of the nonlinear elements simply computed as a scalar product of the weight and input vectors.

For training patterns that are linearly separable, the modification could involve either a departure from the concept of the linear discriminant function or a major change in network architecture. Typical nonlinear discriminant functions are chosen to be quadratic or piecewise linear. The piecewise linear discriminant functions can be implemented by a single-layer linear network employing perceptrons [15]. However, our chosen architecture will be that of the multilayer network. Each layer of a multilayer network is composed of a linear network; it is based on the original concept of the linear discriminant function.

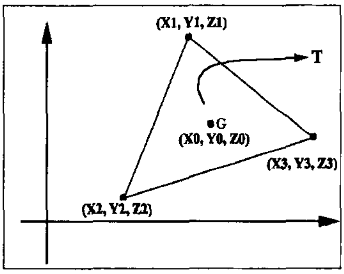
This standard class of neural networks architecture can approximate virtually any multivariable function of interest, sufficiently provided that many hidden neurons are available. The studies of Funanashi, Hornik, Stinchcombe, and White prove that multilayer feedforward networks perform as a class of universal approximators [16,17]. The results also provide a fundamental basis for establishing the ability of multilayer feedforward networks to learn the connection strengths that achieve the desired accuracy of approximation.

3.2 Learning Pattern

For the approximation of a topographic characteristics function, backpropagation neural networks composed of an input layer, hidden layer and output layer are used. The neural network inputs are calculated as follows. Let T be the patched triangle and defined by

$$T= \{P_1(x_1, y_1, z_1),\ P_2(x_2, y_2, z_2),\ P_3(x_3, y_3, z_3)\} \quad (1)$$

[Fig. 3] showing triangle T and height value G at the center of T.



[Fig. 3] Triangle T and Height Value G

Learning data are obtained from T and G. In order to get the features of the patched triangle to reflect the terrain's characteristics, its steepness, width, tilt and length are defined. Steepness is the angle between horizontal plane and triangle. Width is the angle between two sides which faces the base of the triangle. Tilt is the angle between the base of the triangle and the tangent line of the lowest point of the triangle. The length refers to the base of the triangle.

1) Steepness : $S=\tan^{-1}\{\frac{(z_1-z_2)}{\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}}\}$ (2)

2) Width :

$$W=\cos^{-1}\{\frac{(x_2-x_1)(x_3-x_1)+(y_2-y_1)(y_3-y_1)+(z_2-z_1)(z_3-z_1)}{|Z12|\ |Z13|} \quad (3)$$

where $|Z12|=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2+(z_1-z_2)^2}$

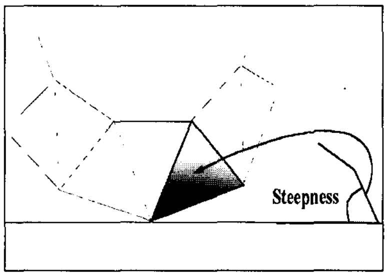
$|Z13|=\sqrt{(x_1-x_3)^2+(y_1-y_3)^2+(z_1-z_3)^2}$

3) Tilt : $T=\tan^{-1}\frac{(y_2-y_3)}{(x_2-x_3)}$ (4)

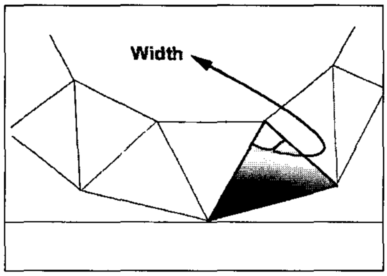
4) Length : $L=\frac{\sqrt{(x_3-x_2)^2+(y_3-y_2)^2+(z_3-z_2)^2}}{N}$ (5)

where N is a normal constant.

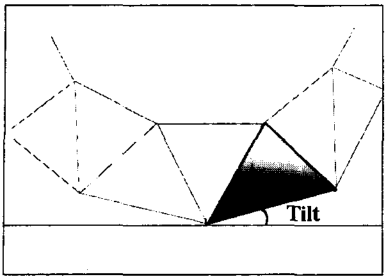
[Fig. 4] shows the feature data defined in equations (2) through (5).



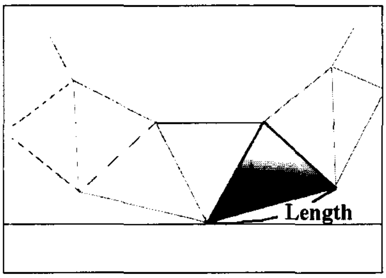
(a) Steepness



(b) Width



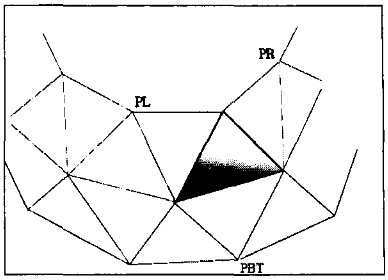
(c) Tilt



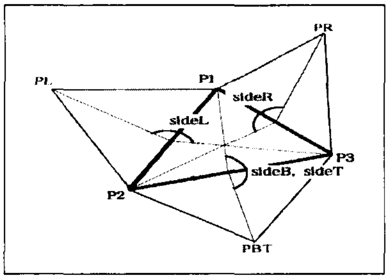
(d) Length

[Fig. 4] Feature Data I

For concrete feature data, equations (6), (7) and (8), are defined with the angles of intersection between the patched triangle and the triangles adjacent to the patched one. L is meaning left side of the patched triangle, R is right side of it, BT is bottom or top side of it. Mid is the middle point on the adjacent edge.



(a) Relationship of Adjacency Triangular Patches



(b) Details in Relationship of Adjacency Triangular Patches

[Fig. 5] Feature Data II

1) Left_adjacency: (6)

$$side\_L = \cos^{-1}\frac{(x_3-x_{mid})(x_L-x_{mid})+(y_3-y_{mid})(y_L-y_{mid})+(z_3-z_{mid})(z_L-z_{mid})}{\sqrt{(x_3-x_{mid})^2+(y_3-y_{mid})^2+(z_3-z_{mid})^2}\sqrt{(x_L-x_{mid})^2+(y_L-y_{mid})^2+(z_L-z_{mid})^2}}$$

2) Right_adjacency: (7)

$$side\_R = \cos^{-1}\frac{(x_2-x_{mid})(x_R-x_{mid})+(y_2-y_{mid})(y_R-y_{mid})+(z_2-z_{mid})(z_R-z_{mid})}{\sqrt{(x_2-x_{mid})^2+(y_2-y_{mid})^2+(z_2-z_{mid})^2}\sqrt{(x_R-x_{mid})^2+(y_R-y_{mid})^2+(z_R-z_{mid})^2}}$$

3) Base/Top_adjacency: (8)

$$side\_BT = \cos^{-1}\frac{(x_1-x_{mid})(x_{BT}-x_{mid})+(y_1-y_{mid})(y_{BT}-y_{mid})+(z_1-z_{mid})(z_{BT}-z_{mid})}{\sqrt{(x_1-x_{mid})^2+(y_1-y_{mid})^2+(z_1-z_{mid})^2}\sqrt{(x_{BT}-x_{mid})^2+(y_{BT}-y_{mid})^2+(z_{BT}-z_{mid})^2}}$$

[Fig. 5]-(a) shows triangles adjacent to the patched triangle and (b) depicts the adjacency in detail.

3.3 Learning Algorithm

We define the output of neural network in the following equations:

$$y_i = f\left(\sum_{l=1}^{L} u_{il} v_l\right) \tag{9}$$

$$o_k = g\left(\sum_{i=1}^{L} w_{ki} y_i\right) \tag{10}$$

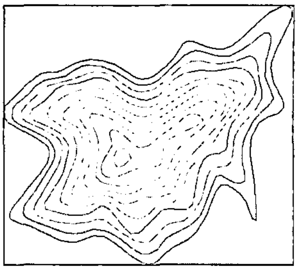
u_{il} denotes weight from l'th neuron input layer to i'th neuron hidden layer. w_{ki} denotes weight from i'th neuron hidden layer to k'th neuron output layer. y_i, o_k denotes the value of hidden layer and output layer respectively. Activation function f(x) should be a non-linear function, it is sigmoid. g(x) is a function which maps feature data to a desired output. It is a first-order function which increases monotonically. The learning algorithm used is an erroreous backpropagation learning rule. To improve the learning time, a momentum method is adapted to the learning.

4. Implementation and Reality Evaluation

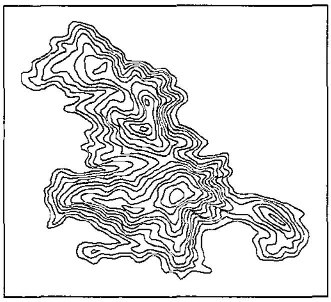
Input data for reconstruction are both contour map and grid data. The flat topography in [Fig. 6]-(a) and mountainous topography in figure 6-(b) are used for implementation.

As input data for mountainous topography, contour lines are divided at an interval of 10m. A section between 110m and 230m is selected as the height of topography. The grid data matrix size is 157 rows by 139 columns.

In the case of flat topography, contour lines are divided at an interval of 5m. A section between 360m and 405m is selected as the height of topography. The grid data matrix size is 111 rows by 117 columns. For mountainous topography, 1548 feature points are extracted from 2579 raw data. For flat topography, 410 feature points extracted from 513 raw data are selected.



(a) Input Contour Map(Flat Topography)



(b) Input Contour Map(Mountainous Topography)

[Fig. 6] Input Contour Line

The polygons which approximate the contour lines are patched with triangles. The observation is based on the patched triangles. The number of learning triangles is 72 for flat topography, and 288 for mountainous topography.

The structure of the learning neural network is simple. It has one input layer, one hidden layer and one output layer. Input parameters of the ANN are composed with 7 neurons: steepness, width, tilt, length, and three neighbours. The desired output is the height of the centroid of the triangle and the output layer has one neuron. The number of neurons in the hidden layer differs; 15 neurons for flat area and 16 neurons for a mountainous area. The learning rate is a constant that determines the efficiency of the learning process. The learning parameters are 0.1 and 0.01 respectively. <Table 1> shows input parameters for learning in flat topography and mountainous topography.

<Table 1> Input Parameters

(a) Flat Topography

ANN Parameters	Set Value
Number of Layers (Hidden+Output)	2
Number of Input Neurons	7
Number of Training Patterns	72
Number of Hidden (1) Neurons	16
Number of Hidden (2) Neurons	0
Number of Output Neurons	1
Learning Parameter	0.1
Momentum Parameter	0.7
Maximum Error	2.0
Parameter of a Sigmoid Function	1.5

(b) Mountainous Topography

ANN Parameters	Set Value
Number of Layers (Hidden+Output)	2
Number of Input Neurons	7
Number of Training Patterns	288
Number of Hidden (1) Neurons	15
Number of Hidden (2) Neurons	0
Number of Output Neurons	1
Learning Parameter	0.01
Momentum Parameter	0.7
Maximum Error	10
Parameter of a Sigmoid Function	2.5

<Table 2> is parts of learning data for flat and mountainous topography.

<Table 2> Learning Data

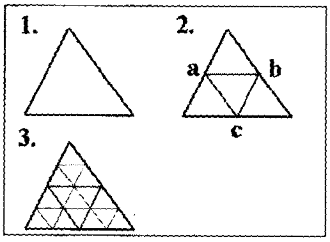
(a) Flat Topography

Steepness	Width	Tilt	Length	Side_L	Side_R	Side_BT	Bias	Height
0.324	0.800	1.249	1.296	2.107	1.010	1.285	-1.000	4.227
0.430	1.058	1.456	1.095	1.106	1.989	2.626	-1.000	2.453
0.282	0.964	0.581	1.479	2.001	1.536	0.432	-1.000	2.080
0.206	0.330	1.494	1.609	2.295	1.811	1.573	-1.000	3.148
0.487	0.696	0.759	0.758	1.428	1.174	0.703	-1.000	2.689
0.325	1.280	1.421	1.605	1.908	1.802	0.508	-1.000	1.445
0.122	0.459	0.351	0.694	0.000	0.994	0.000	-1.000	4.319
0.335	0.124	0.803	0.594	0.878	1.165	1.534	-1.000	3.891
0.245	1.123	1.347	0.757	0.000	1.210	0.000	-1.000	3.767
0.359	0.524	0.694	0.893	0.000	2.234	0.000	-1.000	2.152

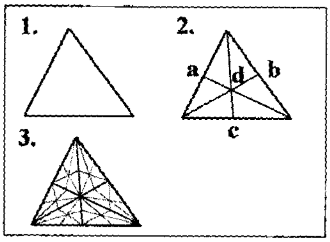
(b) Mountainous Topography

Steepness	Width	Tilt	Length	Side_L	Side_R	Side_BT	Bias	Height
0.426	0.910	0.398	2.102	2.295	1.811	1.573	-1.000	3.702
0.281	1.133	0.358	2.178	1.428	1.174	0.703	-1.000	4.160
0.316	0.795	1.055	3.262	1.908	1.802	0.508	-1.000	5.757
0.272	1.579	0.173	2.299	0.000	0.994	0.000	-1.000	7.497
0.369	0.856	0.689	3.898	0.878	1.165	1.534	-1.000	5.614
0.363	0.629	1.046	2.239	0.000	1.210	0.000	-1.000	3.480
0.417	1.452	0.588	1.654	2.489	2.234	1.469	-1.000	2.072
0.282	1.004	1.411	3.060	2.090	2.103	1.871	-1.000	4.538
0.432	1.106	0.543	3.029	1.469	1.097	2.405	-1.000	7.123
0.481	0.938	1.499	2.147	2.305	1.461	0.976	-1.000	6.589

Each triangle can be subdivided into four smaller triangles by connecting the midpoints of the sides of the triangles. The original triangle is a fractal triangle whose irregular surface consists of many smaller triangular facets. This process continues with smaller triangles until the desired resolution is obtained, resulting in a fractal quadrilateral whose surface is composed of many smaller quadrilateral facets. The traditional RFMD processing unit and the newly proposed processing unit are shown in [Fig. 7]-(a) and (b), respectively. Each height at the midpoints of the sides of the triangles is obtained from displacement of the RFMD. The height of the center of the triangle is obtained by the learning algorithm.



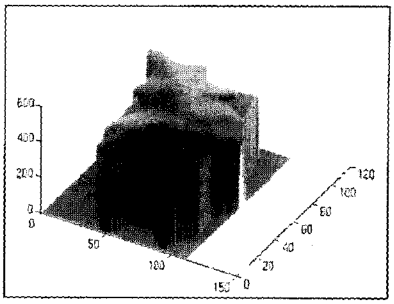
(a) The Traditional RFMD Processing Unit



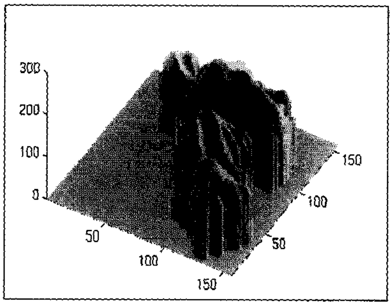
(b) The Newly Proposed Processing Unit

[Fig. 7] Basic Triangles for Reconstruction

[Fig. 8] shows the results of the three different reconstruction methods by interpolation. [Fig. 8]-(a) is a surface produced using grid data method, and (b) is a surface produced using the fractal method. [Fig. 8]-(c) is a surface produced using the proposed method.

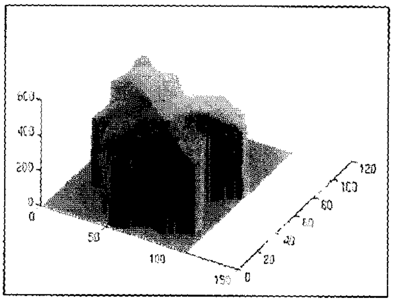


① Flat Topography

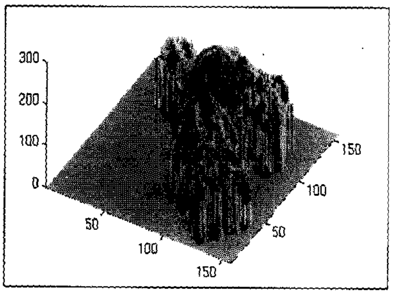


② Mountainous Topography

(a) Representation by Grid Data Method

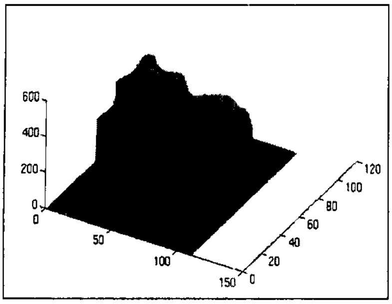


① Flat Topography

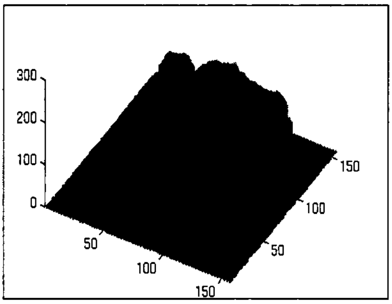


② Mountainous Topography

(b) Reconstruction by Fractal Method



① Flat Topography



② Mountainous Topography

(c) Reconstruction by Proposed Method

[Fig. 8] Results of Three Different Reconstruction Methods

Reality evaluation using SNR(Signal-to-Noise Ratio) and MSE(Mean Square Error) is performed between the raw grid data and extracted grid data obtained from the reconstructed terrains, and the results are shown in <Table 3>.

<Table 3> Results of Reality Evaluation

(a) Flat Topography

Method	SNR(dB)	MSE
Fractal Method	8.80	29.45
Proposed Method	9.18	27.29

(b) Mountainous Topography

Method	SNR(dB)	MSE
Fractal Method	6.01	43.62
Proposed Method	9.51	29.17

The results show that the proposed algorithm is more efficient in the reconstruction of mountainous topography that possessed irregular characteristics than the RFMD method.

5. Conclusion

We have proposed an algorithm which can reconstruct a 3D terrain from contour line data. We performed a polygonal approximation that extracts the feature points from the contour line data. We obtained the triangular patches from the approximated polygons and rectified them to reflect the characteristics of contour line on an actual mountainous region having complex ridges and valleys.

The proposed algorithm has been implemented with respect to flat topography and mountainous topography with asymmetry and randomness. This implementation was compared with the other existing method and is shown that the reconstruction of mountainous topography is shown to be more efficient in reality evaluation. By using the learning of neural networks, we can utilize the statistical characteristics of irregular data. We have shown that the algorithm has better performance than the others by the point of view from the similar evaluation.

The newly proposed learning algorithm requires additional learning time in comparison with the other traditional approaches. However, the learning is achieved separately based on typical data. When it is used after learning, the newly constructed algorithm does not require any excessive computation time and memory.

※ REFERENCES

[1] O. Guenther and A. Buchmann, "Research Issues in Spatial Databases," *ACM SIGMOD Record*, Vol. 19, No. 4, pp. 61-68, 1990.

[2] Jin-Seon Lee, Seung-Jong Chung, "A Raster- based Algorithm for Reconstruction of 3-D Terrain Data from The Contour Map," *Journal of KISS*, Vol. 22, No. 8, Aug. 1995.

[3] W. Barrett, E. Mortensen and D. Taylor, "Automated Height Information Acquisition from Topographic Map," *Proceedings of IAPR Workshop on Machine Vision Applications*, pp. 219-221, Nov. 1990.

[4] K. Cheng and M. Idesawa, "A Simplified Method of Data Form Conversion from Contour Line Surface Model to Mesh Surface Model," *Proceedings of IEEE International Conference on Pattern Recognition*, pp. 582-585, 1986.

[5] D. Meyers, S. Shelley, and S. Kenneth, "Surfaces From Contours," *ACM Transactions on Graphics*, Vol. 11, No. 3, pp. 228-258, July 1992.

[6] A. Fournier, D. Fussel, and L. Carpenter, "Computer Rendering of Stochastic Models," *Communication of ACM*, Vol. 25, No. 6, pp. 371-384, June 1982.

[7] A. D. Kelley, M. C. Malin, and G. M. Nielson, "Terrain Simulation using a Model of Stream Erosion," *In ACM SIGGRAPH*, Vol. 22, pp. 263-268, Aug. 1988.

[8] J. P. Lewis, "Generalized Stochastic Subdivision," *In ACM Transactions on Graphics*, Vol. 6, pp. 167-190, July 1987.

[9] F. K. Musgrave, G. E. Kolb, and R. S. Mace, "The Synthesis and Rendering of Eroded Fractal Terrains," *In ACM SIGGRAPH*, Vol. 23, pp. 41-50, July 1989.

[10] 安居院, 宫田, 中嶋, "3次元 自然形態の擬似符號化について," *グラフィックスとCAD*, Vol. 26, No. 2, 1986.

[11] Su-Sun Kim, Dong-Yoon Kim, Ha-Jin Kimn, "Reconstruction of 3D Shapes from Contour Line Data using The Backpropagation Neural Networks," *Journal of KISS(A): Computer Systems and Theory*, Vol. 23, No. 10, Oct. 1996.

[12] Su-Sun Kim, "*A Reconstruction of 3D Topography from The Contour Line Data*," Ph. D. Dissertation, Ajou University, Feb. 1997.

[13] K. Reumann and A.P.M. Witkam, "Optimizing Curve Segmentation in Computer Graphics," *in International Computer Symposium, A. Gunther, B. Levrat, and H. Lipps, Ed. New York: Elsevier*, pp. 467-472, 1974.

[14] B. Mandelbrot, *The Fractal Geometry of Nature*, W. H. Freeman and Company, 1982.

[15] N. J. Nilsson, *Learning Machines: Foundations of Trainable Pattern Classifiers*, New York: McGraw Hill Book Co., 1965.

[16] K. I. Funanashi, "On the Approximate Realization of Continuous Mappings by *Neural Networks*," Neural Networks 2: pp. 183-192, 1989.

[17] K. Hornik, M. Stinchcombe, and H. White, "Multilayer Feedforward Networks Are Universal Approximators," *Neural Networks* 2: pp. 359-366, 1989.

김수선

1984년 서울산업대학교
전자계산학과 (공학사)
1986년 한양대학교 산업대학원
산업공학과 (공학석사)
1997년 아주대학교 대학원
컴퓨터공학화 (공학박사)
1991년~현재 한양여자대학
전산정보계열 부교수
관심분야 : 컴퓨터 그래픽스,
멀티미디어 등