

Contour Map Reconstruction via Multi-module Parallel Computational Scheme

Miyuki Kawashima[†] Ryuji Tokunaga^{††} Yuzo Hirai^{††}

[†]Master's Degree Program in Scientific Technology,
University of Tsukuba, 1-1-1, Tennodai, Tsukuba, Ibaraki 305, Japan

^{††}Institute of Information Sciences and Electronics,
University of Tsukuba, 1-1-1, Tennodai, Tsukuba, Ibaraki 305, Japan

Abstract

Reconstruction of a set of imperfect contour curves which is extracted from a topographical map is one of the important problems in the field of geometrical information processing. This paper reports a novel multi-module parallel computational scheme to solve the reconstruction problem. The system recovers the three-dimensional ground configuration only from the imperfect contour curves, and then reconstructs the contour curves by taking the zero-crossing lines for the obtained configuration. The system performance is also shown by using a 1/25000 scaled topographical map.

I. Introduction

Recently, there are considerable interests in computerized map reading, since it enables us to construct map data base which can be used for broad social activities, e.g., natural resource assessment, regional planning, and traffic navigation systems. Here segregation/extraction of different features from a map is a very important processing. Since contour curves are interrupted by other types of map symbols, such as Chinese and Japanese characters, numbers, and road symbols, extracted curves are partially cut off and a couple of segments are lost. Therefore the method to reconstruct (interpolate or extrapolate) the lost segments is indispensable for a map information processing system. On this reconstruction problem, several approaches have been reported. All of them, however, pay no attention to the geometric structure which can be understood from contour curves. Hence their performance is qualitatively low.

In this paper, we would like to propose a novel multi-module parallel computational scheme, which recovers the three-dimensional ground configuration only from a set of imperfect contour curves, and reconstructs contour curves by taking the zero-crossing lines for the obtained configurations. Note that instead of the absolute heights, the relative heights between neighboring contour curves are important for reconstruction of imperfect contour curves. Also note that the concavity and convexity of reconstructed configuration does not matter for the reconstruction of contour curves. In order to obtain the relative height information, consistent gradient direction of contour curves must be determined, since they specify which contour curve is above and which contour curve is below. The proposed computational scheme consists of the following three procedures:

Procedure-1: Computation of gradient direction,

Procedure-2: Assignment of relative height to contour curves,

Procedure-3: Reconstruction of three-dimensional configuration.

In section II, three parallel processing modules which implement the above procedures are designed in the light of regularization and energy minimization techniques. In section III, in order to show the system performance, the computational scheme is applied to 1/25000 scaled maps published by Geographical Survey Institute of Ministry of Construction of Japan.

II. System Architecture

The proposed system consists of the following three modules which carry out the above procedures:

Module-1 : Computation of gradient directions,

Module-2 : Assignment of discrete height to contour curves,

Module-3 : Reconstruction of three-dimensional configurations.

In order to make our explanation clear, let us define several notations.

Definition-1: Digital image plane D and (i,j) -node.

Let D denote a two-dimensional digital image plane which is given by

$$D = \{(i,j) \mid 0 \leq i < \Delta L, 0 \leq j < \Delta L\}. \quad (1)$$

Let the element $(i,j) \in D$ be called the (i,j) -node.

Definition-2: Contour curves c_n and the tangent vector $\mathbf{k}_{i,j}$ of the (i,j) -node.

Using the chain-code^[1], n -th skeletonized contour curve on D is given by

$$[p_0(n), k_0(n), k_1(n), \dots, k_{M-1}(n)]$$

where $p_0(n) \in D$ and $\mathbf{k}_m(n) \in \{(0,\pm 1), (\pm 1,0), \pm(1,1), \pm(1,-1)\}$ denote the initial node and the m -th direction vector, respectively. Therefore the m -th point of c_n is located at (k,l) -node on D which is given by

$$(k,l) = p_0(n) + \sum_{m'=0}^{m-1} \mathbf{k}_{m'}(n).$$

The contour curves is redefined by the set of the above nodes :

$$c_n = \{p_0(n), p_0(n) + \sum_{m'=0}^0 \mathbf{k}_{m'}(n), \dots, p_0(n) + \sum_{m'=0}^{m-1} \mathbf{k}_{m'}(n)\}.$$

Let $\mathbf{k}_{k,l}$ be a normalized vector of $\mathbf{k}_m(n)$ which is called the tangent vector of the (k,l) -node. In the case that the (i,j) -node does not belong to a contour curve, set $\|\mathbf{k}_{i,j}\| = 0$.

Definition-3: Neighborhood ${}^n\Delta_{i,j}$ of the (i,j) -node.

Let ${}^n\Delta_{i,j}$ ($n=4,8,24,48$) denote the subset of D which is given by

$${}^n\Delta_{i,j} = \{(i+k,j+l) \mid 0 < |k|, |l| \leq \frac{\sqrt{n+1}-1}{2}\} \quad \text{for } n = 8,24,48$$

$${}^n\Delta_{i,j} = \{(i+1,j), (i,j+1), (i-1,j), (i,j-1)\} \quad \text{for } n = 4 \quad (2)$$

where $k, l \in \mathbb{N}$, and the symbol $\Delta_{i,j}$ indicates the neighborhood of the (i,j) -node.

Definition 4: Neighborhood ${}^n\Delta_{c_n}$ of the contour curve c_n .

Let ${}^n\Delta_{c_n}$ denote the subset of D which given by

$${}^n\Delta_{c_n} = \bigcup_{(i,j) \in c_n} {}^n\Delta_{i,j} \cup c_n. \quad (3)$$

where the symbol Δ_{c_n} indicates the neighborhood of the contour curve c_n .

2.1 Designing the module 1

Let us design the first module which computes a set of normalized gradient vectors denoted by $\{\mathbf{n}_{i,j} \in \mathbb{R}^2 \mid (i,j) \in D\}$. In the module, a set of tangent vectors denoted by $\{\mathbf{t}_{i,j} \in \mathbb{R}^2 \mid (i,j) \in D\}$ is computed in stead of gradient vectors. The main mechanism of the module is based on the energy minimization technique. Hence, a couple of cost functions are derived from the geometric properties which the contour map generally satisfies. Assume that :

Assumption-1 : Each of contour curves are sufficiently smooth,

Assumption-2 : There is no singular point in the neighborhood of contour curves,

Assumption-3 : A subset of tangent vectors $\{\mathbf{t}_{k,l} \mid (k,l) \in {}^4\Delta_{i,j}\}$ are sufficiently parallel with each other in the neighborhood of a ordinary point located at (i,j) .

In the case that the (i,j) -node *does not* belong to a contour curve, from assumption-3, the optimal value of $\mathbf{t}_{i,j}$ minimizes the following cost function :

$$E_1(i,j) = \sum_{(k,l) \in {}^4\Delta_{i,j}} (1 - \mathbf{t}_{i,j} \cdot \mathbf{t}_{k,l})^2. \quad (4)$$

On the other hands, in the case that the (i,j) -node belongs to a contour curve c_n , the associated cost function is modified. Recall the definition-2, and note that $\mathbf{t}_{i,j}$ must be parallel to $\mathbf{k}_{i,j}$. Using the variable $s_{i,j} \in \mathbb{R}^1$ whose magnitude is less than 1, set

$$\mathbf{t}_{i,j} = s_{i,j} \mathbf{k}_{i,j}. \quad (5)$$

Substituting (5) into (4), we obtain the following cost function :

$$E_2(i,j) = \sum_{(k,l) \in {}^4\Delta_{i,j}} (1 - s_{i,j} \mathbf{k}_{i,j} \cdot \mathbf{t}_{k,l})^2 \quad (6)$$

From assumption-1 and 2, the optimal value of $s_{i,j}$ minimizes $E_2(i,j)$. Furthermore, there must be a couple of (k,l) -nodes which belong to the same contour curve in the neighborhood of the (i,j) -node. From definition-1 and assumption-1, the sign of $s_{k,l}$ must be equal to that of $s_{i,j}$. Hence the optimal value of $s_{i,j}$ minimizes the following cost function as well as (6) :

$$E_3(i,j) = \sum_{(k,l) \in {}^2\Delta_{i,j}} \|\mathbf{k}_{k,l}\| (s_{i,j} - s_{k,l})^2. \quad (7)$$

Using these three cost functions, total energy of the module is defined by

$$E(i,j) = (1 - \|\mathbf{k}_{i,j}\|)E_1(i,j) + \|\mathbf{k}_{i,j}\| (E_2(i,j) + E_3(i,j)). \quad (8)$$

let $^x\mathbf{t}_{i,j}$ and $^y\mathbf{t}_{i,j}$ denote the first component and second component of $\mathbf{t}_{i,j}$, respectively. The differential dynamics of the module is obtained by applying the steepest decent method to the total energy function (8).

$$\frac{\partial^x \mathbf{t}_{i,j}(\tau)}{\partial \tau} = -\alpha \frac{\partial E(i,j)}{\partial^x \mathbf{t}_{i,j}} \quad (9.a)$$

$$\frac{\partial^y \mathbf{t}_{i,j}(\tau)}{\partial \tau} = -\alpha \frac{\partial E(i,j)}{\partial^y \mathbf{t}_{i,j}} \quad (9.a)$$

$$\frac{\partial s_{i,j}^*(\tau)}{\partial \tau} = -\alpha' \frac{\partial E(i,j)}{\partial \tau} \quad (9.b)$$

$$s_{i,j} = f(s_{i,j}^*)$$

where τ , α and α' denote time parameter and a pair of time constant, and where $f(\cdot)$ denote a threshold function which is given by

$$f(x) = \begin{cases} x & \text{for } |x| < 1 \\ \frac{x}{|x|} & \text{for } |x| \geq 1 \end{cases} \quad (10)$$

By selecting a contour curve c_0 arbitrarily, the initial conditions of $\mathbf{t}_{i,j}$ and of $s_{i,j}$ are set as follows:

$$\mathbf{t}_{i,j} = \mathbf{k}_{i,j} \quad \text{and} \quad s_{i,j}(0) = 1 \quad \text{for } (i,j) \in c_0, \quad (11.a)$$

$$\mathbf{t}_{i,j} = 0 \quad \text{and} \quad s_{i,j}(0) = 0 \quad \text{for } (i,j) \notin c_0. \quad (11.b)$$

Fig.1(a) shows the schematic drawing of c_n and initial conditions on D. A set of optimal values of $\mathbf{t}_{i,j}$ and $s_{i,j}$ are obtained by iterating the dynamics (9). (see Fig.1(b))

Remark : In the actual case, these assumptions sometimes does not hold, e.g., a skeletonized contour curve loose its smoothness, and a singular point appears in the neighborhood of a contour curve. In order to avoid these problems, another simple mechanism which adjusts the time constant α is applied to the module.

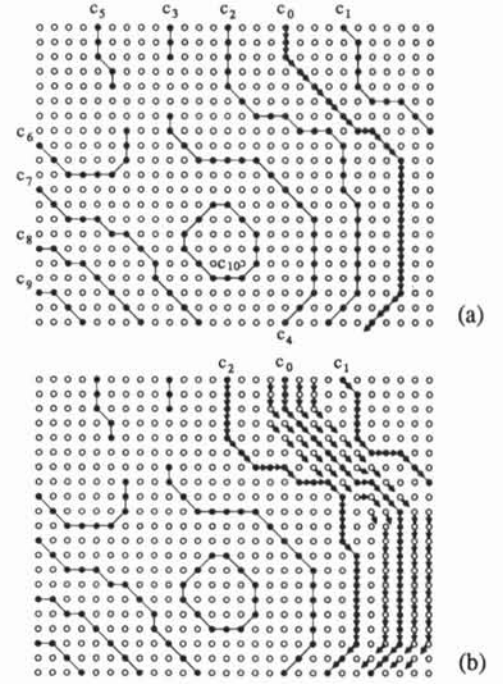


Fig.1 Computation of tangent vectors. (a) The schematic drawing of contour curves on D. The tangent vectors on c_0 indicate the initial conditions. (b) The way how tangent vectors are determined from c_0 .

B. Designing the Module 2

Next, let us design the second module which computes a set of *discrete height* which is denoted by $\{h_{i,j} \in \mathbb{Z}^1 \mid (i,j) \in D\}$. Simply speaking, computing process is as follows. Setting $h_{i,j} = 0$ of the (i,j) -node on c_0 , and propagating the discrete height information from c_0 to the entire map in accordance with the gradient directions, $\mathbf{n}_{i,j}$, which are given by

$$\mathbf{n}_{i,j} = (\frac{\partial^y \mathbf{t}_{i,j}}{\partial \tau}, -\frac{\partial^x \mathbf{t}_{i,j}}{\partial \tau}) / \|\frac{\partial \mathbf{t}_{i,j}}{\partial \tau}\|. \quad (12)$$

One can easily determine whether the (k,l) -node $((k,l) \in {}^8\Delta_{i,j})$ belongs to the "upward" or "downward" neighborhood of the (i,j) -node by computing the following value:

$$(\frac{k,l}{i,j} \cdot \mathbf{n}_{k,l}) - \theta \quad (13)$$

where

$$\frac{k,l}{i,j} = (k-i, l-j) / ((k-i)^2 + (l-j)^2)^{1/2},$$

and θ denotes a threshold value. If the above value is positive, the (k,l) -node is in the upward neighborhood of the (i,j) -node, otherwise it is in the downward neighborhood. The upward (respectively downward) propagation is governed by the following simple rules:

Propagation rule-1 : In the case that the (i,j) -node is not on any contour curves, if the (k,l) -node is in the downward (respectively upward) neighborhood of the (i,j) -node, then substitute the same discrete height of the (k,l) -node to that of the (i,j) -node.

Propagation rule-2 : In the case that the (i,j) -node and the (k,l) -node are on the same contour curve c_n , in comparison of their relative heights, the height of the c_n takes the larger value. On the other hand, if (k,l) -node is in the downward (respectively upward) neighborhood of the (i,j) -node, then add (respectively subtract) the constant integer value denoted by Δh to the height of the (k,l) -node, and the obtained value is substituted to the discrete height of the (i,j) -node.

The above two propagation rules are realized by a simple difference dynamics as is shown below.

Consider a control parameter of the process, $m \in [-1,1]$. In the case that $m=1$ (or $m=-1$), the upward (or downward) propagation is carried out. Using (13), **propagation rule-1** is realized by the following difference dynamics:

$$h'_{i,j} = \frac{1}{2}(m+1) \max_{(k,l) \in \delta_{\Delta i,j}} \{ \Theta^m[a_{k,l}, h_{k,l}], h_{i,j} \} + \frac{1}{2}(1-m) \min_{(k,l) \in \delta_{\Delta i,j}} \{ \Theta^m[a_{k,l}, h_{k,l}], h_{i,j} \} \quad (14)$$

where

$$a_{k,l} = \Psi(m^{(k,l)} p_{i,j} \cdot n_{k,l}) - \theta \Psi(\mu - lh_{k,l}),$$

and μ denotes a positive large number. $\Psi(\cdot)$ and $\Theta^m[\cdot]$ are a pair of threshold functions given by :

$$\Psi(x) = \begin{cases} 1 & \text{for } x > 0 \\ 0 & \text{for } x \leq 0 \end{cases} \quad (15)$$

$$\Theta^m[x,y] = \begin{cases} y & \text{for } x > 0 \\ -m\mu & \text{for } x \leq 0 \end{cases} \quad (16)$$

In the same manner, **propagation rule-2** is realized by the following difference dynamics:

$$h'_{i,j} = \frac{1}{2}(m+1) \max_{(k,l) \in \delta_{\Delta i,j}} \{ \Theta^m[b_{k,l}, h_{k,l}], \Theta^m[c_{k,l}, h_{k,l} + m\Delta h], h_{i,j} \} + \frac{1}{2}(1-m) \min_{(k,l) \in \delta_{\Delta i,j}} \{ \Theta^m[b_{k,l}, h_{k,l}], \Theta^m[c_{k,l}, h_{k,l} + m\Delta h], h_{i,j} \} \quad (17)$$

where

$$b_{k,l} = \|k_{k,l}\| \Psi(\mu - lh_{k,l})$$

$$c_{k,l} = (1 - \|k_{k,l}\|) \Psi(\mu^{(k,l)} p_{i,j} \cdot n_{k,l}) - \theta \Psi(\mu - lh_{k,l}).$$

Using (14) and (17), the dynamics of the module is given by

$$h'_{i,j} = \frac{1}{2}(m+1) \max_{(k,l) \in \delta_{\Delta i,j}} \left\{ \Theta^m[(1 - \|k_{i,j}\|) a_{k,l}, h_{k,l}], \Theta^m[\|k_{i,j}\| b_{k,l}, h_{k,l}], \Theta^m[\|k_{i,j}\| c_{k,l}, h_{k,l} + m\Delta h], h_{i,j} \right\} + \frac{1}{2}(1-m) \min_{(k,l) \in \delta_{\Delta i,j}} \left\{ \Theta^m[(1 - \|k_{i,j}\|) a_{k,l}, h_{k,l}], \Theta^m[\|k_{i,j}\| b_{k,l}, h_{k,l}], \Theta^m[\|k_{i,j}\| c_{k,l}, h_{k,l} + m\Delta h], h_{i,j} \right\}. \quad (18)$$

Only the first term of the right hand side of (18) operates with $m=1$, i.e., upward propagation, while only the second term of the right hand side of (18) operates with $m=-1$, i.e., downward propagation.

Next let us illustrate how the module operates. Set the initial conditions for each discrete height as follows (see Fig.2(a)):

$$h_{i,j} = \begin{cases} 0 & \text{If } (i,j) \in c_0 \\ -m\mu & \text{otherwise} \end{cases}$$

Upward propagation is carried out by iterating the dynamics with $m=1$. Figure 2(a) shows a schematic drawing of $h_{i,j}$ obtained from the difference dynamics of (18) in the upward propagation. After convergence of each discrete height, then set m and the initial conditions as follows:

$$m = -1,$$

$$h'_{i,j}(0) = \|k_{i,j}\| \Psi(\mu - lh_{i,j}) h_{i,j} + (1 - \|k_{i,j}\|) \Psi(\mu - lh_{i,j}) m\mu.$$

In the similar manner, upward and downward propagations are carried out alternatively, until discrete heights for all contour curves converges. Figure 2(c) shows a schematic drawing of $h_{i,j}$ obtained from the difference dynamics of (18) in the downward propagation.

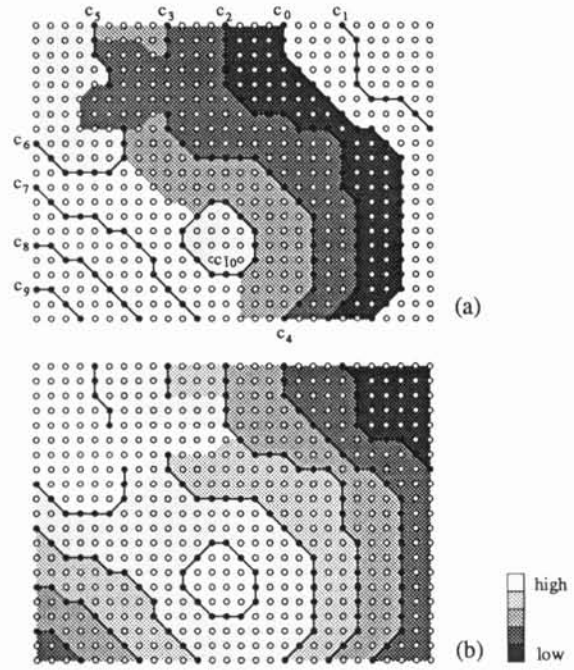


Fig.2 Assignment of discrete values to the contour curves as their height. The way how the height information are propagated (a) toward the upward direction, (b) toward the downward direction.

C. Designing the module 3

In the third module, three-dimensional ground configuration is reconstructed exploiting regularization technique [2][3]. Assume that three-dimensional configuration, $z=h(x,y)$, is two times continuously differentiable. Then, the associated smoothness constraint is given by the following cost function [4] :

$$U = \iint_D \left\{ \left(\frac{\partial^2 h}{\partial x^2} \right)^2 + \left(\frac{\partial^2 h}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 h}{\partial y^2} \right)^2 \right\} dx dy \quad (19)$$

In order to accelerate the reconstruction process, the pyramidal computation method is used. Let D_ρ denote a digital image plane which is scaled by $\rho \in \{0,1,2,3,4\}$:

$$D_\rho = \{(p,q) \mid 0 \leq p < (2\delta L + \Delta L)2^{-\rho}, 0 \leq q < (2\delta L + \Delta L)2^{-\rho}\}. \quad (20)$$

Let $\{^p h_{p,q} \in \mathbb{R}^1 \mid (p,q) \in D_\rho\}$ denote a set of continuous heights on D_ρ . The discretized cost function associated with (18) is given by

$$E_s(p,q) = (h_{p+1,q}^* - 2h_{p,q}^* + h_{p-1,q}^*)^2 + 2(h_{p+1,q+1}^* - h_{p+1,q}^* - h_{p-1,q+1}^* + h_{p-1,q-1}^*)^2 + (h_{p,q+1}^* - 2h_{p,q}^* + h_{p,q-1}^*)^2 \quad (21)$$

In the case that (p,q)-node is in the neighborhood of the edge of D_ρ , the slightly modified cost function [5] is used in stead of (21). A set of the discrete heights associated with contour curves is used as the boundary conditions. A set of the optimal continuous heights minimize the cost function (21). In the conventional technique, it is not easy to compute a optimal solution, because of a lot of local minima in the energy surface of (21). Especially, the undesirable solutions are often obtained in the following two regions (see Fig.3(a)):

1. Edge region of the image plane.
2. Neighborhood of a contour curve whose structure is complicated.

In order to avoid the undesirable solution in the region 1, we define a frame of the image plane and the associated cost functions. In order to avoid undesirable solutions in the region 2, the discrete height which is obtained by the second module is locally used to correct them.

Let us design several cost functions. Set the subsets of D_ρ as following (see Fig.3(b)):

$$\begin{aligned} R_\rho(0) &= \{(p,q) \mid \delta L 2^\rho \leq p < (\delta L + \Delta L) 2^\rho, \delta L 2^\rho \leq q < (\delta L + \Delta L) 2^\rho\} \\ R_\rho(1) &= \{(p,q) \mid (\delta L + \Delta L) 2^\rho \leq p < (2\delta L + \Delta L) 2^\rho, \\ &\quad (\delta L + \Delta L) 2^\rho \leq q < (2\delta L + \Delta L) 2^\rho\} \\ R_\rho(2) &= \{(p,q) \mid \delta L 2^\rho \leq p < (\delta L + \Delta L) 2^\rho, \\ &\quad (\delta L + \Delta L) r \leq q < (2\delta L + \Delta L) 2^\rho\} \\ R_\rho(3) &= \{(p,q) \mid 0 \leq p < \delta L 2^\rho, (\delta L + \Delta L) 2^\rho \leq q < (2\delta L + \Delta L) 2^\rho\} \\ R_\rho(4) &= \{(p,q) \mid 0 \leq p < \delta L 2^\rho, \delta L 2^\rho \leq q < (\delta L + \Delta L) 2^\rho\} \\ R_\rho(5) &= \{(p,q) \mid 0 \leq p < \delta L 2^\rho, 0 \leq q < \delta L 2^\rho\} \\ R_\rho(6) &= \{(p,q) \mid \delta L 2^\rho \leq p < (\delta L + \Delta L) 2^\rho, 0 \leq q < \delta L 2^\rho\} \\ R_\rho(7) &= \{(p,q) \mid (\delta L + \Delta L) 2^\rho \leq p < (2\delta L + \Delta L) 2^\rho, 0 \leq q < \delta L 2^\rho\} \\ R_\rho(8) &= \{(p,q) \mid (\delta L + \Delta L) 2^\rho \leq p < (2\delta L + \Delta L) 2^\rho, \\ &\quad \delta L 2^\rho \leq q < (\delta L + \Delta L) 2^\rho\} \end{aligned} \quad (22)$$

$R_\rho(i)$ ($i = 1, \dots, 8$) corresponds to the frame of the image plane D , while $R_\rho(0)$ corresponds to image plane D .

In the case that the (p,q)-node belongs to $R_\rho(i)$ ($i = 1, \dots, 8$), cost functions are given by (23)-(25).

$$E_L(p,q) = \sum_{(k,l) \in \Delta_{p,q}^4} (\rho h_{p,q}^* - \rho h_{k,l}^*)^2 \text{ for } (p,q) \in R_\rho(n) \text{ (n=1,3,5,7)} \quad (23)$$

$$E_L(p,q) = \sum_{l \in \{-1,1\}} (\rho h_{p,q}^* - \rho h_{p,q+l}^*)^2 \text{ for } (p,q) \in R_\rho(n) \text{ (n=2,6)} \quad (24)$$

$$E_L(p,q) = \sum_{k \in \{-1,1\}} (\rho h_{p,q}^* - \rho h_{p+k,q}^*)^2 \text{ for } (p,q) \in R_\rho(n) \text{ (n=4,8)} \quad (25)$$

On the other hands, in the case that the (p,q) belongs to $R_\rho(0)$, the cost function depends upon ρ . With $\rho = 0, 1$ or 2 , the following cost function is applied:

$$E_L(p,q) = E_S(p,q) \quad (26)$$

In this case, the discrete heights of contour curves are considered as the boundary conditions. On the other hand, with $\rho = 1/16$ or $1/8$, the cost function is given by:

$$E_L(p,q) = E_S(p,q) + \lambda \mathbf{1}_{k,p,q} \|(\rho h_{p,q}^* - h_{P,Q})^2 \quad (27)$$

where

$$(P,Q) = (p - \delta L, q - \delta L)$$

and $h_{P,Q}$ denotes the discrete height of the (P,Q)-node obtained by the second module.

The differential dynamics of the module is obtained by applying the steepest decent method to (23)-(27).

$$\partial^\rho h_{p,q}(\tau) / \partial \tau = -\alpha \partial E_L(p,q) / \partial h_{p,q} \quad (28)$$

where τ and α denote time parameter and a time constant, respectively.

Finally we illustrate how the third module operates. First a set of discrete heights is computed in the following manner. If the (i,j)-node on D belongs to a contour curve, i.e., c_n , or the upward neighborhood of c_n , the discrete height of c_n is substituted to $h_{i,j}$. On the other hand, if the (i,j)-node belongs to the downward neighborhood of c_n , the discrete height of c_n is subtracted by Δh , and the value is substituted to $h_{i,j}$. These discrete values are used for correction of undesirable solutions, and the above computation is easily carried out by the second module. Next, the dynamics (27) is iterating until all variables converge under the following conditions:

$$\rho = 4, \text{ and } {}^4 h_{p,q}^* = 0 \text{ for all } (p,q) \in D_4.$$

In the next step, using a set of discrete height, i.e., $\{h_{i,j} \mid (i,j) \in D\}$, the set of continuous heights, i.e., $\{h_{p,q}^* \mid (p,q) \in D_4\}$, is corrected as follows:

$$\begin{aligned} \text{If } [{}^4 h_{p,q}^*] - h_{I,J} = 0 \text{ holds, set } {}^4 h_{p,q}^* &= h_{I,J} \\ \text{otherwise, set } {}^4 h_{p,q}^* &= h_{I,J} \end{aligned}$$

where $(I,J) = ((i+\delta L)2^{-4}, (j+\delta L)2^{-4})$, and $[x]$ is the function that rounds off x . Then, initial conditions for a set of discrete height, $\{{}^3 h_{p',q'}^* \mid (p',q') \in D_3\}$, is given by

$${}^3 h_{p',q'}^* = {}^4 h_{[p'/2], [q'/2]}^*$$

The dynamics (28) is iterating again with decreasing ρ , i.e., $\rho=3$. In the similar process is repeated until $\rho = 0$.

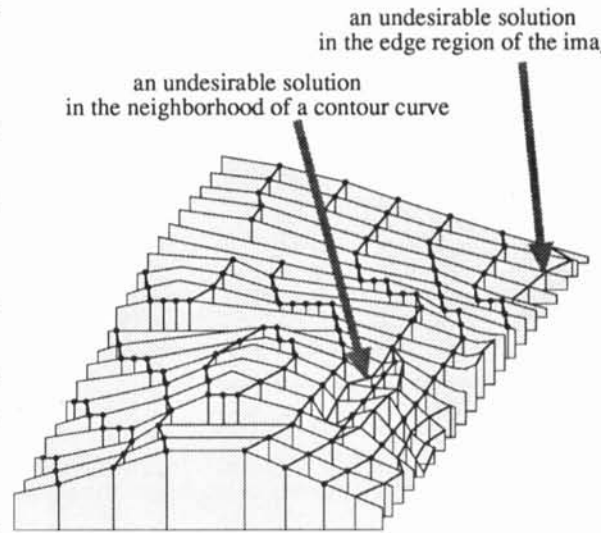


Fig.3(a) The undesirable solutions.

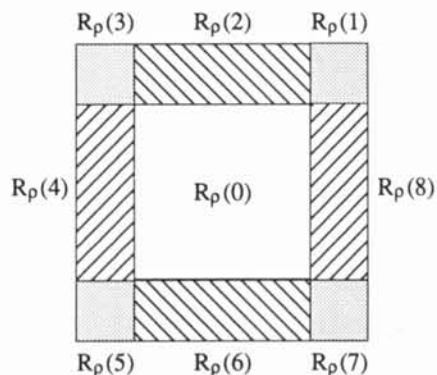


Fig.3(b) Subsets of D_p . $R_p(0)$ corresponds to the original image data. $R_p(n)$ ($n=1, \dots, 8$) correspond to the frame of the image plane D .

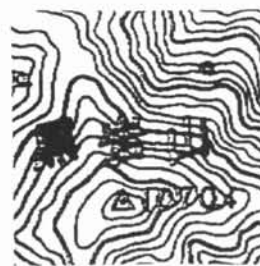
III. Numerical Experiments and Results

To demonstrate the performance of the system, the 1/25,000 scaled topographical map "Hourigawa" which is published by Geographical Survey Institute of Ministry of Construction of Japan is read by a five-bit gray scale scanner with 300dpi resolution. Figure 4 (a) shows the associated original image data with 256x256 pixel, while Figure 4(b) shows the extracted contour map from Fig.4(a). Fig.4 shows the reconstructed three-dimensional configuration and contour curves. It is clearly seen that all of contour curves are completely reconstructed and they are consistent with the original imperfect contour curves.

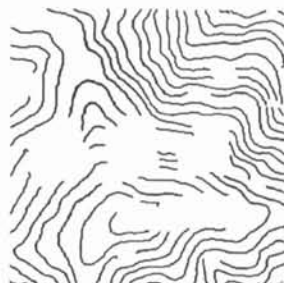
Acknowledgement This research was supported by Nippon Steel Co. The authors would like to thank T.Furukawa of Nippon Steel Co., W.Kim, K.Masu, T.Tsukahara, I.Tokuda of Tsukuba University for useful comments and exciting discussions.

References

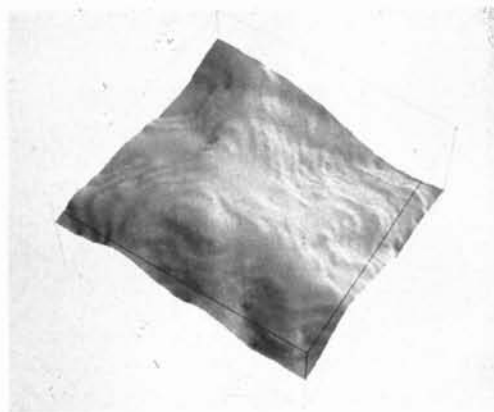
- [1]H.Freeman,"Computer processing of line-drawing images", Computing Surveys, 6,1, pp.57-97,1974
- [2]T.Poggio, et al., "Computational vision and regularization theory", Nature, Vol.317, pp.314-319, 1985.
- [3]K.Ikeuchi, et al., "Numerical shape from shading and occluding boundaries", Artificial Intelligence, Vol.17, pp.141-184, 1981.
- [4]D.Terzopoulos, "The Computation of Visible-Surface Representations", IEEE Trans. PAMI, Vol.PAMI-10, Vol.4, pp.417-438,1988.
- [4]D.Terzopoulos:"Image Analysis Using Multigrid Relaxation Methods", IEEE Trans. PAMI, Vol.PAMI-8, No.2, pp.129-139, 1986.



(a)



(b)



(c)



(d)

Fig.4 Experimental results. (a) Original image. (b) Imperfect contour curves extracted from original image. (c) Reconstructed three-dimensional configuration of the ground. (d) Reconstructed contour curves obtained from the configuration (c), by taking the zero-crossing lines from obtained configuration.

