Region-based Contour Tree によるディジタル画像の位相構造記述とその応用

水田　　忍†　　松田　哲也†

† 京都大学大学院情報学研究科システム科学専攻　〒 606–8501 京都市左京区吉田本町
E-mail: †{smizuta,tetsu}@i.kyoto-u.ac.jp

あらまし　画像による観察対象間の位相関係抽出を目的として，我々は先に，多次元スカラ場で等値面の位相構造を記述する Contour Tree を，ディジタル画像から生成する手法を提案した [1]．提案手法により得られる Contour Tree は，１つの領域が単一のスカラ値で表される領域集合の位相構造を記述するデータ構造と考えることが出来る．本研究では，このようなデータ構造を Region-based Contour Tree と呼ぶものとし，データ構造と生成手法の検討を行う．また，提案手法を応用した幾つかのディジタル画像処理手続きについて述べる．
キーワード　Contour Tree, 領域ベース，位相構造，ディジタル画像

Description of the Topological Structure of Digital Images by Region-based Contour Tree and Its Application

Shinobu MIZUTA† and Tetsuya MATSUDA†

† Department of Systems Science, Graduate School of Informatics, Kyoto University
Yoshida Hon-machi, Sakyo-ku, Kyoto, 606–8501 Japan
E-mail: †{smizuta,tetsu}@i.kyoto-u.ac.jp

Abstract　In order to extract the topological relationship among objects in observed images, we have previously proposed a method to construct Contour Tree, which describes the topological structure of isosurfaces [1]. The Contour Tree constructed by the method can be considered as a data structure describing the topological relationship among regions, each of which has identical field value. We discuss about this data structure named Region-based Contour Tree and the procedure to construct it. We also show several applications of Region-based Contour Tree for digital image processing.
Key words　Contour Tree, Region-based, topological structure, digital image

1. Introduction

With the advance of the imaging technology and the improvement of computer power, the opportunities of using digital images are rapidly increasing. In the medical field, various types of digital images are used such as two-dimensional (2D) X-ray projection images, 3D X-ray computer tomography images, magnetic resonance images, and the temporal series of these images.

The final goal of our research is to extract, analyze and display the topological structure of observed objects from multidimensional digital images. These are difficult issues especially when the images are 3D or higher.

If an object corresponds to a single pixel value in an image, the topological structure of isosurfaces directly represents the structure of objects. Therefore, we will focus on the structure of isosurfaces in digital images.

"Contour Tree" (CT) [2] is a data structure to describe the topological relationship among isosurfaces in multidimensional scalar fields, based on the relationship between the critical points (local maxima, local minima and saddle points) in the fields. In the conventional methods to construct CT, the scalar fields are assumed to be continuous and critical points that do not have area are uniquely found.

When digital images are represented as the set of pixels having area, the position of critical points cannot define in the images. In constructing CT from digital images, the characteristic is a problem because the nodes of CT must correspond to the critical points.

In this paper, we propose a modified data structure of the conventional CT named Region-based Contour Tree (RBCT). RBCT describes the topological relationship among isosurfaces defined by the set of regions, without introducing the information of critical points.

Recently, we have proposed a method to construct CT from digital images [1]. The resulted CT can be considered as RBCT.

Using RBCT, several procedures of digital image processing can be carried out in simple ways. We also show some procedures for image processing with RBCT.

In the following sections, the scalar fields are basically assumed to be 2D. However, the procedures in this paper can be applied to higher-dimensional fields immediately.

2. Contour Tree

2.1 Data structure of Contour Tree

In multidimensional scalar field, isosurfaces are defined as boundaries of field values. In this paper, we limit the isosurfaces to closed surfaces. These boundaries are called "isolines" in 2D space, but we use the term "isosurfaces" for all dimensions.

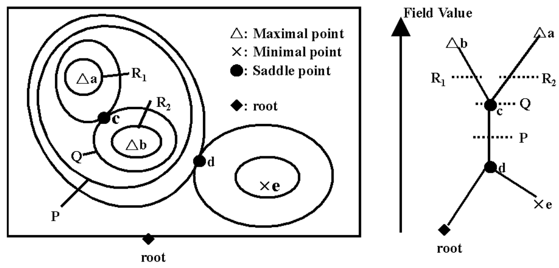
Contour Tree (CT) is a tree-structured graph, representing the transition of isosurfaces (appearance, disappearance, expansion, contraction, join and split) involved in the increase or decrease of the threshold of field value [2]. Figure 1 shows the outline of CT. Figure (a) represents a 2D scalar field, and P, Q, R_1, and R_2 are isosurfaces. a,...,e denote critical points (local maxima, local minima and saddles) where the topological changes of isosurfaces occur. Here, we define a special critical point named "root" on the closed surface surrounding the whole scalar field to evaluate.

Figure (b) is a CT corresponding to the scalar field (a). We define CT based on the references [6] [5] [7] as follows:

- CT is a tree-structured graph having nodes and arcs.
- A node of CT represents a critical point and the corresponding isosurface. A node and a critical point have a one-to-one relationship.
- An arc of CT links two nodes. The arc represents a region bounded by two isosurfaces corresponding to these two nodes. An arc and a region have a one-to-one relationship.

We call the nodes and arcs in the definition "supernodes" and "superarcs", respectively. In a region represented by a superarc, extraction or contraction of an isosurface occurs involved in the increase or decrease of the threshold.

CT can include additional nodes on superarcs to represent isosurfaces in the regions corresponding to the superarcs. The isosurfaces do not include any critical points. We call these nodes "regularnodes." "Nodes" consist of supernodes and regularnodes. We use the word "arcs" as the links between nodes. We call this type of CT Augumented Contour Tree (ACT) [7].



(a) 2D scalar field (level set) (b) Contour Tree

Fig. 1 Contour Tree.

2.2 Construction of Contour Tees from continuous scalar fields

In most of the procedures to construct CT, the scalar field for description is assumed to be continuous. In order to represent the field, multidimensional mesh is commonly used. The mesh consists of vertices and edges, and the scalar field is divided into cells by the mesh. Each vertex of the mesh describes the position in the field and the correspondent field value. Field values inside the cells are represented by interpolation.

Tarasov et al. [6] and van Kreveld et al. [5] have proposed procedures to construct CT for 2D/3D scalar fields efficiently. Carr et al. have improved these procedures and proposed a procedure for scalar fields of any dimensions [7]. Takahashi et al. have proposed a method to construct Volume Skeleton Tree, which is a binary tree describing the topological changes of isosurfaces in 3D scalar field [4]. Volume Skeleton Tree is a variant of CT, and genus change of 3D isosurface is also described. In these procedures, a scalar field is represented by simplicial mesh and the field values inside the cells are extracted by linear interpolation. In this case, it is guaranteed that the critical points are on the vertices of the mesh.

Pascucci et al. have proposed another efficient procedure to construct CT [3]. Their procedure can be applied to rectangular meshes. In the procedure, bilinear and trilinear interpolation is introduced to calculate the field values inside the cells for 2D and 3D scalar field, respectively. In this case, the critical points can be inside of the cell and the procedure to construct CT is more complicated than simplicial mesh case. Recently, Carr have introduced a framework named Finite State Machine [8] for the procedure to construct CT by the procedure of Pascucci [3]. Finite State Machine describes the transition of isosurfaces in the cells by the change of isovalue. From the analysis using this method, the author have selected Marching Cubes [9] [10] instead of trilinear interpolation to construct CT.

3. Region-based Contour Trees from digital images

In digital images, the isosurfaces have different characteristics from those in continuous scalar fields. In this section, we propose a modified data structure of Contour Trees for digital images named Region-based Contour Trees, and describe the procedure to construct them.

3.1 Definition of isosurfaces in digital images

In general, a digital image can be described as a set of pixels on the vertices of a multidimensional, rectangular grid. The pixels have values of non-negative, finite integer.

A digital image can be binarized using a threshold T. Here we denote $R_i(T)$ as the i-th region ($i = 1, 2, ...$) of connected pixels $\{P(t)|t \geqq T\}$ and $S_j(T)$ as the j-th region ($i = 1, 2, ...$) of connected pixels $\{P(t)|t \leqq T\}$. Using R and S, an isosurface can be represented by a pair of regions $[R_a(T), S_b(T-1)]$, where one of the regions surrounds the other and the isosurface is the boundary between them. Figure 2(a) shows examples of isosurfaces for the threshold T=3. In this example, the isosurfaces are represented by $\{[R_1(3), S_1(2)], [R_1(3), S_2(2)], [R_2(3), S_1(2)]\}$.

In binary images, ones of brighter or darker pixels are regarded as foreground, and the other becomes background. In order to avoid the contradiction between the connected regions of foreground / background, different types of connectivity are introduced [11]. Generally, a combination of 8- and 4-connectivity for 2D images is used for foreground and background. For 3D images, 26- and 6-connectivity are commonly used.

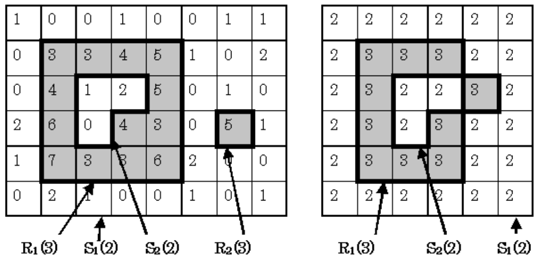
In the following description, we treat brighter $R_i(T)$ as a foreground region, with 8-connectivity of pixels for 2D images. Figure 2(b) shows an example of isosurfaces using this condition of connectivity. In this example, regions $S_1(2)$ and $S_2(2)$ are properly divided as the outside and inside of region $R_1(3)$. If 8-connectivity is also used for background regions, $S_1(2)$ and $S_2(2)$ are connected.

In order to guarantee that foreground regions are surrounded by background regions in any threshold $T > 0$, we set the values of the pixels at outside boundary as 0 in this paper [4]. We can describe the isosurface surrounding the whole image as the outside of $R_1(0)$.

3.2 Characteristics of isosurfaces in digital images

Figure 3 shows six types of the transition of isosurfaces in a digital image along the continuous decrease of threshold t. The range of the threshold for the left images in this figure are $2 < t \leqq 3$ and that for the right ones are $1 < t \leqq 2$. These types of transition can be described by CT (ACT), as mentioned in section 2.

As shown in the figure, the isosurfaces are discontinuously



(a) Example of isosurfaces (Threshold = 3)

(b) Difference of connectivity between foreground/background region

Fig. 2 Isosurfaces of digital images.

changed between integer threshold $t = T$ and $t = T + \epsilon$ where $T = 2$ in this example and ϵ is a small positive value. From the discontinuity, any critical point and corresponding isosurface does not appear during the change of threshold. Here we denote the set of isosurfaces before and after the transition as $C^+(T)$ and $C^-(T)$, respectively.

The change of isosurface between integer threshold T and $T + \epsilon$ is caused by a set of pixels $P(T)$ having value $T = 2$. There are several relationships among the regions relating the transition of isosurfaces as follows:

- All isosurfaces $C_i \in C^+(T)(i = 1, ..., M)$ are the boundaries between one region $S_1(T)$ and adjacent regions $R_i(T+1)$. Here $S_1(T)$ surrounds $R_i(T+1)$ or $R_i(T+1)$ surrounds $S_1(T)$. Similary, All isosurfaces $C_j \in C^-(T)(j = 1, ..., N)$ are the boundaries between one region $R_1(T)$ and adjacent regions $S_j(T-1)$.
- The set of pixels $P(T)$ related to the transition of isosurfaces are included both in $R_1(T)$ and $S_1(T)$.
- $R_i(T+1) \subset R_1(T)$ and $S_j(T-1) \subset S_1(T)$.

If the transition of isosurfaces is "appearance" or "disappearance", $M = 0$ or $N = 0$, respectively. If $M = N = 1$, the transition does not include topological changes.

Different types of transitions can occur in the same time. Figure 4 shows an example for the combined transition. This shows these characteristics again.

3.3 Region-based Contour Tree

As described above, any critical point or corresponding isosurface does not appear in the change of isosurface. Therefore, the nodes of CT from a digital image cannot correspond to any critical point. The fact is against the definition of CT in section 2. Here we modify the definition of CT to describe digital images properly. The definition of the proposed CT as the structure of ACT is as follows:

- CT is a tree-structured graph having nodes and arcs.
- A supernode of CT represents a set of isosurfaces related to one transition of isosurfaces involving topological changes.

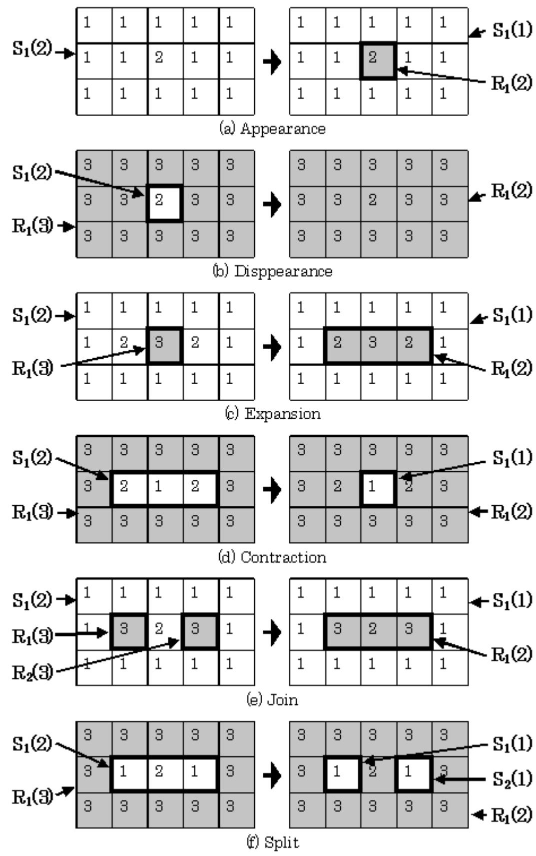


Fig. 3 Transition of isosurfaces in a digital image.

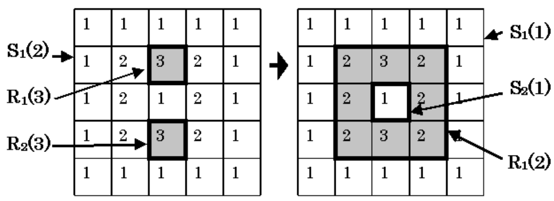


Fig. 4 Combined Transition of isosurfaces (join and split).

- A superarc of CT links two nodes. The arc represents a region bounded by two isosurfaces corresponding to these two nodes. An superarc and a region have a one-to-one relationship.
- A regularnode is on a superarc. The node represents a transition of isosurface in the regions corresponding to the superarc. Nodes consist of supernodes and superarcs. Arcs links nodes.

From the characteristics of isosurfaces in digital images, we can define:

- $n_i(i = 1, ..., X)$: a node, and a transition of isosurfaces.
- $a_j(j = 1, ..., Y)$: an arc.
- $C(n_i) =\{C^+(n_i), C^-(n_i)\}$: a set of isosurfaces related to n_i.
- $P(n_i)$: a set of pixels related to n_i.
- $V(n_i)$: a field value that the pixels $P(n_i)$ have.
- $R(n_i)$: a region of connected pixels having pixel value $t \geqq V(n_i)$ and including $P(n_i)$.
- $S(n_i)$: a region of connected pixels having pixel value $t < V(n_i)$ and including $P(n_i)$.

Here we can represent all the transitions of isosurfaces in a digital image by the set of nodes. Since any pixel in an image takes part in a transition of isosurfaces, each pixel is an element of $P(n_i)$ of exactly one node n_i.

If all the transitions of isosurfaces are represented by the nodes of CT, an arc a_j which links n_p and n_q represents exactly one isosurface. If $V(n_p) > V(n_q)$ the isosurface is the boundary between $R(n_p)$ and $S(n_q)$ where the threshold t is $V(n_p) \geqq t > V(n_q)$. Otherwise, it is the boundary between $S(n_p)$ and $R(n_q)$ where the threshold t is $V(n_q) \geqq t > V(n_p)$. If the nodes represent all the transitions of isosurfaces, the arcs represent all the isosurfaces in the image. If necessary, we can represent the isosurface surrounding the whole image in the following procedure:

- Set a node which represents outside region of an image. We call the node "virtual node".
- Find a node n_r where $P(n_r)$ includes the pixels at outside boundary. We call the node "root node".
- Set an arc which links the virtual node and the root node. The arc represents the isosurface surrounding the whole image. We call the arc "root arc".

It can be considered that the nodes of the proposed CT represent regions in an image, and the CT describes the topological structure of the image from the relationship among the regions. From this characteristic, we call the CT as "Region-based Contour Tree (RBCT)." Figure 5 illustrates the RBCT of a digital image.

The nodes of the conventional CT describe isosurfaces, and the arcs describe regions bounded by isosurfaces. On the other hand, the nodes of RBCT describe the transition of isosurfaces and related regions, and the arcs describe isosurfaces.

RBCT is uniquely constructed from a digital image, describing all the isosurfaces and their relationship.

3.4 Rooted tree representation

In this section, we first describe the characteristics of isosurfaces described by conventional (Augmented) CT.

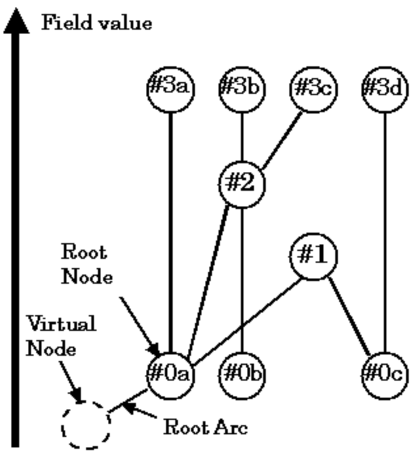
Let $\gamma(n)$ be the region surrounded by an isosurface corresponding to the node n on CT.

Lemma 1. *When nodes n_1 and n_2, n_2 and n_3 are connected by arc respectively, if $\gamma(n_1) \supset \gamma(n_2)$ then $\gamma(n_2) \supset \gamma(n_3)$.*

Proof. If $\gamma(n_2) \subset \gamma(n_3)$, both $\gamma(n_1)$ and $\gamma(n_3)$ have regions at the outer side of $\gamma(n_2)$, being against the definition of CT. Therefore $\gamma(n_2) \supseteqq \gamma(n_3)$. Since the isosurface for n_2 is not the same as that for n_3, $\gamma(n_2) \supset \gamma(n_3)$.

0	0	0	0	0	0	0	0	0	0	0
0	2	2	3	0	1	1	1	1	1	0
0	2	0	2	0	1	0	0	0	1	0
0	3	2	2	0	1	0	3	0	1	0
0	0	0	0	0	1	0	0	0	1	0
0	0	0	3	0	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0

(a) Input Image



(b) Region-based Contour Tree of (a)

Fig. 5 Region-based Contour Tree.

Let RN be the node corresponding to the isosurface surrounding the whole scalar field, defined in Section 2. CT can be considered as rooted tree where RN is the root node. Here, we can introduce the following theorems in the rooted tree representation of CT. Let $A(n)$ be the area of $\gamma(n)$.

Theorem 1. *For a node n and the child node n^c of n, $\gamma(n) \supset \gamma(n^c)$.*

Proof. If n is RN, obviously $\gamma(n) \supset \gamma(n^c)$. If n^p is the parent node of n and $\gamma(n^p) \supset \gamma(n)$, $\gamma(n) \supset \gamma(n^c)$ from Lemma 1. Therefore, for arbitrary node n, $\gamma(n) \supset \gamma(n^c)$.

Theorem 2. *For a node n and the all child nodes $n_i^c (i = 1, ..., N)$ of n, $A(n) > \sum_{i=1}^{N} A(n_i^c)$.*

Proof. From Theorem 1, $\gamma(n) \supset \gamma(n_i^c)$ for all i. Hence $\gamma(n) \supset \bigcup_{i=1}^{N} \gamma(n_i^c)$. If $i \neq j$, $\gamma(n_i^c) \cap \gamma(n_j^c) = \phi$. Therefore $A(n) > \sum_{i=1}^{N} A(n_i^c)$.

We can easily modify these theorems for RBCT. The modified theorems are as follows:

Theorem 1'. *For a node n, the parent arc a^p and a child arc a^c of n, $\gamma(a^p) \supset \gamma(a^c)$.*

Theorem 2'. *For a node n, the parent arc a^p and the all child arcs $a_i^c (i = 1, ..., N)$ of n, $A(a^p) > \sum_{i=1}^{N} A(a_i^c)$.*

Here,

- Parent arc a^p of node n is the arc between node n and the parent node n^p of n.
- Child arc a_i^c of node n is the arc between node n and a child node n_i^c of n.
- $\gamma(a)$ is the region surrounded by an isosurface corresponding to the arc a.
- $A(a)$ is the area of $\gamma(a)$.
- RN is the root node of RBCT.

From the characteristics of the transitions of isosurfaces in digital images, it can be also introduced:

Theorem 3. *For a node n, the parent node a^p and the all child arcs $a_i^c (i = 1, ..., N)$ of n, $A(a^p) = \sum_{i=1}^{N} A(a_i^c) + A(n)$.* Here, $A(n)$ is the area of the set of pixels $P(n)$, defined in section 3.3. Simply, we can treat $A(n)$ as the number of pixels in $P(n)$.

4. Construction of Region-based Contour Trees

4.1 Previous work for construction of Contour Trees from digital images

Several procedures have been proposed to construct tree structures from digital images [12] [13] [14]. In these procedures, constructed structures describe the topological change of connected regions in one of decreasing or increasing the threshold, where local minima or local maxima are not evaluated, respectively.

Asano et al. have introduced CT for digital images [15], where the method by van Kreveld et al. [5] has been used for the construction. However, precise procedure including the treatment of the difference of the pixel connectivity is not described.

4.2 Proposed method

Recently, we have proposed a procedure to construct CT from gray-scale digital images representing scalar fields [1]. This procedure is based on the method by Carr et al. [7]. Although the first aim of our method was to construct the conventional CT, the resulting CT can be considered as RBCT.

4.2.1 Construction of Contour Tree using Join Tree and Split Tree

In this section, we describe the outline of the procedure by Carr et al. [7]. Using this procedure, we can efficiently construct CTs from scalar fields in all dimensions. In this procedure, a Join Tree (JT) and a Split Tree (ST) are constructed from a simpicial mesh. The CT is constructed by merging the JT and ST.

At the beginning, the vertices of the mesh are sorted by field values in descending order. Then to construct a JT, the following procedure to each vertex is carried out in order. JT has nodes corresponding one-on-one with the all vertices of the mesh.

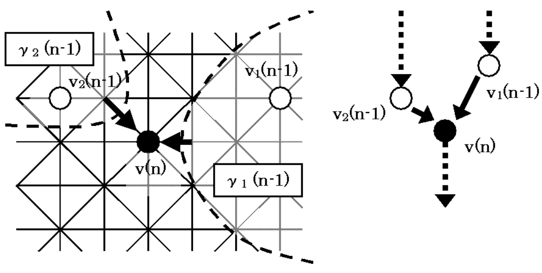
The procedure at step n searches the connectivity between the focused vertex and regions constructed by step $n-1$, and merge the vertex to the connected regions. We can classify the types of the merger into appearance (the vertex is connected to no region), expansion (one region), and join (plural regions).

Figure 6 shows the outline of the procedure. $v(n)$ denotes the n-th vertex. $\gamma_i(n-1), i = 1, ..., N$, are regions constructed by the $(n-1)$-th step, being connected to $v(n)$. $v_i(n-1), i = 1, ..., N$, are vertices having minimum field values in $\gamma_i(n-1)$; Here, one of n-th regions $\gamma(n)$ is constructed by connecting $\gamma_i(n-1), i = 1, ..., N$, and $v(n)$. In this step, the node corresponding with $v(n)$ is connected to the nodes corresponding with $v_i(n-1)$.

Figure 7 (a) shows an example of JT. In this figure, the numbers at the nodes indicates the field values. JT represents the appearance, expansion, and join of the regions surrounded by isosurfaces involved in the decrease of the corresponding field value.

ST can be constructed with the same procedure of JT to the vertices of ascending order. Figure 7 (b) shows an example of ST. ST represents the appearance, expansion, and join of the regions involved in the increase of the corresponding field value. In other words, ST represents split, contraction, and disappearance of the regions with the decrease of the field value.

Since each node in JT and ST correspond to a vertex of the mesh, the relation of nodes between JT and ST can be extracted immediately. CT having the correspondent nodes with JT and ST is constructed by merging these two trees, considering the structures of them. Figure 7 (c) shows an example of CT by merging JT (a) and ST (b).

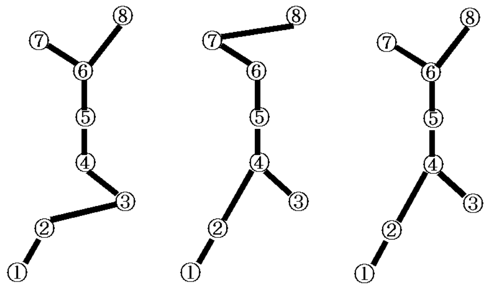


(a)two-dimensional mesh (b)Join Tree

Fig. 6 Procedure to construct Join Tree.

4. 2. 2 Construction of Contour Trees considering the connectivity of pixels

In the procedure described above, JT is constructed by processing from vertices with higher field values to them with lower, and the procedure for ST is opposite order. Therefore, the procedure to construct JT corresponds to the expansion of foreground regions, and that for ST corresponds to the expansion of background.

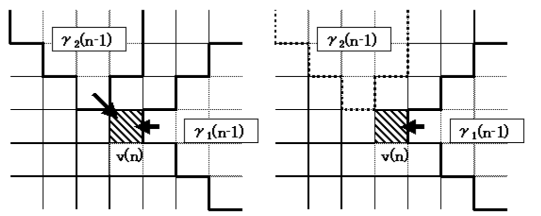


(a) Join Tree (b) Split Tree (c) Contour Tree

Fig. 7 Construction of Contour Tree.

As mentioned in Section 3. 1, different types of connectivity are introduced for foreground/background region in digital images. In our proposed method, the connectivity for foreground is used in the procedure to construct JTs, and that for background is used for STs. Figure 8(a) shows the procedure to extend foreground regions in constructing JT from digital images, corresponding to Figure 6(a) for triangular meshes. Figure 8(b) shows the procedure to extend background regions, where the connectivity to use is different from that for foreground. The procedure to construct CT from JT and ST can be same as the method by Carr et al. [7].

Since a node $v(n)$ of CT represents a pixel taking part in a transition of isosurfaces, the resulting CT can be considered as RBCT.



(a)Foreground (8-connectivity) (b)Background (4-connectivity)

Fig. 8 Procedure to extend foreground/background regions in digital images.

4. 2. 3 Influence of identical pixel value

When the pixels of digital images have values of finite integer, it frequently occurs that plural pixels have identical value. At the sorting of pixels in the procedure to construct JT and ST, the order of these pixels having the same value depends on another condition such as position of pixels. However, the order is irrelevant to the topological structure of isosurfaces. In order to eliminate this influence, we introduce following procedure. After constructing CT having nodes corresponding to all pixels, two nodes connected by an arc are merged into one node, when these nodes correspond to the pixels with identical value. The procedure is repeated until such pair of nodes disappears.

5. Applications for digital image processing

Using RBCT, several procedures of digital image processing can be carried out in simple ways. In this section, we introduce some procedures. Before the description of the procedures, we define several words as the supplement of the definition shown in section 3. 3:

- $n^p(a), n^c(a)$: parent and child node of arc a, respectively, which means that $n^p(a)$ is the parent of $n^c(a)$ in the two nodes linked by arc a.
- $n^d(a)$: descendant node of arc a, which means that $n^d(a)$ is $n^c(a)$ itself or the descendant of $n^c(a)$.

5. 1 Thresholding

In thresholding a digital image, any pixel having pixel value t is equal or greater than threshold T correspond to node n of RBCT where $V(n) \geqq T$. Considering that any connected region of $\{p\}$ is surrounded by isosurfaces corresponding to arc a_i where $V(n^c(a_i)) < T$, extraction of connected regions where pixel values $t \geqq T$ can be carried out by tracing nodes from $n^c(a_i)$ to the descendant nodes $n_j^d(a_i)$ connecting to $n^c(a_i)$, where $V(n_j^d(a_i)) \geqq T$. The procedure is as follows:

- Find arcs $\{a_i\}$ $(i = 1, ..., M)$ where $V(n^c(a_i)) \geqq T$ and $V(n^c(a_i)) < T$.
- For all a_i:
  - Find descendant nodes $n_j^d(a_i)(j = 1, ..., N_i)$ connecting to $n^c(a_i)$, where $V(n_j^d(a_i)) \geqq T$.
  - Set a region as $\{P(n_j^d(a_i))\}$.

In the procedure of finding a_i, the number of the connected regions is counted. Figure 9 illustrates the procedure of thresholding for Figure 5 (a).

5. 2 Segmentation

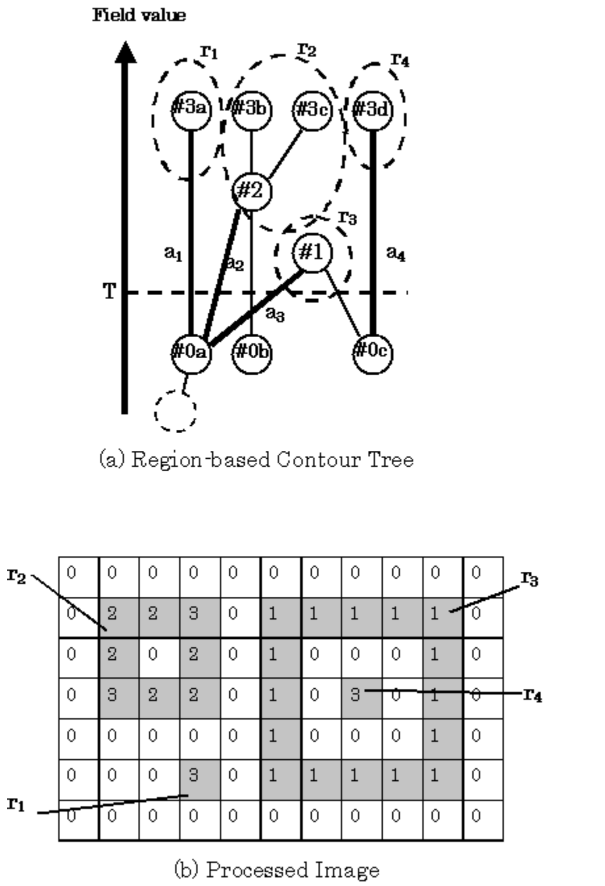
By applying Theorem 3 from chile node of arc a to its descendant nodes $n_j^d(a)$ recursively, we can introduce that $\gamma(a) = \{P(n_j^d(a))\}$. Segmentation of the region $\gamma(a)$ of selected arc a with hole filling can be carried out by tracing all descendant nodes $n_j^d(a)$ from the child node of a. The procedure is as follows:

- Find descendant nodes $\{n_j^d(a)\}$ $(j = 1, ..., N)$.
- Set a region as $\{P(n_j^d(a))\}$.

Figure 10 illustrates the procedure of thresholding for Figure 5 (a). The area of the region can be calculated by $A(a) = \sum_{j=1}^{N} A(n_j^d(a))$.

5. 3 Noise reduction

As described in Section 5. 2, the area of region for any arc



(a) Region-based Contour Tree

(b) Processed Image

Fig. 9 Thresholding using Region-based Contour Tree.

a can be calculated. Therefore, we can find small regions and remove them as noise reduction. The procedure is similar as morphological area opening/closing [16].

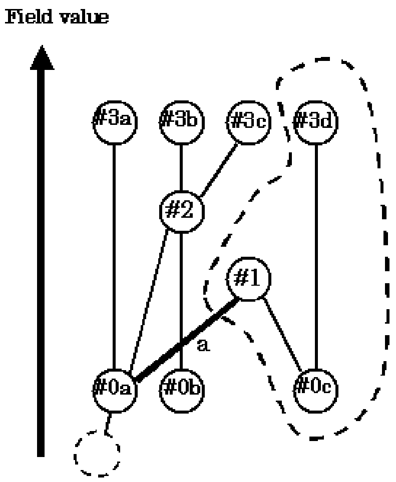
Noise reduction by removing small regions which area is smaller than A_min is carried out as follows:

- Find arcs $\{a_i\}$ $(i = 1, ..., M)$ where $A(a_i) < A_min$.
- For all a_i:
  - Find descendant nodes $n_j^d(a_i)(j = 1, ..., N_i)$.
  - For each pixel $p \in \{P(n_j^d(a_i))\}$, set pixel value $V(n^p(a_i)$

Figure 11 illustrates the procedure of thresholding for Figure 5 (a).

6. Conclusion

In order to extract the topological relationship among objects in observed images, we have previously proposed a method to construct Contour Trees (CT), which describe the topological structures of isosurfaces [1]. Since the isosurfaces of digital images have different characteristics from those of continuous scalar fields, we have proposed a modified data structure of conventional CT named Region-based Contour Tree (RBCT). RBCT can be considered as a data structure describing the topological relationship among regions. In this paper, we have discussed about the data structure of RBCT



(a) Region-based Contour Tree

0	0	0	0	0	0	0	0	0	0	0
0	2	2	3	0	1	1	1	1	1	0
0	2	0	2	0	1	0	0	0	1	0
0	3	2	2	0	1	0	3	0	1	0
0	0	0	0	0	1	0	0	0	1	0
0	0	0	3	0	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0

(b) Processed Image

Fig. 10 Segmentation using Region-based Contour Tree.

and the procedure to construct it. We have also shown several applications of RBCT for digital image processing.

Acknouledgement A part of this research was supported by Grant-in-Aid for Scientific Research from Japan Society for the Promotion of Science, and also by Grant-in-Aid for Scientific Research from Ministry of Education, Culture, Sports, Science and Technology.

References

[1] S. Mizuta, Y. Suwa, T. Ono, T. Matsuda, Description of the topological structure of digital images by Contour Tree and its two-dimensional display, IEICE Technical Report, PRMU2004-22, pp.59-64, May 2004 (in Japanese).

[2] R.W.Boyell, H. Ruston, Hybrid techniques for real-time radar simulation, IEEE Proceedings Fall Joint Computer Conference 63, pp.36-37, 1963.

[3] V. Pascucci, K. Cole-McLaughlin, Efficient Computation of the Topology of Level sets, Proc. IEEE Visualization 2002, pp.187-194, 2002.

[4] S. Takahashi, Y. Takeshima, I. Fujishiro, Topological volume skeletonization and its application to transfer function design, Graphical Models, Vol.66, Issue 1, pp.24-49, 2004.

[5] M. van Kreveld, R. van Oostrum, C.L. Bajaj, V. Pascucci, D.R. Schikore, Contour trees and small seed sets for isosurface traversal, in: Proceedings of the 13 th ACM Symposium on Computational Geometry, ACM, pp.212-220, 1997.

[6] S.P. Tarasov, M.N. Vyalyi, Comstruction of contour trees in 3D in O(n log n) steps, in: Proceedings of the 14 th ACM Symposium on Computational Geometry, ACM, 1998, pp.68-75.

0	0	0	0	0	0	0	0	0	0	0
0	2	2	3	0	1	1	1	1	1	0
0	2	0	2	0	1	0	0	0	1	0
0	3	2	2	0	1	0	3	0	1	0
0	0	0	0	0	1	0	0	0	1	0
0	0	0	3	0	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0

(a) Input Image

0	0	0	0	0	0	0	0	0	0	0
0	2	2	2	0	1	1	1	1	1	0
0	2	2	2	0	1	0	0	0	1	0
0	2	2	2	0	1	0	0	0	1	0
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	0	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0

(b) Processed Image

Fig. 11 Noise reduction using Region-based Contour Tree.

[7] H. Carr, J. Snoeyink, U. Axen, Computing contour trees in all dimensions, Computational Geometry, Vol. 24, pp.73-94, 2003.

[8] H. Carr, Topological Manipulation of Isosurfaces, PhD Thesis of The University of British Columbia, April 2004.

[9] W.E. Lorenson, H.E.Cline, Marching Cubes: A high resolution 3D surface construction algorithm, Computer Graphics, Vol.21, No. 4, pp.163-169, 1987.

[10] C. Montani, R. Scateni, R. Scopigno, A modified look-up table for implicit disambiguation of Marching Cubse, Visual Computer, Vol.10, pp.353-355, 1994.

[11] for example: J. Toriwaki, Three-dimensional digital image processing, Shoko-do, Tokyo, 2002 (in Japanse).

[12] P. Salembier, A. Oliveras, L. Garrido, Antiextensive connected operators for image and sequence processing, IEEE Trans. Image Processing, Vol.7, No.4, pp.555-570, 1998.

[13] RE Tarjan, Efficiency of a good but not linear set union algorithm, J. ACM, Vol.22, pp.215-225, 1975.

[14] V. Grau, M.A. Raya, C. Monserrat et al., Hierarchical image segmentation using a correspondence with a tree model, Pattern Recognition, Vol. 37, pp.47-59, 2004

[15] T. Asano, Mathematical Sciences, No. 433, pp.34-40, July 1999 (in Japanese).

[16] L.Vincent, Morphological area openings and closings for gray-scale images, in:Shape in Picture, pp.197-208, Springer-Verlag , 1994