数理解析研究所講究録
第 458 巻 1982 年 201-210

One-Pass Algorithms for Properties of Three-Dimensional Pictures

Kunio Aizawa
Faculty of Integrated Arts and Sciences
Hiroshima University

and Akira Nakamura
Department of Applied Mathematics
Hirishima University

Introduction

Recent years, there are a lot of requirements for three-dimensional (3D) data processing with the advance of computer tomography (CT). Some topological properties of 3D digital pictures are discussed in a series of papers by Rosenfeld and Morgenthaler [1] - [5]. In many cases, the 3D picture is represented by a 3D array of volume elements (voxels for short). For every pair of voxels, the connectedness is defined, and the objects and cavities are defined as the equivalence classes of the connectedness relation. These correspond to two-dimensional (2D) objects and holes, respectively. Moreover, in 3D case, there exist 3D holes whose properties are quite different from 2D ones. In this paper, we shall discuss algorithms such that for every 3D digital picture they compute the number of objects, cavities, and holes. Since we usually receive a series of 2D pictures as the output of a CT scanner, it seems to be natural that these algorithms scan such output plane by plane, from top to bottom, and stop when all of them are scanned. Thus our computational model consists of a 2D array of finite-state automata which scans a 3D digital picture, one plane at a time, in one-pass only. In [6], Selkow has discussed such algorithms for 2D digital pictures.

1. Preliminaries

Let Σ be a 3D array of lattice points, which we may assume without loss of generality to be n×n×n, i.e., $\Sigma = \{(i,j,k) \mid 1 \leq i,j,k \leq n\}$. A *3D digital picture* f is a mapping from Σ to {0,1}, i.e., $f:\Sigma \rightarrow \{0,1\}$. Each point (i,j,k) is called a *voxel*. To avoid special case we assume that $f(i,j,k) = 0$ if one of these i,j,k is equal to 1 or n. And the set of such points, $\{(i,j,k) \mid i=1 \vee i=n \vee j=1 \vee j=n \vee k=1 \vee k=n\}$, is called the *border* of Σ. Usually, the subset of Σ, $\{(i,j,k) \mid f(i,j,k)=1\}$, called *S*, and its complement is called $\overline{S}$. For every pair of points $X=(x_1,x_2,x_3)$ and $Y=(y_1,y_2,y_3)$, X and Y are *6-adjacent* if $|x_1-y_1|+|x_2-y_2|+|x_3-y_3|=1$; X and Y are *26-adjacent* if $\max(|x_1-y_1|, |x_2-y_2|,|x_3-y_3|)=1$. If points P and Q are 6-adjacent (26-adjacent), then P is called a *6-neighbor*(*26-neighbor*) of Q. To avoid ambiguous situations we assume that opposite types of adjacency are used for S and $\overline{S}$. A *6-path*(*26-path*) π is a sequence of points, $\pi = p_0,p_1,\ldots,p_m$, where, for all i such that $1 \leq i \leq m$, p_i is a 6-neighbor(26-neighbor) of p_{i-1}. Any two points P, Q of S called *connected in S* if there exists a path $P=p_0,\ldots,p_m=Q$ from P to Q, where $p_i \in S$. Evidently, "connected" is an equivalence relation. This relation partitions S into equivalence classes. These classes are called the *connected components* of S. In the same way, we may define *connected in* $\overline{S}$ and the *connected components* of $\overline{S}$. A connected component of S is called an *object* of S. Clearly, exactly one component of $\bar{S}$ contains the border of Σ. This component is called the *background* of S; all other components of $\bar{S}$ are called *cavities* of S.

Even in ordinary topology it is difficult to characterize holes. A hole may be thought of as a property of a boundary surface which makes it topologically equivalent to a torus. In another approach, an object is defined to have no holes if every simple closed curve in the object is continuously deformable within the object to a single point. We see from these remarks that the consept of a hole is different from those of objects and cavities; we cannot point to or label the points which constitute hole. Indeed, the points of objects and cavities cover the space, so that a hole is a property of these collections of points. Thus, when considering an object (and its cavities) we shall here try only to understand what is meant by the number of holes in the object, and not what is meant by a hole.

On the other hand, the *genus G(S)* of a set S in a 3D digital picture is defined as the number of objects in S (O(S)) plus the number of cavities in S (C(S)) minus the number of holes in S (H(S)). As already mentioned the definition of holes is not simple, and in paticular holes cannot be labelled to facilitate counting them. Since this can be done with objects and cavities, the definition of genus would defin the number of holes in S, and conversely. In [4], Morgenthaler has shown the methods computing G(S) directly from the local patterns of S:

(1) When 26-adjacency is used for S,

$$G_{26}(S) = \phi_1 - \phi_2 + \phi_3 - \phi_4 + \phi_5 - \phi_6 + \phi_7 - \phi_8,$$

where $\phi_1 = \#[2]$

$\phi_2 = \#[3] + \#[4] + \#[5]$

$\phi_3 = \#[6] + \#[7] + \#[8]$

$$\phi_4 = \#[9] + \#[10] + \#[11] + \#[12] + \#[13] + \#[14]$$
$$\phi_5 = \#[15] + \#[16] + \#[17]$$
$$\phi_6 = \#[18] + \#[19] + \#[20]$$
$$\phi_7 = \#[21]$$
$$\phi_8 = \#[22]$$

and by #[n] we mean the number of times the configuration n of Appendix A occurs in the picture S (in all orientations).

(2) When 6-adjacency is used for S,

$$G_6(S) = \psi_1 - \psi_2 + \psi_3 - \psi_4,$$

where
$$\psi_1 = \#[2]$$
$$\psi_2 = \#[3]$$
$$\psi_3 = \#[9]$$
$$\psi_4 = \#[22].$$

Morgenthaler has also shown that $G_{26}(\bar{S}) - G_6(S) = 1$ and $G_6(\bar{S}) - G_{26}(S) = 1$.

2. The number of objects, cavities, and holes

A *scanner* is an n×n array of finite-state automata

$$\left\langle \begin{matrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & & \\ \vdots & & & \vdots \\ \sigma_{n1} & & \cdots & \sigma_{nn} \end{matrix} \right\rangle .$$

Each automaton σ_{ij} is defined by a 7-tuple $< Q_{ij}, \delta_{ij}, \alpha_{ij}, a_{ij}, \lambda_{ij}, \beta_{ij}, b_{ij} >$, where Q_{ij}, the set of states, is a finite subset of integers, δ_{ij} is the next state function of the following form:

$$\delta_{ij}: \prod_{k=1}^{a_{ij}} Q_{\alpha_{ij}(k)} \times \{0,1\} \rightarrow Q_{ij},$$

where a_{ij} is an integer, and $\alpha_{ij}:\{1,\ldots,a_{ij}\} \rightarrow I\times I$ is a one-to-one function which enumerates the next state neighborhood of σ_{ij}. λ_{ij} is the output function of the following form:

$$\lambda_{ij}: \prod_{k=1}^{b_{ij}} Q_{\beta_{ij}(k)} \times \{0,1\} \rightarrow I,$$

where b_{ij} is an integer, and $\beta_{ij}:\{1,\ldots,b_{ij}\} \rightarrow I\times I$ is a one-to-one function which enumerates the output neighborhood of σ_{ij}. We will use $Q_{ij}(t)$ to represent the state of σ_{ij} at time t. It is assumed that the scanner advances one plane each unit of time and that it scans the first plane at time t=1. Thus the input to scanner element σ_{ij} at time t is f(t,i,j).

The *counter* C monitors the output of each element of the scanner,

thus $C(t) = C(t-1) + \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{ij}(t)$.

Now we shall describe the algorithms computing the number of objects, cavities, and holes.

(1) <u>Objects</u>

The set of states of scanner element σ_{ij} is $Q_{ij} = \{x \mid x$ is an integer and $|x| \leq (i+j-1)(i+j-2)/2+j\}$. Each automaton σ_{ij} starts in state 0 and remains in that state as long as 0's are scanned. When a voxel containing a 1 is reached, σ_{ij} will assume state (i+j-1)(i+j-2)/2+j. As σ_{ij} tracks a string of 1's, an extension of the component of S is sought, i.e., two automata which are actively tracking 1's are tracking same object if they are spatially neighbors or if they are in same state. All automata which are tracking same object assume same state (the state of the automaton having the smallest state). An automaton which passes a lower border of an object and has been in a state k enters the state -k for one period. The next time it would go directly to state 0 unless a 1 is encounted. If σ_{ij} enters the state -((i+j-1)(i+j-2)/2+j) and no others are in the state (i+j-1)(i+j-2)/2 +j, then σ_{ij} will output 1, i.e., one object has been scanned. The precise definition of δ_{ij} and λ_{ij} are represented in Appendix B.

(2) Cavities

Since all components of $\bar{S}$ except background component are cavities of S, the algorithm for counting objects of S can be also used for counting cavities of S by interchanging the roles of 1 and 0. In this case, the initial value of the counter C must be -1 to remove background component from cavities of S.

(3) Genus

For any S, every 2×2×2 local patterns in Appendix A is easily counted by our computational model. Thus the algorithm computing genus of S is easily constructed.

(4) Holes

Finally from the algorithms (1) - (3), we can construct the algorithm counting the number of holes in S since H(S) = O(S) + C(S) - G(S).

References

[1] Rosenfeld, A.: Three-Dimensional Digital Topology, Technical Report TR-936, Computer science Center, University of Maryland (1980).

[2] Rosenfeld, A. and D.G. Morgenthaler: Some Properties of Digital Curves and Surfaces, Technical Report TR-942, Computer Science Center, University of Maryland (1980).

[3] Morgenthaler, D.G. and A. Rosenfeld: Surfaces in Three-Dimensional Digital Images, Technical Report TR-940, Computer Science Center, University of Maryland (1980).

[4] Morgenthaler, D.G.: Three-Dimensional Digital Topology: The Genus, Technical Report TR-980, Computer Science Center, University of Maryland (1980).

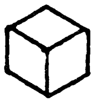
[5] Morgenthaler, D.G.: Three-Dimensional Simple Points: Serial Erosion, Parallel Thinning, and Skeltonization, Technical Report TR-1005, Computer Science Center, University of Maryland (1981).

[6] Selkow, S.M.: One-Pass Complexity of Digital Picture Properties, J.ACM Vol. 19, pp. 283-295 (1972).

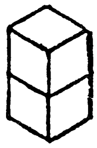
Appendix A

(All zeros)

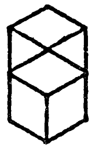
1 (=$\overline{22}$)



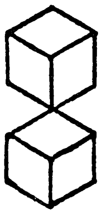
2 (=$\overline{21}$)



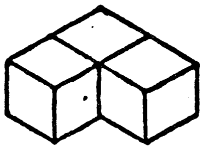
3 (=$\overline{20}$)



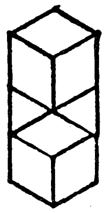
4 (=$\overline{19}$)



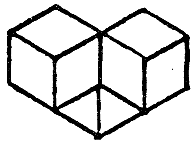
5 (=$\overline{18}$)



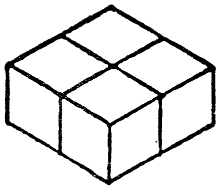
6 (=$\overline{17}$)



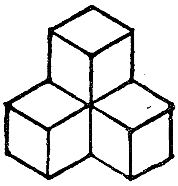
7 (=$\overline{16}$)



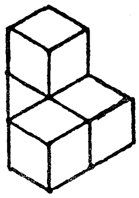
8 (=$\overline{15}$)



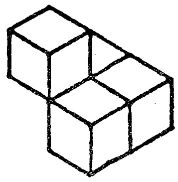
9 (=$\overline{9}$)



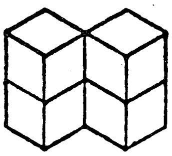
10 (=$\overline{10}$)



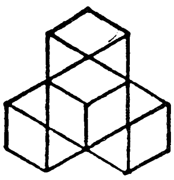
11 (=$\overline{11}$)



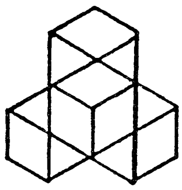
12 (=$\overline{12}$)



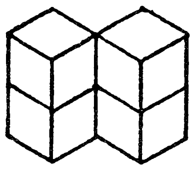
13 (=$\overline{13}$)



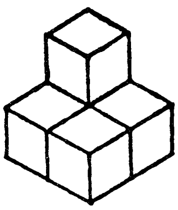
14 (=$\overline{14}$)



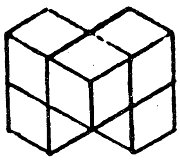
15 (=$\overline{8}$)



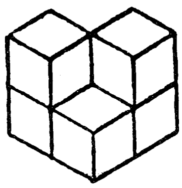
16 (=$\overline{7}$)



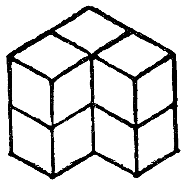
17 (=$\overline{6}$)



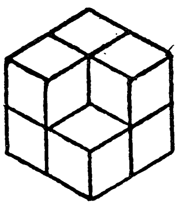
18 (=$\overline{5}$)



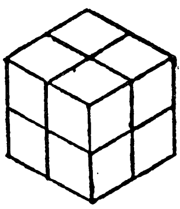
19 (=$\overline{4}$)



20 (=$\overline{3}$)



21 (=$\overline{2}$)



22 (=$\overline{1}$)

Appendix B

(1) When 6-adjacency is used for S,

$$\delta_{ij}(Q_{ij}(t-1), f(t,i,j)) = \begin{cases} ((i+j-1)(i+j-2)/2+j)\cdot f(t,i,j) & \text{if } Q_{ij}(t-1) \leqq 0 \\ \mathrm{LINK}(Q_{ij}(t-1)) & \text{if } Q_{ij}(t-1) > 0 \wedge f(t,i,j) = 1 \\ -\mathrm{LINK}(Q_{ij}(t-1)) & \text{if } Q_{ij}(t-1) > 0 \wedge f(t,i,j) = 0, \end{cases}$$

$$\lambda_{ij} = \begin{cases} 1 & \text{if } Q_{ij}(t) = -((i+j-1)(i+j-2)/2+j) \quad \text{for all } x,y \text{ such that } (x+y-1)(x+y-2)/2+y > (i+j-1)(i+j-2)/2+j,\ Q_{xy}(t) \neq -Q_{ij}(t), \\ 0 & \text{otherwise,} \end{cases}$$

where $\mathrm{LINK}(g) = \min_{1\leqq k\leqq n^2} \{(x_k+y_k-1)(x_k+y_k-2)/2+y_k \mid (\exists x_1)\ldots(\exists x_{k-1})(\exists y_1)\ldots(\exists y_{k-1})(Q_{x_1y_1}(t-1)=g)\wedge(\forall j)(2\leqq j\leqq k)((Q_{x_jy_j}(t-1)>0)\wedge((x_j,y_j) \text{ and } (x_{j-1},y_{j-1}) \text{ are 6-adjacent})\vee(Q_{x_jy_j}(t-1)=Q_{x_{j-1}y_{j-1}}(t-1)))))\}$.

(2) When 26-adjacency is used for S,

$$\delta_{ij}(Q_{ij}(t-1), f(t,i,j)) = \begin{cases} 0 & \text{if } Q_{ij}(t-1) \leqq 0 \wedge f(t,i,j) = 0 \\ \mathrm{LINK}((i+j-1)(i+j-2)/2+j) & \text{if } f(t,i,j) = 1 \\ -\mathrm{LINK}((i+j-1)(i+j-2)/2+j) & \text{if } Q_{ij}(t-1) > 0 \wedge f(t,i,j) = 0, \end{cases}$$

λ_{ij} is the same as in the case (1),

where $\mathrm{LINK}(g) = \min_{1\leqq k\leqq n^2} \{(x_k+y_k-1)(x_k+y_k-2)/2+y_k \mid (\exists x_1)\ldots(\exists x_{k-1})(\exists y_1)\ldots(\exists y_{k-1})((x_1+y_1-1)(x_1+y_1-2)/2+y_1 =g)\wedge(\forall j)(2\leqq j\leqq k)\ ((Q_{x_jy_j}(t-1)\ 0)\wedge((x_j,y_j) \text{ and } (x_{j-1},y_{j-1}) \text{ are 26-adjacent})\vee(Q_{x_jy_j}$

Appendix B (continued)

$$(t-1)=Q_{x_{j-1}y_{j-1}}(t-1)))))\}.$$