PROCEEDINGS OF SCIENCE

Fast Centerline Extraction Method of Cardiovascular Virtual Endoscopy

Jiajia Cong[1][2]
Department of Automation, Shanghai Jiao Tong University
No.800 Dongchuan Rd. Shanghai, China, 200240
E-mail: aanonymous@sjtu.edu.cn

Xin Yang
Department of Automation, Shanghai Jiao Tong University
No.800 Dongchuan Rd. Shanghai, China, 200240
E-mail: aanonymous@sjtu.edu.cn

Liping Yao
Shanghai Xinhua Hospital
No.1655 Kongjiang Rd. Shanghai, China, 200092
E-mail: itaa2004@sina.com

This paper introduces a fast centerline extraction method for the cardiovascular virtual endoscopy system. It is based on the current distance mapping algorithm but significantly enhances it. Our method consists of three major parts: image preprocessing and segmentation, euclidean distance transform (EDT) and centerline extraction using pairing heap. The main contribution is the extraction process, we adopt an improved method based on Dijkstra algorithm proposed by Ming Wang and use a new data structure called "pairing heap" to improve this method, and it can speed up the extraction process greatly. The experimental results show that our method is more efficient compared with existing methods.

ISCC 2015
18-19, December, 2015
Guangzhou, China

[1] Speaker;Corresponding Author
[2] Thank Shanghai Xinhua Hospital for providing the DSCT dataset.

1.Introduction

The Virtual Endoscopy (VE) technology is a new method of computer-aided medical diagnosis. It is a virtual roaming through the 3-D reconstruction of patient's organs, so that the doctors can observe the internal structures of patient's organs to assist disease diagnosing and surgical planning. Virtual Endoscopy (VE), compared with the traditional endoscopy, has many benefits: it is convenient and non-invasive, and patients are not required to suffer. After the reconstruction, the doctors can inspect the organs at any time. It can inspect some place where the traditional endoscopy can't access, such as the Cardiovascular Virtual Endoscopy System (CVE System) implemented by our lab [1].

The processes of virtual endoscopy are divided into five steps [1]: image data acquisition, image segmentation, 3D reconstruction, path planning and real-time rendering. The path planning of VE is mainly centerline extraction, in this paper, we do much research on the centerline extraction algorithm and propose an improved fast centerline extraction method for our Cardiovascular Virtual Endoscopy System, which is faster than the conventional centerline extraction method.

The remaining sections of our paper are arranged as follows: in section 2, we firstly introduce the current centerline extraction methods, then we propose our improved method for the Cardiovascular Virtual Endoscopy System; in section 3, we validate our proposed method in the CTA dataset from Shanghai Xinhua Hospital and make a comparison with the current methods; in the last section, we make a summary of this paper.

2.Methodology

2.1Overview of the Current Methods

In 1967, Blum introduced the definition of centerline[2]: it is a curve that connects all the center voxels of a tubular organ like heart or colon. Generally, the centerline of vessels usually has branches, and it is usually defined as the skeleton of the lumen; but in our Cardiovascular Virtual Endoscopy System, the centerline has no branches, and we just need to extract the centerline from the aorta to the left atrium and the left ventricle. The centerline extraction of tubular organ (skeleton) mainly has three methods: method based on image sequences [3], method based on 3D reconstruction grid data [4] and method based on the volume data. The method based on volume data can generally be divided into three categories: topological thinning [1], level set marching [1] and distance mapping [5].

2.1.1Topological Thining

Topological thinning method is considered to provide the highest precision centerline. It is to peel the voxel layer by layer, until there is only one voxel while maintaining the connectivity. Its principle is very simple, but the complexity of this algorithm is very high. For instance, a colon CT data, whose size is $512 \times 512 \times 512$, Hong et al. [6] spent several hours getting the final result.

2.1.2Level Set Marching

The principle of level set algorithm is to take the contour(or surface in 3D volume data) as a function (φ) in higher dimensional space (which is called level set function). The boundary of the shape in a plane is zero level set of this function. By calculating the boundary evolution, we find the path. The function's dimension is very high and computationally expensive. The Fast Marching Method (FMM) can speed up it, but energy function is sometimes difficult to obtain.

2.1.3 Distance Mapping

The distance mapping method is the fastest centerline extraction method so far. It calculates the distance field first and takes the volume data as a graph. Every voxel in the volume data is the vertex of the graph and the distance value is considered as the edge. In this algorithm, the centerline can be defined as the shortest/longest path from the start voxel to the end voxel in the graph. However, the precision of this method is not very high, especially the "hug corner" (in Fig.1) phenomenon. Countless researchers devoted themselves to this problem. In 2002, Ming Wang et al. proposed a minimum spanning tree algorithm based on the distance from boundary [5]. It can solve the "hug corner" problem effecitively.

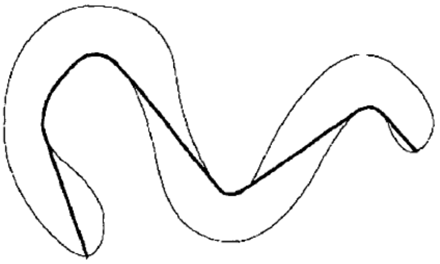

Figure 1: The "Hug Corner" in Path Planning

2.2 Description of the Proposed Method

In our CVE System, the volume data is acquired from Dual Source CT (DSCT). The DSCT has great advantage in heart imaging. It uses both two-ray sources and two detectors systems and it has fast imaging speed. This makes our system do better in auxiliary diagnosis. The method we proposed is on the basis of distance mapping method. It consists of three steps: image preprocessing and segmentation, euclidean distance transform (EDT) and centerline extraction using pairing heap.

2.2.1 Image Preprocessing and Segmentation

The raw DSCT data consists of all organs including heart, lung, ribs, spine and others. First, we need to segment the heart from the raw data. In our CVE System, we use 3-D region growing method to segment the heart. The basic formulation can be described as follows:

$$\begin{aligned}
&(\mathrm{a})\quad \cup_{i=1}^{n} V_i = V.\\
&(b)\quad V_i \quad is \quad a \quad connected \quad region, \quad i = 1, 2, \ldots, n.\\
&(c)\quad V_i \cap V_j = \varnothing \quad for \quad all \quad i = 1, 2, \ldots, n.\\
&(d)\quad P\left(V_i\right) = TRUE \quad for \quad i = 1, 2, \ldots, n.\\
&(e)\quad P\left(V_i \cup V_j\right) = FALSE, \quad i \neq j
\end{aligned} \tag{2.1}$$

Here, $P(V_i)$ indicates whether the region V_i meets the conditions above.

Fig.2 shows the result before and after the segmentation:

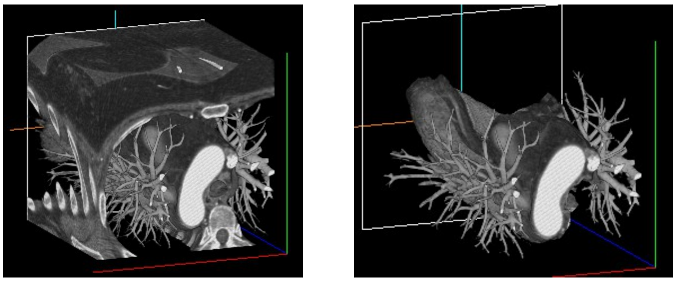

Figure2: The Result before and after the Segmentation

2.2.2 Euclidean Distance Transform(EDT)

There are numerous implementations of DFB-field method [1], which can solve the “hug problem” effectively. Nonetheless, it is not very fast in our CVE System due to the large size of DSCT data. In this paper, we use the Modified Distance from Boundary (MDFB) field metioned T. Saito et.al. [7] instead of the traditional DFB-fields. This method consists of two improvements: (1) by using squared DFB-distance, it doesn’t need to calculate the squared root and will reduce the time consumed in EDT process; (2) by using parallel DFB-field algorithm,it can decrease the time complexity from $O(m^2)$ to $O(\log^2 m)$ （m is the size of the volume data, size $m_1 \times m_2 \times m_3$）.

2.2.3 Centerline Extraction using Pairing Heap

Dijkstra algorithm is designed for finding single source shortest path in a graph. It was proposed by Edsger W. Dijkstra in 1956 and published after 3 years [8]. Suppose that S is a set of vertices, the shortest paths from the original source vertex s to each vertex in set S have been calculated. The algorithm selects the vertex $u \in V - S$ which has the minimum shortest path repeatedly, and relaxes all the edges leaving u. The pseudo code of Dijkstra algorithm can be expressed as below. Here we use a minimum heap H of vertices which is sorted by the d value to improve the original Dijkstra algorithm [9]:

```
FUNCTION Dijkstra(G, w, s)
initilization
S ← empty set,create vertex set
H ← V[G], create minimum heap
WHILE H is not empty set DO
    u ← H.extract_min(), extract the top node of minimum heap H
    S.add(u), add node u to set S
    FOR each vertex v belongs to the adjacencies of u, DO
        relax(u, v, w)
    END FOR
END WHILE
```

The time complexity of the Dijkstra algorithm depends on the specific implementation of the minimum heap. The most common implementation is the binary heap. Its time complexity is

$O((V+E)\lg V)$(V is the vertices number, E is the edges number) [9]. In this paper, we use the improved Dijkstra algorithm proposed by Ming Wang and use a different data structure called "pairing heap" to implement the minimum heap[5,10], which can accelerate the extraction process remarkably.

The pairing heap can be considered as a self-adjusting binomial heap [9], it is more straightforward to implement than Fibonacci heap, but as efficient as Fibonacci heap. An example of minimum-pairing heap is as shown in Fig. 3:

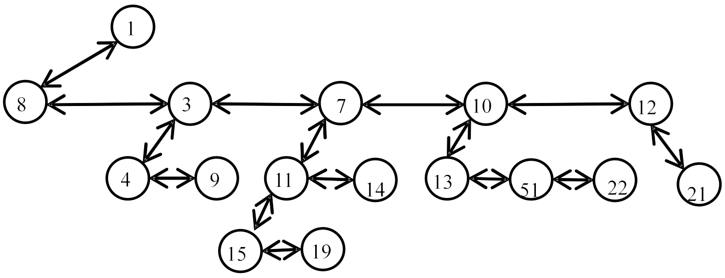

Figure 3: An Example of Minimum-pairing Heap

The node of pairing heap not only includes the elements used for sorting but also have three pointers, which are: the pointer pointing to its previous node(previous pointer), the pointer pointing to its left child node(left child pointer) and the pointer pointing to its right sibling node(right sibling pointer).

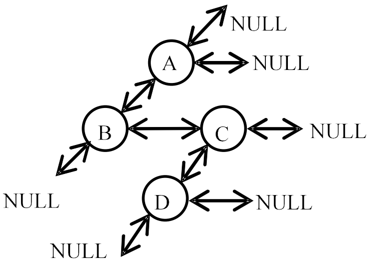

Figure 4: Pointers' Relationship among Pairing Heap Nodes

Take an example in Fig. 4, suppose that A is the root node of a pairing heap, and its previous pointer and sibling pointers are NULL; B is A's child and A's left child pointer points to B, because B is the first node of A's child list, and B's previous pointer points to its parent A; C is B's right sibling and B is C's left sibling, so C's previous pointer points to B; in the same way, D's previous pointer points to C and C's left child pointer points to D.

Pairing heap has these advantages (here we talk about the minimum-pairing heap) [11]: 1) the root node has the minimal weight element of the heap; 2) every heap node's weight is less than that of its child node; 3) there is no order between the nodes on the same level, but the first node is the smallest (this is determined by the merge operation).

The operation of pairing heap includes "merge", "insert", "decrease_key" and "extract_min". The "merge" operation is to merge two sub pairing heaps into one pairing heap. It's the basis of the other operations. The process of "merge" operation is to let the sub heap whose root has a bigger weight element be the child of another sub heap. The "insert" operation is to add a new node into the pairing tree. The process is to take the new node as a sub heap which has only one node, then merge the original heap with this sub heap. The “decrease_key” operation is to change the weight of one node. Its process is to remove the node and its children as a sub heap from the pairing heap, change its value and merge it into the original heap. The “extract_min” operation is a little complex, first we remove the minimal node(root node) of the pairing tree and cut down its child list into sub heaps, then take the first sub heap as a base heap, and merge the other sub heaps into the base heap one by one. For more details of pairing heap, may read the paper written by ML Fredman et.al.[10,11].

The procedures of our centerline extraction can be characterized in Fig. 5:

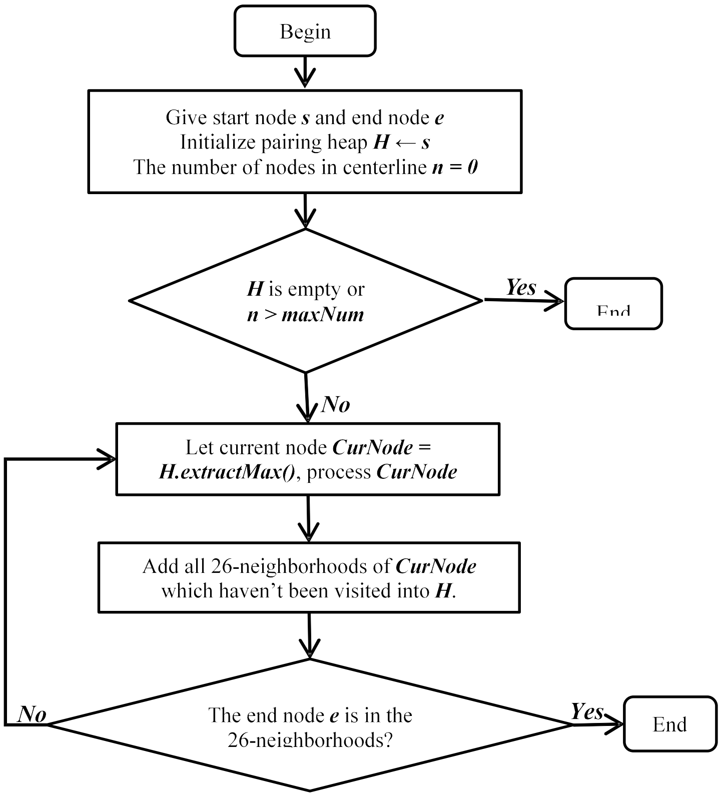
Begin
Give start node *s* and end node *e*
Initialize pairing heap *H* ← *s*
The number of nodes in centerline *n = 0*
H is empty or *n > maxNum*
Yes
End
No
Let current node *CurNode* = *H.extractMax()*, process *CurNode*
Add all 26-neighborhoods of *CurNode* which haven’t been visited into *H*.
No
The end node *e* is in the 26-neighborhoods?
Yes
End

Figure 5: Flow Chart of Centerline Extraction Algorithm

In Ming Wang’s algorithm [5], the termination condition is that all the nodes are processed. This algorithm gets the longest centerline, but in our CVE system, we just need to get a part of the centerline and do not need to calculate all the nodes, and we add two termination conditions for our centerline extraction algorithm: the end node and the max number of nodes in the centerline, which will save some time in the extraction process.

3.Experiments and Results

We test the Dual Source CT dataset whose size is 512*512*354 provided by Shanghai Xinhua Hospital. The dataset includes 30 patients. The environment of our test is a normal PC(CPU: Intel i5 750 @2.67GHz, GPU: NVIDIA GeForce GTX 465, RAM: 6GB).

Fig. 6 shows the result of our centerline extraction method, the left is in pseudo-color display mode, and the right is in linear display (gray) mode:

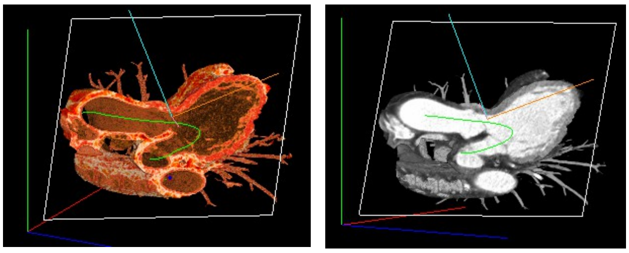

Figure 6: Result of Centerline Extraction Using Our Method.

In order to measure the extraction speed of our method, we do some experiments to compare with other method [1,5]. Table 1 shows two experiments' results, and we can see that our method can greatly accelerate the centerline extraction process:

	Exp1	Exp2
Other Method	63121ms	90496ms
Our Method	13182ms	14882ms

Table 1: Extraction Time of two Methods

4.Conclusion

This paper describes the centerline extraction method for cardiovascular virtual endoscope system, and introduce a fast centerline extraction method based on distance mapping method. A new data structure pairing heap is employed to improve the Dijkstra extraction process. Compared with other methods, our method increases the extraction speed significantly, but our method is just appropriate for structure without branches. In the future, we will improve our method and strengthen it to extract centerline for more complicate shapes.

References

[1] J Li, X Yang, LP Yao, K Sun. *3-D Centerline Extraction for Virtual Cardiovascular Endoscopy* [C], 2014 11th International Computer Conference on Wavelet Active Media Technology and Information Processing, Chengdu, 19-21 December, 2014.

[2] H Blum, *Biological shape and visual science (Part I)*[J]. Journal of theoretical Biology. 1973, 38(2): 205-287.

[3] O Wink, WJ Niessen, *MA Viergever. Fast delineation and visualization of vessels in 3-D angiographic images*[J]. IEEE Transactions on Medical Imaging, 2000, 19(4):337 - 346.

[4] ND Cornea, D Silver, P Min. *Curve-skeleton properties, applications, and algorithms*[J]. Visualization & Computer Graphics IEEE Transactions on, 2007, 13(3):530--548.

[5] M Wang, Z Liang, Q Ke, L Ke, I Bitter, A Kaufman, et al. *Automatic Centerline Extraction for Virtual Colonoscopy* [J]. IEEE Transactions on Medical Imaging, 2002, 21(12):1450--1460.

[6] L. Hong, A Kaufman, YC Wei, *A Viswambharan, M Wax, Z Liang. 3D virtual colonoscopy*[C]//Biomedical Visualization, IEEE Computer Society, 1995:26-32,83.

[7] T Saito, JI Toriwaki. *New Algorithms For Euclidean Distance Transformation Of An N-Dimensional Digitized Picture With Applications* [J]. Pattern Recognition, 1994, 27(94):1551–1565.

[8] E W. Dijkstra. *A Note on Two Problems in Connexion with Graphs* [J]. Numerische Mathematik, 1959, 1(1):269--271.

[9] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. *Introduction to Algorithms* [M], 3rd Edition, The MIT Press, 2009, 658-662.

[10] ML Fredman, R Sedgewick, D D Sleator, RE Tarjan. *The pairing heap: A new form of self-adjusting heap* [J]. Algorithmica, 1986, 1(1):111-129.

[11] LG Zhang, JY Fang, PW Shen, *An Improved Dijkstra Algorithm Based on Pairing Heap* [J], Journal of Image and Graphics, 2(5):419-422.