

A New Algorithm of Shape Boundaries Based on Chain Coding

Xin Zhao¹, Jun Zheng², Yan Liu^{*3}

¹Shanghai Zhongqiao College, Shanghai, China

²School of Computer Science and Software, Engineering, East China Normal University, Shanghai, China

³School of Computer Science and Software, Engineering, East China Normal University, Shanghai, China

¹ zhaoxin203@126.com, ³ yliu@cc.ecnu.edu.cn

Abstract; A new method to obtain connected component in binary images is presented. The method uses DFA automaton to obtain chain code and label the component boundary. It is theoretically proved that the algorithm improves the image encoding efficiency closer to the lowest time consumption.

Keywords-image shape; boundary tracing; chain code; automaton; boundary processing

1. Introduction

Connected component obtaining in a binary image is one of the most fundamental operations in pattern recognition and image analysis. The purpose of labeling is to treat the individual connected components of a digitized image as separate objects.

Rosenfeld presented tracing idea in 1978[1]. The subsequent work[2,3] are effective, but have less efficiency than Rosenfeld .The well-known TP algorithm is proposed in [4], but some errors including losing inner contours, connectivity and traversing some of the boundary pixels more than one time can be found.

Several algorithms based on label equivalences have been proposed. Chang et al. [5] outperforms other methods using label equivalences and ordinary computers and thus is picked out to represent existing methods based on label equivalences with more than two passes.

The report of Seong-Dae Kim et al. described a two-step run-length algorithm [6]. The method computes chain code from run-length coding representation. It preserves the connectivity information between runs and never loses the inner or outer contours.

A fast algorithm is proposed in this paper. It uses DFA automaton to obtain chain code and label the component boundary specifically for binary image. The algorithm makes full utilization of the automaton in the labelling method to avoid tracing contours more than once. Our algorithm improves the current binary image encoding efficiency regarding algorithmic complexity and memory usage.

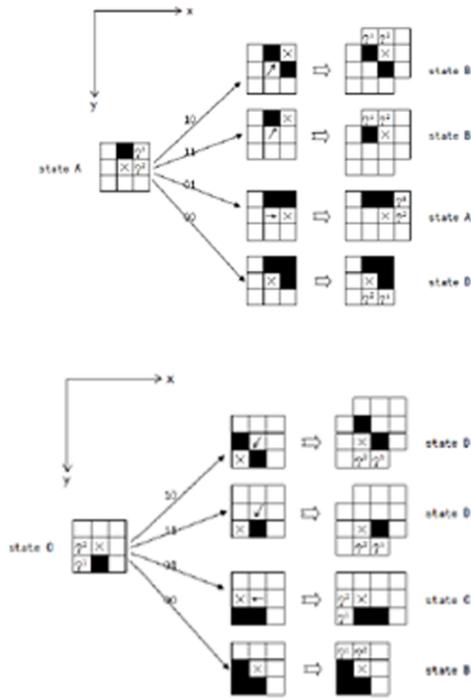
2. Proposed Algorithm

Freeman introduced chain code to represent digital curves [7] first. Ernesto Bribiesca used a derivative of the Freeman chain code for shape representation and defined vertex chain code (VCC) [8] in 1999. Then Bribiesca presented a new measure of tortuosity for 2D curves which based on geometric structure of Slope Chain Code (SCC)[9].

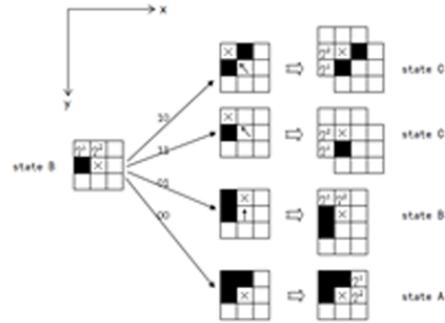
It has been proved that chain code provide a compact representation and preserve all the information of the image [10, 11]. Bribiesca also proved that chain coding data has higher ratio of lossless compression [12].

The deterministic finite automaton (DFA) is suitable for modelling the collaboration entities in a collaboration session. A finite automaton is a five-tuple notation $A = (Q, \Sigma, \delta, S, F)$.

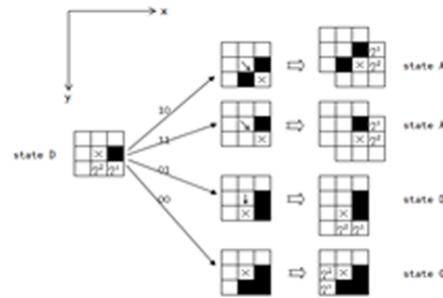
A is the name of the automaton. Q is the finite set of states of the automaton. Σ is the finite set of input symbols. $\delta : Q \times \Sigma \rightarrow Q$ is the transition function, which takes as parameters as state from Q and a symbol(from Σ ,and returns a state in Q for deterministic Finite Automata(DFA). $S \in Q$ is the start state. $F \in Q$ is the finite set of final or accepting states, which is a subset of Q .



The transition of state A is given in fig.2 (a). Because of the space symmetry of the internal state of the automaton, the state transition graph of State B, C and D can be obtained by rotate the state transition graph of State A by 90°, 180°, and 270° as shown in (b),(c),(d) of fig.2.



(a)Transition of state A (8-connected) (b) Transition from state B (8-connected)



(c)transition from state C (8-connected) (d)transition from state D (8-connected)

In DFA, for each $Q \times \Sigma$ of state and input there is a unique next state given by the transition function.

For the usage of automaton, pixels on image contour are divided into 4 states. All states can transform to related states under given condition. However, the automaton traverses on the outside boundary of the image.

2.1 The 8-Connected Automaton

The new examining sequence is proposed and pitched as in Fig.1. Assuming position with arrow is the current position of tracer, which is a white pixel on the outside boundary. The points labeled by 'a', 'b' should be examined in turn until the automaton gets the next state. That is, the automaton only uses four points - the black point, the point with arrow, and the points with 'a' and 'b'. Point with 'a' must be examined first. And only under special condition, point with 'b' will be examined later.

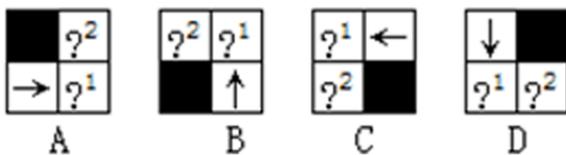


Figure1. The four states of 8-connected automaton.

So value of points 'a' and point 'b' are input of DFA. The input can be '0x', '10' or '11' according to the following definition:

- 0x: a=0(a is black, b is not examined)
- 10: a=1, b=0(a is white, b is black)
- 11: a=1, b=1 (a is white, b is white)

Figure2. Automaton transition of 4 states.

Procedure of transition of state A in 8-connected image:

When current state is state A, the automaton first examines the point with 'a' (pixel at coordinates (x+1, y)). If it is black, change current state to state D.

If it is white, examine the point with 'b'. If pixel where 'b' located is black, state is still state A. And automaton goes forward to coordinates (x+1, y). If pixel with 'b' is white, change state A to state B, and automaton goes forward to coordinates (x+1, y-1).

According to the transition image, the automaton is given in fig 3.

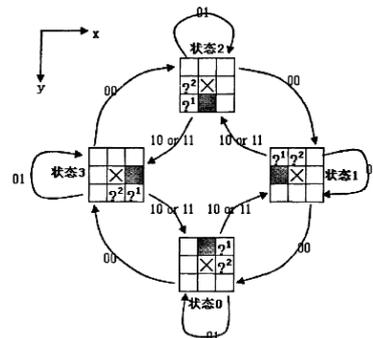


Figure3. DFA of 8-connected image tracer.

In addition, chain code can be obtained the same time. Freeman 8-direction code is Output according to tab.1.

Table1. Output of Freeman 8-direction code

	State A	State B	State C	State D
State A	0	1	-	-
State B	-	2	3	-
State C	-	-	4	5
State D	7	-	-	6

2.2 How to Find a New Contour

To find a new contour and start the automaton needs to examine whether current situation complies with state A — upper pixel is black and lower pixel is white. Starting procedure of automaton with 8-connected image is shown in fig.7.

To terminate the automata is simple, it only needs to analyze whether the automata have returned to the starting point. If automata traces back to the starting position again, it will terminate and begin to find a new contour - the next position with state A.

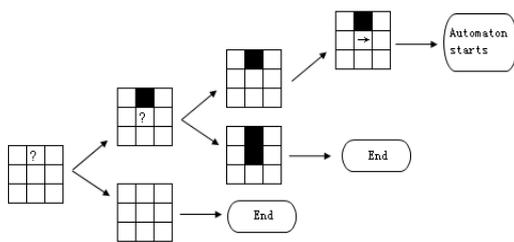


Figure4. Starting procedure of automaton.

3. Performances Comparison

The performances of algorithms are shown in tab 3. (AK stands for Kim algorithm; AFC stands for Fuchang method; AA_e stands for algorithm using automaton) In these figures, the size of the test image (M pixels) is plotted along the horizontal axis, while the average processing time (ms) of each method is plotted along the vertical axis.

Table2. Performances of compared methods

Document type	Image size(M pixels)	Algorithms		
		AA-e	AFC	AK
		Average processing time(ms)		
Halftone picture	0.16	7.1	9.54	10.23
	0.64	20.8	26.3	31.4
	2.56	85.2	115.85	128.7
Headlines	0.16	8.03	11.2	12.01
	0.64	19.89	28.1	32.63
	2.56	66.5	91.7.	103.3
Legacy document	0.16	6.4	9.2	10.8
	0.64	20.7	27.42	33.4
	2.56	67.6	90.1	107.1
Newspaper	0.16	7.5	10.9	12.01
	0.64	22.8	28.64	36.05
	2.56	70.8	92.37	115.2

4. Conclusion

This paper presents a new method to obtain connected component in binary images. The time consumption in this method is, in expense of additional work memory, this method is closer to the lowest time consumption.

The algorithm makes full utilization of the automaton in the labeling method to avoid tracing contours more than once. The connectivity information is fully preserved and the inner and outer contours can be distinguished. The new algorithm only labels west side of the outer contour and east side of the inner contour and it is capable of generating all types of chain code after tracing the contours.

It is theoretically proved that our algorithm improves the current binary image encoding efficiency regarding algorithmic complexity and memory usage. In experiments on six types of images of various sizes, we compare our method with other algorithm. The results show that our algorithm outperforms that in all six types of binary image in terms of computational speed especially for complex image.

References

- [1] Rosenfeld A. Algorithms for image/vector conversion [J]. Computer Graphics, 1978(12): 135-139.
- [2] Hermilo S C, Bribiesca E, Ramon M R D. Efficiency of chain codes to represent binary objects [J]. Pattern Recognition, 2007, 40: 1660-1674.
- [3] Merrill R D. Representation of contours and regions for efficient computer search [J]. Comm ACM, 1973, 16 (2): 69-82.

- [4] Pavlidis T, Algorithms for Graphics and Image Processing [M]. Rockville: Computer Science Press, 1982.
- [5] F. Chang, C.J. Chen, C.J. Lu, A linear-time component-labeling algorithm using contour tracing technique, *Comput. Vis. Image Und.* 93 (2004) 206–220.
- [6] Kim S D, Lee J H, Kim J K, A new chain-coding algorithm for binary images using run-length codes [J]. *CVGIP*, 1988, 41: 114-128.
- [7] H. Freeman, Computer processing of line drawing images, *Computational Surveys* 6 (1974) 57-97.
- [8] Bribiesca E, A new chain code [J]. *Pattern Recognition*, 1999, 32: 235-2
- [9] Bribiesca E, A measure of tortuosity based on chain coding, *Pattern Recognition* ,Volume 46, Issue 3, March 2013, Pages 716–724
- [10] Liu Y K, Wei W, Wang P J. Compressed vertex chain codes [J]. *Pattern Recognition*, 2007, 40: 2908–2913.
- [11] Hermilo S C, Bribiesca E, Ramon M R D. Efficiency of chain codes to represent binary objects [J]. *Pattern Recognition*, 2007, 40: 1660-1674.
- [12] Hermilo S C, Bribiesca E, Ramon M R D. Efficiency of chain codes to represent binary objects [J]. *Pattern Recognition*, 2007, 40: 1660-1674.
- [13] Gu G Q, Chen K, Region-Labeling-Automata for Images in Square, Triangular and Hexagonal Lattice, *Advances in Computer and Information Technology, The Journal of Three Dimensional Images*, 17 (2003)161-165.