

An Improved Parallel Thinning Algorithm

Lei Huang, Genxun Wan, Changping Liu
Institute of Automation, Chinese Academy of Sciences, P. R. China
lei.huang@mail.ia.ac.cn

Abstract

This paper describes an improved thinning algorithm for binary images. We improve thinning algorithm from the fundamental properties such as connectivity, one-pixel width, robust to noise and speed. In addition, in order to overcome information loss, we integrated the contour and skeleton of pattern and proposed the threshold way. Some fundamental requirements of thinning and the shape of pattern are preserved very well. Algorithm is very robust to noise and eliminate some spurious branch. Above all, it can overcome the loss of information in pattern. Experimental results show the performance of the proposed algorithm.

1. Introduction

In the past several decades, many thinning algorithms have been developed [1-4]. Some classical algorithms have obtained good results, but there are still some deficiencies. For example, some fully parallel algorithms cannot preserve the connectedness of an image. In Datta's algorithm [1], some basic properties of thinning are ensured. But it uses multi-pass iterative and is not fully parallel algorithm. Han et al [2] proposed a fully parallel which is based on the information of in 5×5 mask. But it costs much time. In addition, all these algorithms are not efficient in some peculiar case, such as the loss information of pattern.

In this paper, we propose an improved parallel thinning algorithm, which is fast and efficient for handwritten character recognition. Some new rules are used to solve the problem of dis-connectivity. And contour information is added to overcome the possible loss of information produced by thinning. Using this algorithm, some basic properties of thinning like one pixel thickness and connectivity are ensured. The skeleton is close to the medial axes, which preserve the topology of image. This paper is organized as follow. In section 2, the proposed thinning algorithm is described. At the same time, the new rules and templates are introduced. In section 3, the threshold method will be discussed in detail. In section 4, the results are compared with those of other algorithms.

2. The proposed Algorithm

Though there is no general agreement in literature on the exact definition of thinness, a good thinning algorithm should preserve some basic properties. Some basic definitions are prescribed here; they followed some publication paper [3,6]. The binary image is represented by black pixels and white pixels, the number 1 and 0 denote black and white pixel respectively.

Definition 1: The 4-connected component of non-object pixels which contain the top and bottom rows, and the rightmost and leftmost columns of the image is background; and any other four-component of non-object pixels is a hole.

Definition 2: A skeletal leg is a limb of thickness one with one end not connected to anything. This pixel not connected to anything is called an end pixel. In fact, the end pixel has exactly one 8-neighbors.

2.1 Elimination Rules

Elimination rule is the kernel of a thinning algorithm, which decides the performance the algorithm. Datta et al. [1] use four 1×3 templates and one 3×3 window. In Han et al's algorithm, they use the weight-values (the sum of 8-neighbors) of black pixel. In our algorithm, we also use 3×3 window. All kinds of relation (256) formed by 8-neighbors of object pixel have been considered. From these cases, a group of elimination rules can be obtained. All rules are given in Fig. 1 (blank denote 0). These rules were classified according to the amount of black pixels in the 8-neighbors. The first column denotes the number of black pixel in 8-neighbors of object black pixel and the second column shows the elimination rules. Though many windows are same with Han's rules [2], there are essential differences between them. In Han's algorithm, they depend on not only the 8-neighbors, but also the weight-values of 8-neighbors pixels. In fact, they use the information in 5×5 window. In our algorithm, we use only information of 3×3 window. In addition, some elimination rules that may result in the distortion of shape and the loss of connectivity are removed.

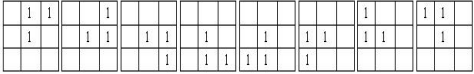
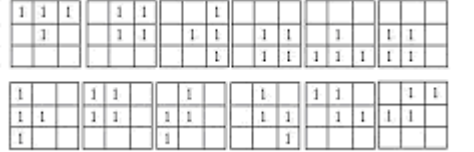
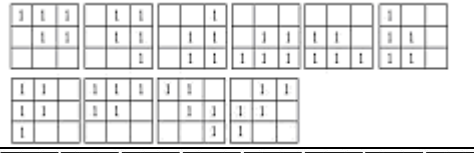

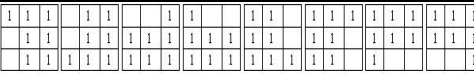

Amount	Elimination Rules
0	Never
1	Never
2	
3	
4	
5	
6	
7	
8	Never

Fig 1. New elimination rules, blank denote the white pixel

2.2 Two-Pixel-Width Process

All the rules are applied simultaneously to each pixel. In some peculiar cases, these cannot be performed very well. A two-pixel-width or even pixel width in horizontal or vertical direction may be deleted, which would cause the loss of connectivity of object pattern. For example, a two-pixel-width rectangular pattern will disappear completely. In order to keep up the connectivity, these pixels should not be deleted. On the other hand, if all two-pixels-width are retained, the skeleton would not be one pixel width. In Datta's algorithm [1], since a single template is applied in each pass and output of each pass is passed on to the next pass, the connectivity and one-pixel width is guaranteed. Jang et al use ten 3×4 , 4×3 and 4×4 windows to resolve this problem [3]. In our algorithm, we modify Jang's templates. New templates are presented in Fig. 2. If the object pixel matches one of the templates in Fig. 2, the pixel should be preserved.

Summarize above paragraphs, we proposed our thinning algorithm:

- 1) Create the lookup table.
- 2) The index number is calculated for each object pixel.

- 3) Use the index number and lookup table to decide if the pixel is eliminable. If the pixel is not eliminable, go to step 5.
- 4) Get the width of object pixel. If it is not two-pixel-width, delete it, otherwise, use the 3×4 , 4×3 and 4×4 templates to decide if the pixel should be preserved. If the object pixel has not the requirement of preservation, delete it, or else preserve it.
- 5) Repeat 2-4 until no pixel can be eliminated.

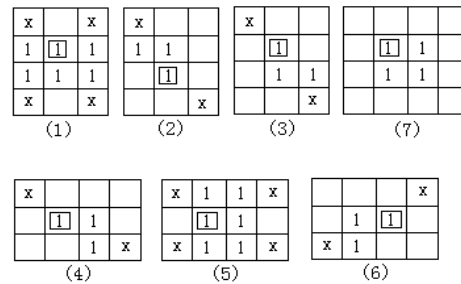


Fig 2. The preserved template used in our algorithm, blank denote 0, "x" are don't care

3 Compensate information loss

There is no general agreement in literature on the exact definition of thinness. As illustrated in Fig. 3, the larger black block whose length and width are both greater than 1, is usually changed to a line or a dot. In the field of character recognition, this is not reasonable in some case. For example, in our handwritten numeral recognition system, owing to the unconstraint of handwritten character, there are lots of samples contain the black block, such as the top of number nine and bottom of number six. Generally, the blocks are thinned to a dot or a line, which may result in loss of information and misidentification. In order to retain information of pattern, we integrate the result of thinning with the contour of pattern. The skeleton of larger black will be not a dot or line, but rectangle (Fig. 3(e)).

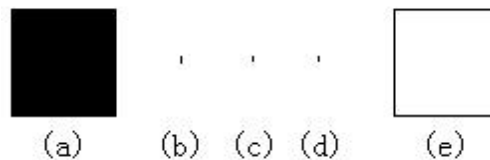


Fig. 3 The thinning result of image (a) Original image, (b) Datta's algorithm, (c) Han's algorithm, (d) Proposed algorithm don't consider information loss, (e) Proposed algorithm consider information loss

Jang and Chin [3] have introduced a measure (m_m) to quantify the closeness of the extracted skeleton to the ideal medial axis. They have defined m_m by

$$m_m = \frac{\text{Area}[S']}{\text{Area}[S]}$$

where the Area[.] is an operator that counts the number of pixel; and S, S' denotes the input image and the maximal digital disk (included in S) centered at all the skeletal pixels respectively. Enlightened by this, we incorporate the contour of image information with the skeleton and propose threshold method to resolve the problem of information loss.

As we know, the skeleton should be near the ideal medial axis and one pixel width. Ideally, the pixels of the contour are as twice much as these of the skeleton. When there is a larger black block is changed to line or dot, the pixels in the skeleton will be reduced tremendously. We define a parameter R by

$$R = \frac{\text{Area}[\text{skeleton}]}{\text{Area}[\text{contour}]}$$

If R is lower than the threshold, we will judge if there is information loss. If lots of information is lost, we will retain the contour of image. The selection of threshold is very important. If selected threshold is too big, some images which haven't lost information will be contained, on the contrary, if selected threshold is too low, some images which have lost information will be discarded. In our experiment, we take threshold as 0.4. The proposed algorithm is described as follows:

- (1) Get the contour of image and count the number of pixel in the contour.
- (2) The image was thinned by above proposed algorithm. Based on the result of thinning, we can get the skeleton pixel's number.
- (3) Calculate parameter R. If the R is larger than threshold, return the result of thinning and stop.
- (4) If a larger black block is changed to line or dot, return the contour of image, or else return the thinning result.

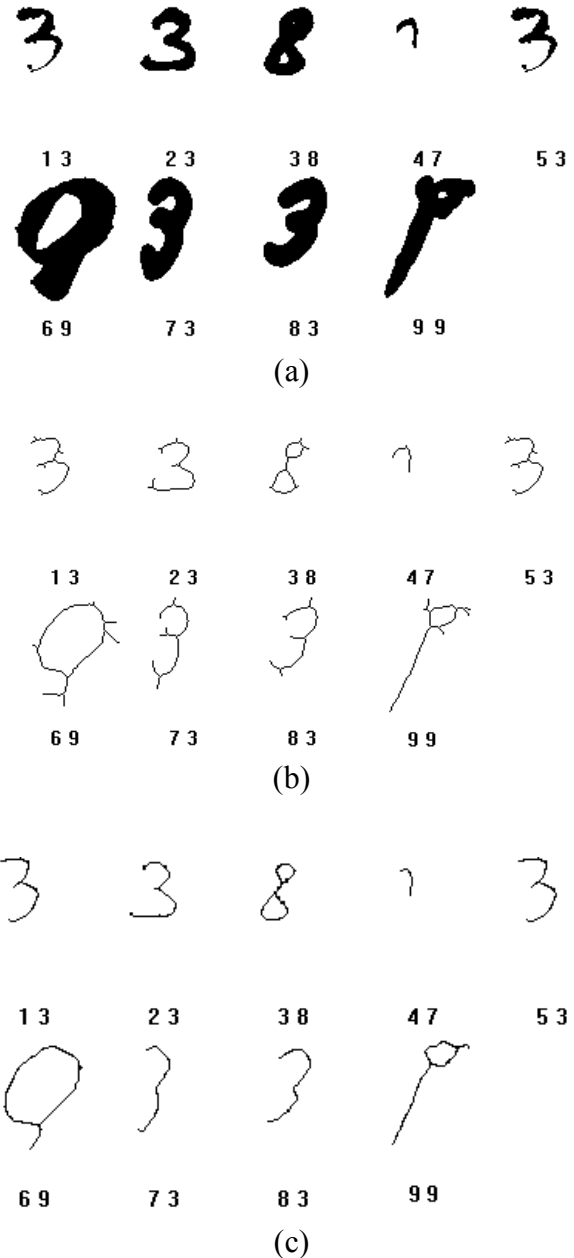
4. Experimental results

When evaluating the quality of a thinning algorithm, we mostly consider the connectivity preservation, the width of skeleton and robustness to noise. In this paper, we will consider another important aspect: information preservation.

As illustrated in Fig. 4, there are some handwritten number images and thinning results. We use Datta's, Han's and proposed algorithm respectively. The results prove that proposed algorithm is very efficient. Compared proposed algorithm (Fig.4 (b)) with Datta's algorithm (Fig.4 (d)), we find that our algorithm is more robust to noise. Many spurious branches in Datta's results can be eliminated. Han's algorithm (Fig.4(c)) can eliminate the

spurious branch too, but the skeleton is not one pixel width and not smooth.

The results are presented in Fig. 5(b) and (c). As illustrated in Fig.5 (b), the shapes of original number images have been destroyed completely. All the numbers are very similar and misidentified. In Fig.5 (c), this deficiency has been eliminated and the shapes of numbers have been preserved. The results indicated that our algorithm is very valid.



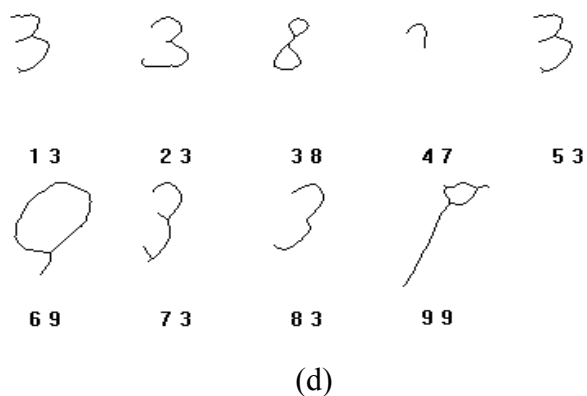


Fig.4 Comparison of result of different methods (a) handwritten number, (b) Datta's algorithm, (c) Han's algorithm, (d) our algorithm.

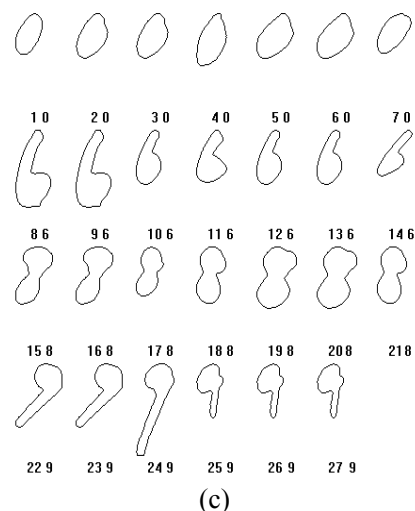
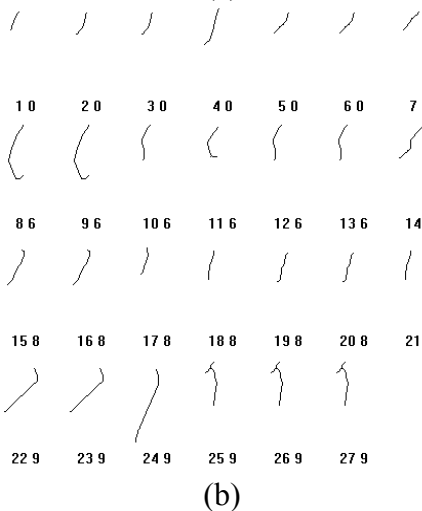
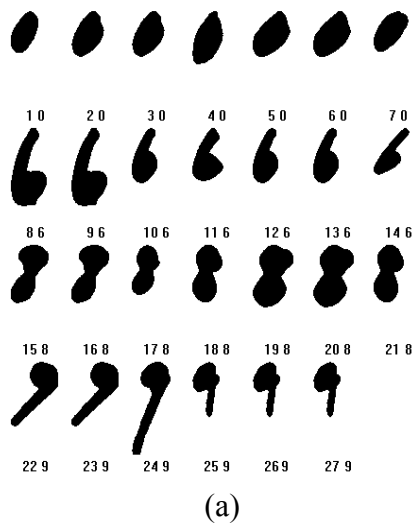


Fig. 5 Compensate lost information (a) Handwritten number (b) The result of proposed algorithm don't compensate the lost information, (c) The result of proposed algorithm compensate the lost information



5. Conclusions

This paper improves the existent fully parallel thinning algorithm. In order to preserve the connectivity, some 3×4 , 4×3 and 4×4 masks are added as preservation template. In addition, we add some extra operation to overcome the loss of information, which is necessary in some fields. The results of experiment indicate that our algorithm is very efficient. The skeleton is close to the medial axis and very robust to noise of contour. In addition, the important information of pattern are reserved. In future work, it is planned to give location of information loss and incorporate skeleton and contour.

6. References

- [1] A.Datta and S.K. Parui, "A Robust Parallel Thinning Algorithm for Binary Images," Pattern Recognition, Vol.27, No.9, pp.1181-1192, 1994.
- [2] N.H.Han, C.W.La, and P.K.Rhee, "An Efficient Fully Parallel Thinning Algorithm," in Proc. IEEE Int. Conf.Document Analysis and Recognition, Vol. 1, pp.137-141(1997).
- [3] B.K.Jang and R.T.Chin, "One-Pass Parallel Thinning Analysis, Properties, and Quantitative Evaluation," IEEE Trans. Patt. Anal. Machine Intell. , Vol. 14, No.11, pp.869-885, 1992.
- [4] S.Suzuki, K.Abe, "Binary picture thinning by an iterative parallel two subcycle operation," Pattern Recognition 20 (3) (1987) 297-307