Neural Processing Letters (2007) 25:49–62
DOI 10.1007/s11063-006-9030-9

Binary Image Thinning Using Autowaves Generated by PCNN

LIFENG SHANG*, ZHANG YI and LUPING JI
Computational Intelligence Laboratory, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, P. R. China.
e-mail:superkangaroo@163.com

Abstract. This paper proposes a novel binary image thinning algorithm by using the autowaves generated by Pulse Coupled Neural Network (PCNN) . Once the autowaves travelling in different directions meet, the PCNN delivers the thinning results. Four meeting conditions are given for autowaves meeting. If a neuron satisfies one of the four conditions, the pixel corresponding to this neuron belongs to the thinning result. Moreover, the specification of the PCNNs parameters is given, which makes the implementation of the proposed thinning algorithm easy. Experimental results show that the proposed algorithm is efficient in extracting the skeleton of images (such as Chinese characters, alphabet letters, numbers, fingerprints, etc.). Finally, a rate called "R_{MSkel}" is given to evaluate the performance of different thinning algorithms, and comparisons show that the proposed algorithm has higher "R_{MSkel}" and costs less time.

Key words. Medial Axis Transformation, Pulse Coupled Neuron Network, Binary Images Thinning, Autowaves, Grassfire

1. Introduction

Image thinning plays an important role in image processing as it simplifies object representation and pattern analysis. The skeleton is defined via the medial axis transformation (MAT) proposed by Blum[17]. In this definition, the skeleton of an image is defined as the set of the centers of all maximal inscribed discs. An essential property of skeletons in digital space is that they include all or part of these centers, defined with respect to Euclidean or other distances. The MAT definition is equivalent to the intuitive definition in terms of "prairie fire simulation" [17]. However, direct implementation of MAT is expensive computationally. So there has been considerable interest in finding new methods to rapidly thin images.

Numerous algorithms have been proposed for extracting the skeleton of digital object over decades [7–9, 15]. Zhang and Suen proposed a fast parallel algorithm for thinning digital patterns [20]. The process of this algorithm is repetitively removing the pixels of the edge, until only the central pixels remaining. Recently, some new techniques (such as wavelet and neural networks) have been introduced into image thinning. Tang and You proposed a wavelet-based scheme to extract

* Author for correspondence.

skeleton of Ribbon-like shape [18]. Krishnapuram and Chen applied recurrent neural network to image thinning [14]. Altuwaijri and Bayoumi used the self-organizing neural network to thin digital images [1]. In this paper, autowaves generated by Pulse Coupled Neural Network (PCNN), a recently developed technology, is applied to image thinning.

Johnson and Padgett introduced a new neural network-PCNN based on the linking model developed by Eckhorn et al. [4,5], Johnson and Padgett [10] and Johnson and Ritter [11]. Synchronization and autowaves are two basic properties of PCNN. Autowaves have been used to solve some interesting problems such as maze problem [3] and shortest path problem [16]. The proposed algorithm applied the autowaves generated by PCNN to images thinning based on the intuitive definition of skeleton-“prairie fire simulation”. In the proposed algorithm, autowaves generated by PCNN are used to mimic the grassfire, and four autowaves meeting conditions are given to obtain the locus where fire fronts meet, and the locus is the thinning result.

Meeting conditions are crucial for the proposed algorithm, which directly affect the final thinning result. Four meeting conditions are given in Section 3. Generally, the PCNN does not need a training process as other traditional artificial neural networks. However, it needs the specification of its parameters in order to solve a specific task. The specification of parameters of PCNN for thinning is particularly discussed in Section 2. In Section 4, a rate called “R_{MSkel}” is used to evaluate the performance of different thinning algorithms.

This paper is organized as follows. In Section 2, the PCNN for binary image thinning is described. In Section 3, the proposed thinning algorithm is given. Experiments are illustrated in Section 4. Finally, the conclusions of the paper are presented in Section 5.

2. PCNN Modal and the Specification of its Parameters

First, the basic pulse coupled neuron model is briefly reviewed. PCNN is a simplified model of the biological visual system, and has been widely used in image processing [2,6,12,13,19]. The internal structure of a single neuron $N_{i,j}$ we used for the network is shown in Figure 1.

In Figure 1, $I_{i,j}$ is the input from external sources and $Y_k (1 \leq k \leq 4)$ is the input from adjacent neurons. $F_{i,j}$ and $L_{i,j}$ are feeding input and linking input of the neuron $N_{i,j}$, respectively. In the modulation field, the internal activation $U_{i,j}$ is calculated as follows:

$$U_{i,j} = F_{i,j}(1 + \beta_{i,j} L_{i,j}), \tag{1}$$

where $\beta_{i,j}$ is the linking strength. In the pulse generator, $U_{i,j}$ is compared with the neuron’s threshold $\theta_{i,j}$, if $U_{i,j}$ is greater than $\theta_{i,j}$, the neuron produces a spike, and its threshold is immediately raised to prevent it from firing again. The output

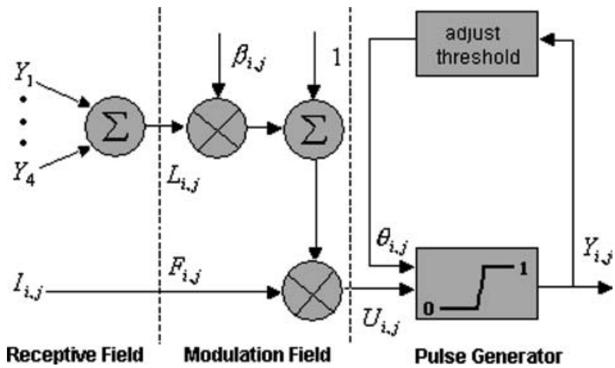

Figure 1. The structural model of the pulse coupled neuron $N_{i,j}$.

$Y_{i,j}$ is represented as follows:

$$Y_{i,j} = \begin{cases} 1, & U_{i,j} > \theta_{i,j}, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

When the PCNN is applied to image thinning, it is a single layer two-dimensional array of laterally linked pulse coupled neurons [12]. Each pixel corresponds to a unique neuron and each neuron is connected with its surrounding neurons. For example, the neuron $N_{i,j}$ is associated with the pixel $P_{i,j}$. Some important features of the PCNN for image thinning are listed below.

1. The feeding input to the neuron $N_{i,j}$ is $T_{i,j}$, which is the intensity value of pixel $P_{i,j}$.
2. The linking neighborhood of the neuron $N_{i,j}$ is shown in Figure 2. The linking input is calculated by the following equation:

$$L_{i,j} = \sum_{k \in N^h_{i,j}} Y_k, \tag{3}$$

   where $N^h_{i,j}$ is the four-neighbor field of the neuron $N_{i,j}$ (The four neighbors of neuron $N_{i,j}$ are neurons: $N_{i,j-1}$, $N_{i-1,j}$, $N_{i,j+1}$, and $N_{i+1,j}$).
3. The threshold $\theta_{i,j}$ is initialized by T^h_A, and then is updated by

$$\theta_{i,j} \leftarrow \theta_{i,j} + Y_{i,j}(T^h_B - T^h_A), \tag{4}$$

   here T^h_A and T^h_B are two const values.

In the next section, the specification of the parameters of the PCNN will be particularly discussed. Once the parameters (T^h_A, T^h_B, and β) are specified, the autowaves travelling process can be determined.

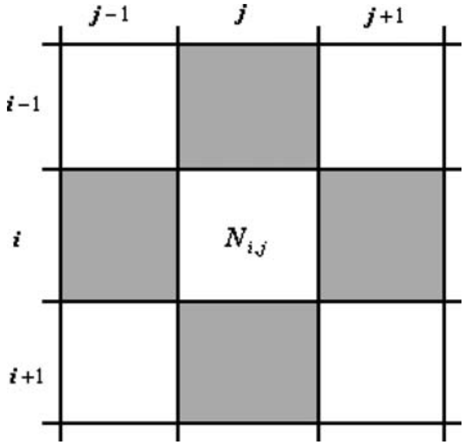

Figure 2. Linking neighborhood of a pulse coupled neuron $N_{i,j}$. (The neurons: $N_{i,j-1}$, $N_{i-1,j}$, $N_{i,j+1}$, and $N_{i+1,j}$ are called the four neighbors.)

2.1. SETTING PARAMETERS

In this work, the parameters: T_A^h, T_B^h, and β should be determined. Consider an image consisting of two regions: object and background. Object pixels form R and background pixels form B. Let T_R and T_B be the intensity values of the object and the background pixels, respectively. To facilitate the implementation of the proposed algorithm, the intensity value T_R is replaced by the constant one, and the intensity value T_B is replaced by the constant two.

In the proposed algorithm, autowaves generated by PCNN is used to mimic grassfire, so producing autowaves with the same travelling speed becomes very important. To produce autowaves with the same travelling speed, the PCNN should have the following three properties:

(1) At the first firing time, background neurons (neurons corresponding to the background pixels) fire and all the object neurons (neurons corresponding to the object pixels) don't fire.

(2) The neuron with T_R as its feeding input fires if and only if it receives linking input from its adjacent neurons. This property ensures that all object neurons can be ignited if they receive linking inputs. Furthermore, this property combined with the first property ensures that only contour neurons (neurons corresponding to the contour pixels of the original image) are ignited at the second firing time.

(3) The fired neurons should not fire again. That is, if a neuron has fired, it will never fire again.

Therefore

$$\begin{aligned} &T_R = 1 < T_A^h < T_B = 2, \\ &T_R(1+\beta) = 1+\beta > T_A^h, \\ &T_B(1+4\beta) = 2+8\beta < T_B^h \end{aligned} \tag{5}$$

Obviously,

$$
\begin{aligned}
&T_A^h = (T_R + T_B)/2 = 1.5,\\
&\beta = T_A^h / T_R = 1.5, \qquad (6)\\
&T_B^h = T_B(1 + 4\beta) + 1 = 15,
\end{aligned}
$$

satisfies (5). By (6), the parameters of the PCNN for image thinning can be determined easily. In the next section, an example is used to illustrate the autowaves travelling process of the PCNN for image thinning with its parameters determined by (6).

2.2. AN EXAMPLE ILLUSTRATING THE AUTOWAVES TRAVELLING PROCESS

In this section, an example is given to illustrate the autowaves travelling process of the PCNN for image thinning. Figure 3(a) shows the original image in which the intensity values of R and B are 7 and 164, respectively. Before injected to PCNN, the intensity value of R is changed to one, and that of B is changed to two. By using (6), the value of the network parameters are $T_A^h = 1.5$, $\beta = 1.5$, and $T_B^h = 15.0$. The firing process is shown in Figure 3. Figure 3(b) shows firing neurons at the first firing time. Obviously, at this time all the background neurons fire and all the object neurons don't fire. Figure 3(c) shows firing neurons at the second firing time. Obviously, at the second firing time, only contour neurons are ignited. Figure. 3(d),(e), and (f) shows firing neurons at the ith, $(N-1)$th, and Nth firing time, respectively. From the firing process, it is clear that: (1) At the first firing time, all the background neurons fire. (2) At the second firing time, the contour neurons are ignited. (3) Fired neurons will not fire again.

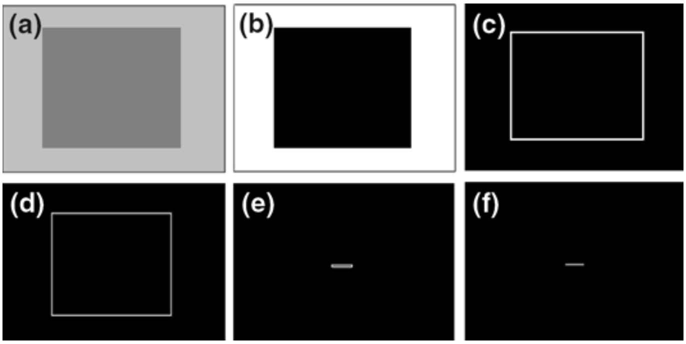

Figure 3. The autowaves travelling process of the PCNN for image thinning: (a) The original image. (b) Autowaves at the first firing time. (c) Autowaves at the second firing time. (d) Autowaves at the ith firing time, $1 < i < N$, where N is the total firing times. (e) Autowaves at the $(N-1)$th firing time. (f) Autowaves at the last firing time. (In these figures except the original image (a), light pixels indicate firing neurons.)

3. Image Thinning Using PCNN

A reliable and robust image thinning algorithm is necessary for many image processing systems (such as fingerprints recognition). The procedure of most existed thinning algorithms is [18]: (1) defining several masks, (2) according to the masks defined in the first step, repetitively removing the pixels of the edge, until only the central pixels remaining, which establish the skeleton of the image to be thinned. The famous Zhang and Suen Algorithm (ZSA) is of this type.

The proposed algorithm is different from these conventional algorithms. In this algorithm, thinning result is obtained when autowaves travelling in different directions meet. The proposed algorithm gives four autowaves meeting conditions to obtain the locus where the autowaves meet.

3.1. MEETING CONDITIONS

The meeting conditions are crucial for the proposed algorithm. The definition of meeting conditions is based on an assumption: the autowaves travelling in different directions meet each other either in the vertical direction or in the horizontal direction. The meeting conditions used in this paper are listed below:

In the horizontal direction, there are two meeting conditions (see Figure 4): "Condition H_1" and "Condition H_2". They are given as below:

Condition H_1 : At the previous time, neuron $N_{i,j}$ did not fire (see Figure 4(a)) and its neighbors: $N_{i,j-1}$ and $N_{i,j+1}$ fired. At the current time, the neuron $N_{i,j}$ fires (see Figure 4(b)) and its neighbors: $N_{i,j-1}$ and $N_{i,j+1}$ do not fire.

Condition H_2 : At the previous time, neurons: $N_{i,j}$ and $N_{i,j+1}$ didn't fire (see Figure 4(c)), and neurons: $N_{i,j-1}$ and $N_{i,j+2}$ fired. At the current time, the neurons: $N_{i,j}$ and $N_{i,j+1}$ fire (see Figure 4(d)), and the neurons $N_{i,j-1}$ and $N_{i,j+2}$ don't fire.

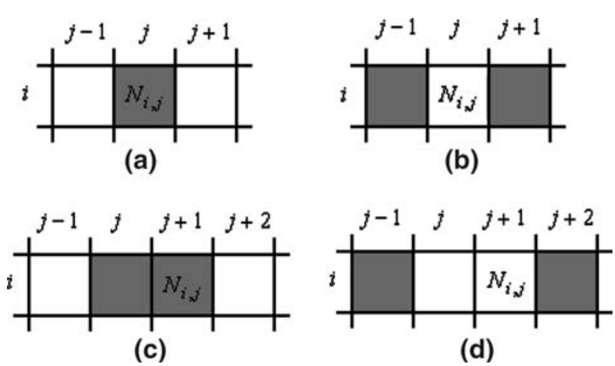

Figure 4. Meeting conditions in the horizontal direction. Condition H_1 :(a) Firing neurons at the previous time and (b) at the current time. Condition H_2 : (c) Firing neurons at the previous time and (d) at the current time.

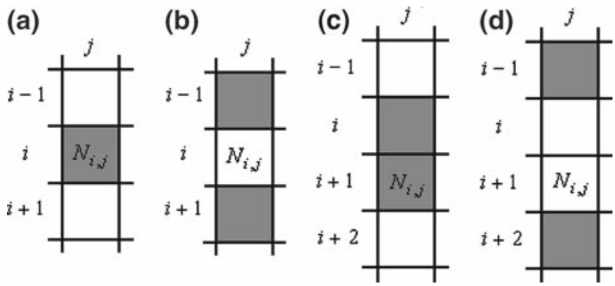

Figure 5. Meeting conditions in the vertical direction. Condition V_1 :(a) Firing neurons at the previous time and (b) at the current time. Condition V_2 : (c) Firing neurons at the previous time and (d) at the current time.

On the other hand, in the vertical direction, there are another two meeting conditions (see Figure 5): "Condition V_1" and "Condition V_2." They are given as below:

Condition V_1 : At the previous time, neuron $N_{i,j}$ (see Figure 5(a)) didn't fire and its neighbors: $N_{i-1,j}$ and $N_{i+1,j}$ fired. At the current time, the neuron $N_{i,j}$ fires (see Figure 5(b)) and its neighbors: $N_{i-1,j}$ and $N_{i+1,j}$ do not fire.

Condition V_2 : At the previous time, neurons: $N_{i,j}$ and $N_{i+1,j}$ didn't fire (see Figure 5(c)), and neurons: $N_{i-1,j}$ and $N_{i+2,j}$ fired. At the current time, the neurons: $N_{i,j}$ and $N_{i+1,j}$ fire (see Figure 5(d)), and the neurons $N_{i-1,j}$ and $N_{i+2,j}$ don't fire.

If a neuron $N_{i,j}$ satisfies one of the four meeting conditions, the pixel corresponding to this neuron belongs to thinning result. In the follows, the thinning algorithm is given.

3.2. THE PROPOSED THINNING ALGORITHM

First some symbols are given. F is the feeding input matrix, each element of which is either one or two. L is the linking input matrix. The value of L is given according to

$$L = Y \otimes K, Y_{i,j} = Y_{i-1,j} + Y_{i,j-1} + Y_{i,j+1} + Y_{i+1,j}, \tag{7}$$

where '$\otimes$' indicates two-dimensional convolution, Y is the binary output matrix, $Y_{i,j}$ is the (i, j)th entry of Y and K is a 3×3 kernel matrix.

$$K = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

U is the internal activity matrix,

$$U = F.*(1 + \beta L), \qquad U_{i,j} = F_{i,j}(1 + \beta L_{i,j}), \tag{8}$$

where '.*' indicates array multiplication, $U_{i,j}$ is the (i, j)th entry of U, and β is the linking strength. T^h is the threshold matrix is initialized by T_A^h and updated by

$$T^h \leftarrow T^h + Y.*(T_B^h - T_A^h),\ T_{i,j}^h \leftarrow T^h + Y_{i,j}(T_B^h - T_A^h), \tag{9}$$

here T_A^h and T_B^h are two const values, and the specification of them has been given in Section 3.

Output matrix Y is computed according to

$$Y = \text{Step}(U - T^h), \qquad Y_{i,j} = \begin{cases} 1, & U_{i,j} > T_{i,j}^h, \\ 0, & \text{otherwise}, \end{cases} \tag{10}$$

here Step is a step function giving $Y_{i,j} = 1$ where $U_{i,j} - T_{i,j}^h > 0$, and $Y_{i,j} = 0$ elsewhere.

T_r is a binary matrix recording the thinning results. F_{ired} is also a binary matrix recoding the fired neurons before the current firing time.

The proposed binary image thinning algorithm is listed below.

(1) $L = U = T_r = F_{\text{ired}} = 0$, $Y = Y_{\text{pre}} = 0$, and $T^h = T_A^h$. Parameters T_A^h, T_B^h and β are initialized according to the method presented in Section 3.

(2) If all the elements of the matrix F_{ired} are one, that is all neurons have fired, thinning process is accomplished and T_r is the thinning result, else go to the next step.

(3) $L = Y \otimes K$,
$U = F.*(1 + \beta L)$,
$Y_{\text{pre}} = Y$,
$Y = \text{Step}(U - T^h)$,
$F_{\text{ired}} \leftarrow F_{\text{ired}} + Y$,
$T^h \leftarrow T^h + Y.*(T_B^h - T_A^h)$.

(4) According to the meeting conditions, together with matrix Y_{pre} and Y, the neurons which satisfy one of the four meeting conditions are obtained. Then the pixels corresponding to these neurons are saved to T_r.

(5) Go to Step (2).

The proposed thinning algorithm by using autowaves generated by PCNN, combined with the four meeting conditions, can accomplish binary images thinning. In the follows, a thinning example is given to show the thinning process of our algorithm.

3.3. A THINNING EXAMPLE USING THE PROPOSED ALGORITHM

In this section an example (see Figure 6) is given to illustrate the thinning process of the proposed algorithm.

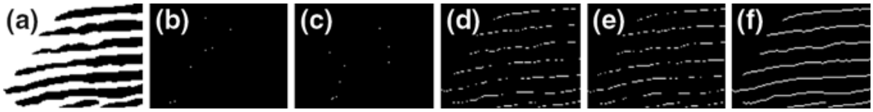

Figure 6. A thing example using the proposed algorithm. (a) The original image, (b) all neurons meeting the condition H_1, (c) H_2, (d) V_1, and (e) V_2. (f) Neurons meeting one of the four conditions.

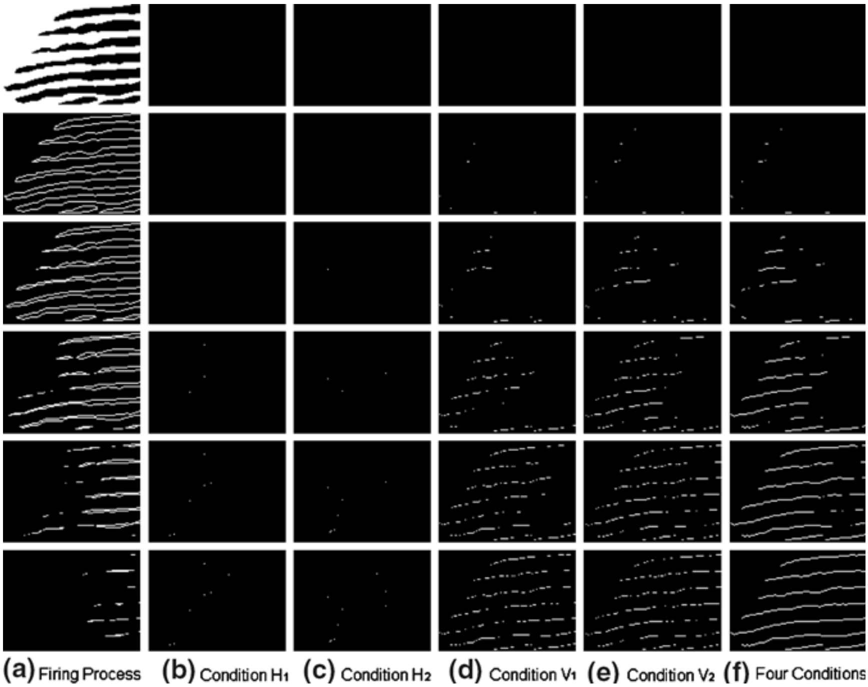

Figure 7. Thing process using the proposed algorithm.

Figure 6(a) is the original image to be thinned, Figure 6(b), (c), (d), and (e) show the neurons meeting the condition H_1, H_2, V_1, and V_2, respectively. Figure 6(f) show all the neurons meeting one of the four conditions, and which is the final thinning result. Obviously, Figure 6(f) can be obtained by fusing Figure 6(b), (c), (d), and (e).

Figure 7 shows the thinning process of the original image: Figure 6(a). The first column of Figure 7 shows the autowaves travelling process of the PCNN with Figure 6(a) as its input. The last colum of Figure 7 shows the thinning results using the four conditions at each firing time. The second, third, fourth, and fifth colums show the the thinning results using condition H_1, H_2, V_1, and V_2 at each firing time, respectively. This example illustrate the proposed algorithm is promising in binary image thinning.

4. Simulations

In this section, we focus on the verification of the effectiveness of the proposed thinning algorithm by experiments. Experiments on both artificial and realistic images have been done. These images include alphabet letters, Chinese characters, fingerprints, etc. Some examples will be presented below. All programs run on the same Intel P4 2.66 GHz personal computer.

4.1. SOME THINNING EXAMPLES

The first example is shown in Figure 8. The original image Figure 8(a) consists of two Chinese characters "tao tie." By using the algorithm of this paper, the skeleton is extracted, which is shown in Figure 8(b). The Figure 8(c) is obtained from the famous ZSA. The second example is shown in Figure 9. The original image, numbers "3, 4, 6", is shown in Figure 9(a). The thinning results obtained from our algorithm and the ZSA are shown in Figure 9(b) and (c), respectively. The third example is shown in Figure 10. The original image, alphabet letters "a, b, c", is shown in Figure 10(a). The thinning results obtained from our algorithm and the ZSA are shown in Figure 10(b) and (c), respectively. The simulation of fingerprint thinning is shown in Figure 11. The original fingerprint image and its thinning

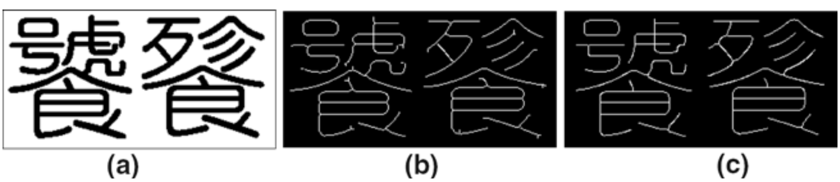

Figure 8. Thinning results of Chinese characters: (a) The original image. (b) The thinning results by the proposed algorithm. (c) The thinning results by the ZSA.

Figure 9. Thinning results of numbers: (a) The original image. (b) The thinning results by the proposed algorithm. (c) The thinning results by the ZSA.

Figure 10. Thinning results of alphabet letters: (a) The original image. (b) The thinning results by the proposed algorithm. (c) The thinning results by the ZSA.

Figure 11. Thinning results of fingerprint: (a) The original image. (b) The thinning results by the proposed algorithm. (c) The thinning results by the ZSA.

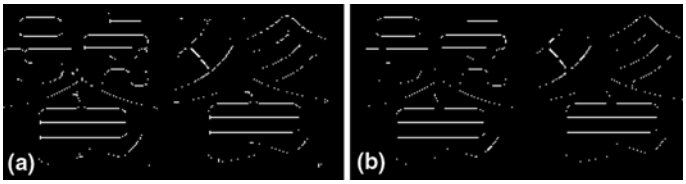

Figure 12. The "MAPs" of Figure 8(a) by using (a) our algorithm and (b) ZSA, respectively. (Light pixels indicate "MAPs.")

result by our algorithm are shown in Figure 11(a) and 11(b), respectively. Figure 11(c) is the thinning result obtained by ZSA.

Obviously, these thinning samples illustrate the proposed algorithm yields good thinning results.

4.2. EVALUATION

Finally, the performance of the proposed algorithm is evaluated. Commonly, the skeleton of an image is defined via the MAT. The MAT of an image is defined as follows. Given an image I with boder B. For each pixel p in I, finding its closest neighbor in B, if p has more than one such neighbor, it belongs to the "medial axis" of R. An essential property of skeletons is that they should include all or part of such pixels as p. In practice, only some pixels of thinning result belong to "medial axis." For simplicity, we call the pixels like p as "MAPs." In this paper, the rate of "MAPs" in skeleton pixels (all the pixels belonging to final thinning result) is used to evaluate the performance of a thinning algorithm. For simplicity, we call the rate as "R_{MSkel}."

Figure 12(a) and (b) show the "MAPs" of Figure 8(a) by using our algorithm and ZSA, respectively. Obviously, Figure 12(a) contains more "MAPs" than the Figure 12(b) does. That is, there are more "MAPs" in Figure 8(b) than in Figure 8(c).

Similarly, Figures 13(a), 14(a), and 15(a) show the "MAPs" of Figure 9(a), 10(a), and 11(a) by using our algorithm, respectively. Figure 13(b), 14(b), and 15(b) show the "MAPs" of Figure 9(a), 10(a), and 11(a) by using ZSA, respectively.

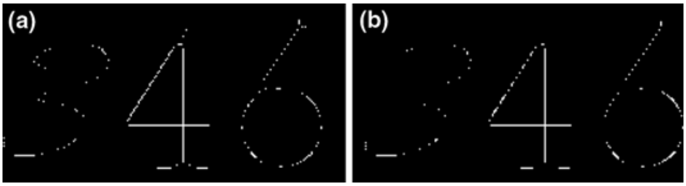

Figure 13. The “MAPs” of Figure 9(a) by using (a) our algorithm and (b) ZSA, respectively. (Light pixels indicate “MAPs.”)

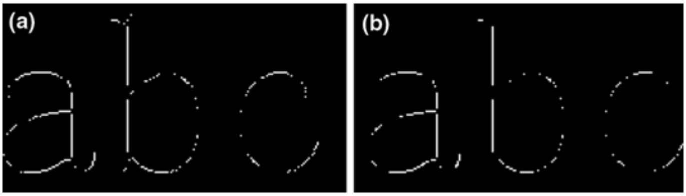

Figure 14. The “MAPs” of Figure 10(a) by using (a) our algorithm and (b) ZSA, respectively. (Light pixels indicate “MAPs.”)

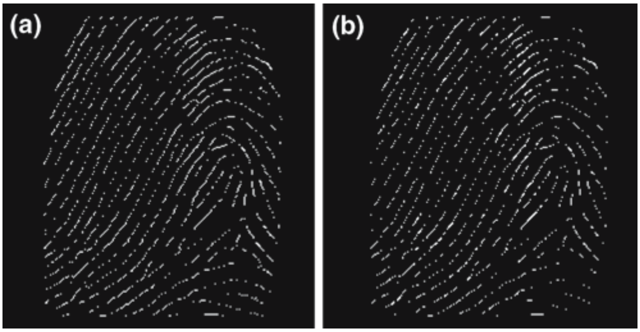

Figure 15. The “MAPs” of Figure 11(a) by using (a) our algorithm and (b) ZSA, respectively. (Light pixels indicate “MAPs.”)

Table 1. Comparision of the proposed algorithm and Zhang–Suen algorithm.

		The proposed algorithm				Zhang–Suen algorithm			
Figures' name	Figures' size	Time(s)	Amount of MAPs	Amount of skeleton pixels	R_{MSkel}	Time(s)	Amount of MAPs	Amount of skeleton pixels	R_{MSkel}
Figure 8(a)	279×143	0.591	983	1887	0.521	1.221	862	1900	0.454
Figure 9(a)	226×115	0.520	292	624	0.468	1.022	271	651	0.416
Figure 10(a)	208×110	0.351	311	535	0.581	0.591	263	558	0.471
Figure 11(a)	256×256	0.911	1822	4256	0.428	1.663	1581	4988	0.317

The detailed comparison of our algorithm and ZSA is listed in Table 1. Our algorithm is costs less time than ZSA does. The "R_{MSkel}" of our algorithm is higher than that of ZSA.

5. Conclusions

Based on the intuitive definition of MAT, a fast parallel binary image thinning algorithm is proposed by using autowaves generated by PCNN. We use the autowaves to mimic the grassfire, and give four meeting conditions to obtain the locus where autowaves meet. Determination of PCNN's parameters is also given. Experimental results confirmed the proposed algorithm is efficient in thinning digital images. Furthermore, we give a rate called "R_{MSkel}" to evaluate the performance of different thinning algorithms.

Acknowledgments

We are grateful to the reviewers whose comments have been most useful and encouraging.

References

1. Altuwaijri, M. and Bayoumi, M.: A new thinning algorithm for Arabic characters using self-organizing neural network, In: *IEEE International Symposium on Circuits and Systems*, pp. 1824–1827, Washington, USA, 1995.
2. Berthe, K. and Yang, Y.: Automatic edge and target extraction base on pulse-couple neuron networks wavelet theory, In: *The International Conferences on Info-tech and Info-net*, pp. 504–509, Beijing, China, 2001.
3. Caulfield, H. J. and Kinser, J. M.: Finding the shortest path in the shortest time using PCNN's, *IEEE Transactions on Neural Networks* **10**(3) (1999), 604–606.
4. Eckhorn, R., Frien, A. and Bauer, R.: High frequency (60-90 Hz) oscillations in primary visual cortex of awake monkey, *Neuroreport* **4**(3) (1993), 243–246.
5. Eckhorn, R., Reitboeck, H. J. and Arndt, M.: Feature linking via synchronization among distributed assemblies: Simulation of results from cat cortex, *Neural Computing* **2**(3) (1990), 293–307.
6. Gu, X. D., Yu, D. H. and Zhang, L. M.: Image thinning using pulse coupled neural network, *Pattern Recognition Letters* **25**(9) (2004), 1075–1084.
7. Hall, R. W.: Fast parallel thinning algorithms: parallel speed and connectivity preservation, *Communications ACM* **32**(1) (1989), 124–131.
8. Holt, C. M., Stewart, A., Clint, M., et al.: An improved parallel thinning algorithm, *Communications ACM* **30**(2) (1987), 156–160.
9. Ji, X. H. and Feng, J. F.: A new approach to thinning based on time-reversed heat conduction model, In: *The International Conference on Image Processing*, pp. 653–656, Singapore, 2004.
10. Johnson, J. L. and Padgett, M. L.: PCNN models and applications, *IEEE Transactions on Neural Networks* **10**(3) (1999), 480–498.
11. Johnson, J. L. and Ritter, D.: Observation of periodic waves in a pulse-coupled neural network, *Optics Letters* **18**(15) (1993), 1253–1255.

12. Kuntimad, G. and Ranganath, H. S.: Perfect image segmentation using pulse coupled neural networks, *IEEE Transactions on Neural Networks* **10**(3) (1999), 591–598.
13. Kinser, J. M.: Foveation by a pulse-coupled neural network, *IEEE Transactions on Neural Networks* **10**(3) (1999), 621–625.
14. Krishnapuram, R. and Chen, L. F.: Implementation of parallel thinning algorithms using recurrent neural networks, *IEEE Transactions on Neural Networks* **4**(1) (1993), 142–147.
15. Matsumoto, T. and Chua, L. O., Yokohama T.: Image thinning with a cellular neural network, *IEEE Transactions on Circuits and Systems* **37**(5) (1990), 638–640.
16. Qu, H. and Zhang, Y.: A new algorithm for finding the shortest paths using PCNNs, *Chaos, Solitons and Fractals*, Accepted.
17. Rafael, C. G. and Richard, E. W.: *Digital Image Processing.* Publishing House of Electronic Industry, China, 2004.
18. Tang, Y. Y. and You, X. G.: Skeletonization of Ribbon-like shapes based on a new wavelet function, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(9) (2003), 1118–1133.
19. Yu, B. and Zhang, L. M.: Pulse-coupled neural networks for contour and motion matchings, *IEEE Transactions on Neural Networks* **15**(5) (2004), 1186–1201.
20. Zhang, T. Y. and Suen, C. Y.: A fast parallel algorithm for thinning ditigital patterns, *Communications ACM* **27**(3) (1984), 236–239.