A new thinning algorithm for binary images

Student Name：湯琦恩

Student ID：109323075

Date：112 年 01 月 09 日

Location：暨南大學

Abstract

Read paper: "L. Ben Boudaoud, A. Sider and A. Tari, "A new thinning algorithm for binary images," 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT), 2015, pp. 1-6, doi: 10.1109/CEIT.2015.7233099.
https://ieeexplore.ieee.org/document/7233099/references#references

Skeletonization plays a main role in image recognition and classification. The ZS algorithm proposed by Zheng and Suen (T. Y. Zhang, C. Y. Suen, 1984) is one of the popular and well-proven algorithms in skeletonization. The authors of this paper wanted to produce a faster and more efficient thinning algorithm with as little distortion as possible. The concept is to combine the ZS algorithm's directional approach and subfield approach (Guo, Zicheng and Hall, Richard W., 1989) to generate a new thinning algorithm, which is improved compared to the disadvantage of ZS algorithm: First, it cannot be used for 2× 2, Second, the generated skeleton still has redundant pixels, and third, the diagonal skeleton cannot be detected..

Keywords: Thinning algorithm; Binary images; Skeleton; ZS algorithm;

I. INTRODUCTION

A. The directional approach

For example, the ZS algorithm judges and deletes pixels based on directions or combinations of directions (north, west, east, and south). If they belong to the same direction, they will be deleted in parallel.

B. The subfield approach

As proposed by Guo, Zicheng and Hall, Richard W., 1989, the graph is decomposed into sub-domains, not judging the edge direction, but according to some criteria, such as the parity of pixels, so that sub-domains with the same attribute are deleted in parallel.

II. SKELETONIZATION ALGORITHM

A. ZS algorithm

It is a directional thinning approach, which deletes the southeast boundary pixel and northwest corner pixel, and deletes the northwest boundary pixel and south corner pixel respectively.

P9	P2	P3
P8	P1	P4
P7	P6	P5

Fig. 1 ZS algorithm using 8-connectivity ,working on 3 ×3 neighborhood.

1. Algorithm description:

   For the first iteration , if the edge of point P_1 satisfies the following conditions, it will be deleted P_1：

   $(1) 2 \leq B(P_1) \leq 6$

   $(2) A(P_1) = 1$

   $(3) P_2 \times P_4 \times P_6 = 0$

   $(4) P_4 \times P_6 \times P_8 = 0$

   $A(P_1)$ When traversing the 8 neighbors of P_1， the number of values changing from 0 to 1 (white to black)；$B(P_1) = \sum_{i=2}^{9} P_i$ is the number of 1 in P_1 、P_2 、P_3 ... P_8.

   For the second iteration, if point P_1 satisfies the following conditions, will deleted P_1：

   $(1) 2 \leq B(P_1) \leq 6$

   $(2) A(P_1) = 1$

   $(3) P_2 \times P_4 \times P_8 = 0$

   $(4) P_2 \times P_6 \times P_8 = 0$

2. Disadvantage：

   (1) The ZS algorithm cannot be applied to the 2×2 square, because the 8-connected pixels of a certain pixel cannot be found.

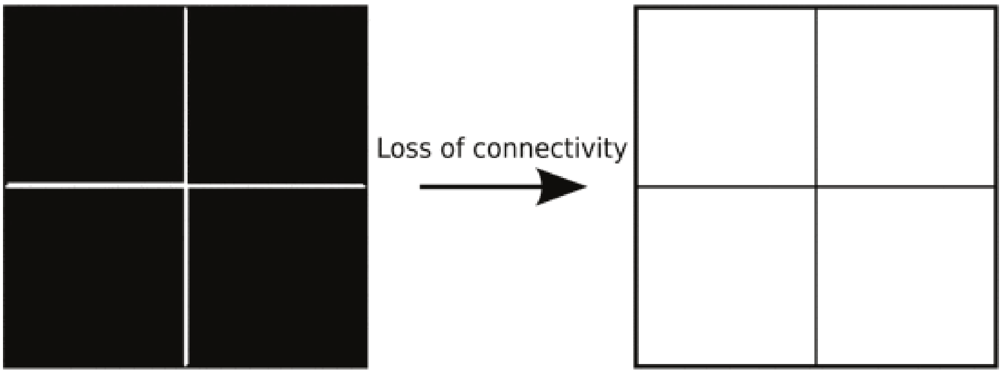


Fig. 2 ZS thinning result of a 2×2 square.

(2) There are redundant pixels in the generated skeleton, even if it is deleted, it will not affect the continuity of the skeleton.

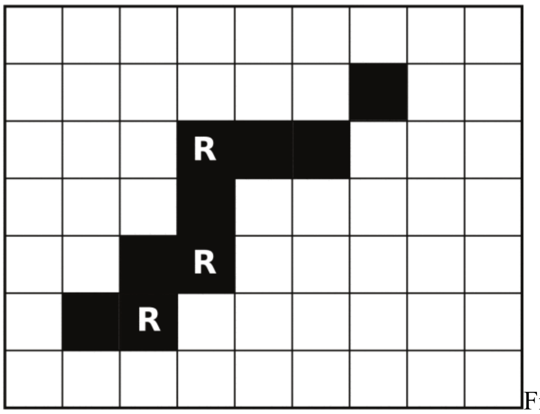


Fig. 3 The place marked r is the redundant pixel in the skeleton

(3) For diagonal lines, pixels will be deleted too much so that the skeleton of the diagonal line cannot be displayed.

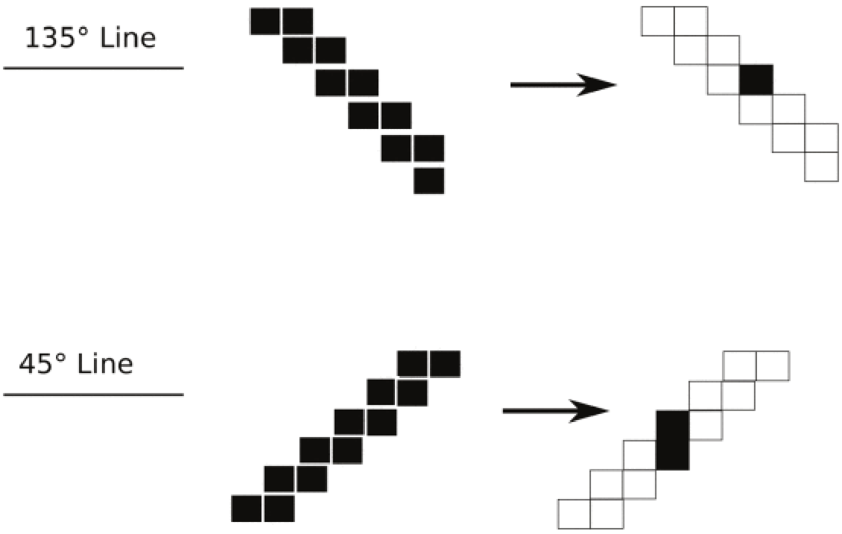


Fig. 4 The ZS algorithm will over-erode the diagonal

B. The new thinning algorithm used in this paper

Combining the ZS algorithm and the subfield approach, the picture is divided into two areas. For each pixel point $P(i,j)$, it can be divided into odd pixels(ie $(i+j)mod2=0$) and even pixels(ie $(i+j)mod2 \neq 0$).

1. Algorithm description：

   (1)Use the same 8-connected neighborhood as the ZS algorithm.

   (2)P_1 is the pixel to be deleted.

   (3)$B(P_1)=\sum_{i=2}^{9} P_i$ is the number of 1 (black grid) in P_1 、P_2 、$P_3 \dots P_8$.

   (4)$C(P_1)=\neg P_2 \wedge (P_3 \vee P_4)+\neg P_4 \wedge (P_5 \vee P_6)+\neg P_6 \wedge (P_7 \vee P_8)+\neg P_8 \wedge (P_9 \vee P_2)$ is the "1" in the 8-connected neighborhood of the number of connected 8-components.

2. For the first iteration, the conditions to be met by P_1 (computing even subfields)：

   (1)$(i+j)mod2=0$

   (2)$C(p)=1$

   (3)$2 \leq B(P_1) \leq 7$

   (4)$P_2 \times P_4 \times P_6 = 0$

   (5)$P_4 \times P_6 \times P_8 = 0$

3. For the second iteration, the conditions to be met by P_1 (compute odd subfields)：

   (1)$(i+j)mod2 \neq 0$

   (2)$C(p)=1$

   (3)$2 \leq B(P_1) \leq 7$

   (3)$P_2 \times P_4 \times P_8 = 0$

   (4)$P_2 \times P_6 \times P_8 = 0$

III. SIMULATION RESULTS

A. Performance Measures：

1. Thinning rate (TR)：The degree of thinness of the image. Where TR=1 indicate the image is perfectly thinned, TR=0 means that the image is not thinned at all.

$$TR = 1 - TM1 \div TM2$$

   TM1：the number of triangles whose three vertices are all black pixels：

$$TM1 = \sum_{i=0}^{n} \sum_{j=0}^{m} triangle\_count(P[i][j])$$

$$TC(p) = (p8 * p9) + (p9 * p2) + (p2 * p3) + (p3 * p4)$$

$TM2$：The largest triangles consist of black pixels for an image can have.

$$TM2 = 4 \times [max(m,\ n) - 1]^2$$

2. Execution Time (ET)：The real time cost ,the execution time of thinning process in second.

3. Sensitivity Measurement (SM)：Measure the noise immunity performance of the contour bobbin. Noise refers to the perturbations of the outline of the object, for example: the edge of the handwritten font, rather than the redundant pixels in the image. The redundant pixels can be processed by other image preprocessing techniques, such as thresholding, cutting, etc. The impact of contour perturbation is mainly discussed here, because this can cause problems such as final skeleton deformation , disconnection and offshoots. Here we let SM be：

$$SM = \sum_{i=0}^{n} \sum_{j=0}^{m} s(P[i][j])$$

$$s(P[i][j]) = \begin{cases} 1 & , if\ A(P[i][j]) > 2 \\ 0 & , othter \end{cases}$$

For an image, the offshoots point should be fixed. If the output skeleton has too many offshoots points, it means that the Thinning algorithm is susceptible to object contour disturbance. That is to say, the Thinning algorithm is overly sensitive

4. Connectivity Measurement (CM)：Measuring the connectivity of the output skeleton computes disconnected lines or discrete points that should not appear, which makes what would otherwise be the same object be considered different objects. If there are disconnected lines, there are endpoints and discrete points. The endpoints and discrete points of an image are fixed . If the output skeleton endpoints and discrete points are more than the original image, it means that the Thinning algorithm is over-eroded. Here we let CM be：

$$CM = \sum_{i=0}^{n} \sum_{j=0}^{m} s(P[i][j])$$

$$s(P[i][j]) = \begin{cases} 1 & , if\ B(P[i][j]) < 2 \\ 0 & , othter \end{cases}$$

The author original used the Thinning rate (TR) and Thinning Speed (TS), measures the number of pixels thinned per time unit (second)

$$TS = \frac{ET}{DP}$$

$$DP = OP - SP$$

Where: DP (deleted points) is the number of black points deleted during thinning, OP (object points) is the number of black points of the image before thinning, SP (skeletal points) is the number of black points remaining after thinning or the number of pixels in the skeleton.

Because the purpose of the author's improvement of a new skeletonization algorithm is to improve the accuracy and efficiency of thinning, Here I use in the implementation 1. Thinning rate , to calculate the performance after thinning, that is, how much thinning, that is, how much skeleton Coarse, it is best to hope that there is only one pixel left after thinning, so the closer to the value 1 of TR is better. 2. Execution Time (ET) (the time required for skeletonization.) 3. Sensitivity Measurement (SM), to calculate the sensitivity of Thinning algorithm to noise, the value listed here is the refined $SM_{\text{after Thinning}} - SM_{Original\ Inage}$, if the value is larger, it means that the Thinning algorithm is more sensitive to noise. 4. Connectivity Measurement (CM), calculate the connectivity of the thinned skeleton, whether there are breakpoints that should not be there, the value listed here is $CM_{\text{after Thinning}} - CM_{Original\ Inage}$, if the value is larger, it means The thinning algorithm erodes too much, disconnecting what should be connected. The Table of Performance Measures is in page 12.

B. Result

1. 2×2 square and 135° line and a 45° line

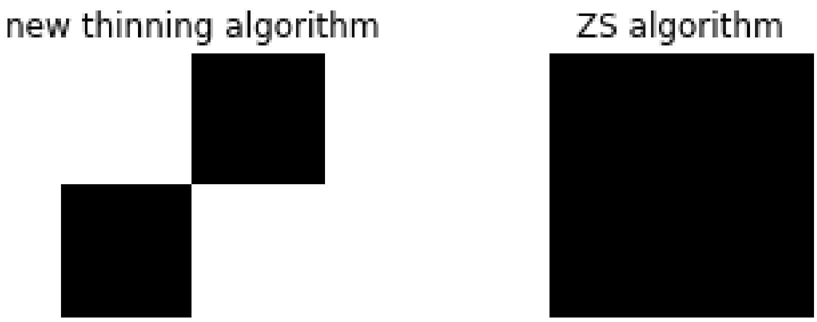


Fig. 5 Thinning result of 2×2 square by ZS algorithm and new thinning algorithm.

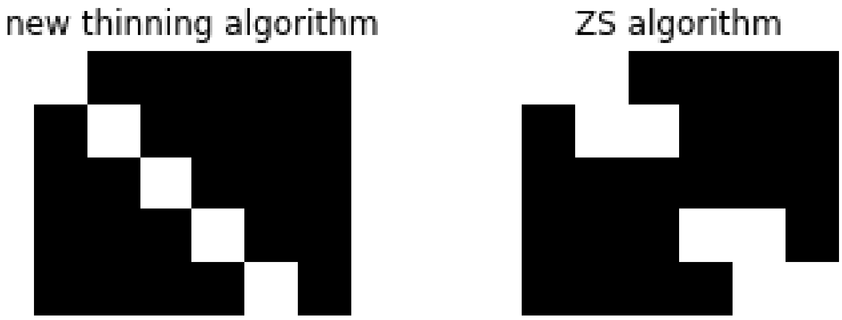


Fig. 6 Thinning result of 135° line by ZS algorithm and new thinning algorithm

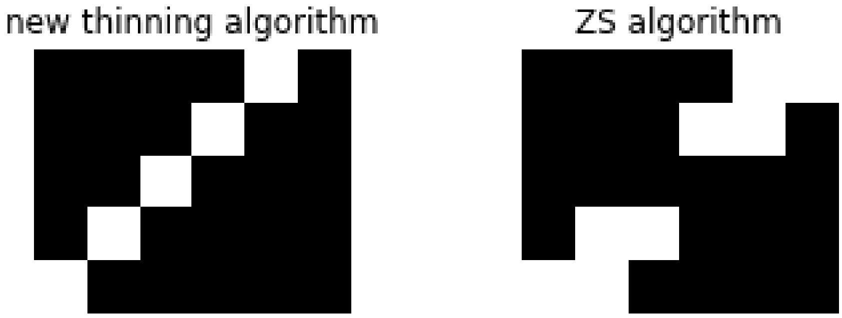
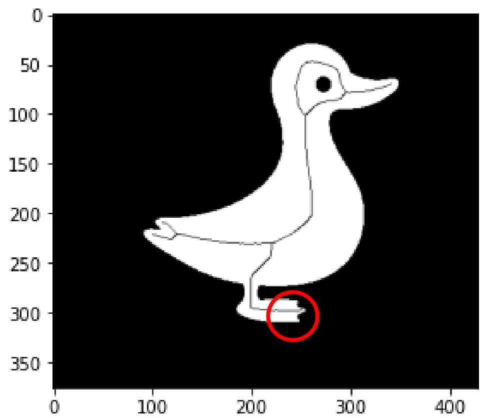


Fig. 7 Thinning result of 45° line by ZS algorithm and new thinning algorithm

2. Duck、Horse



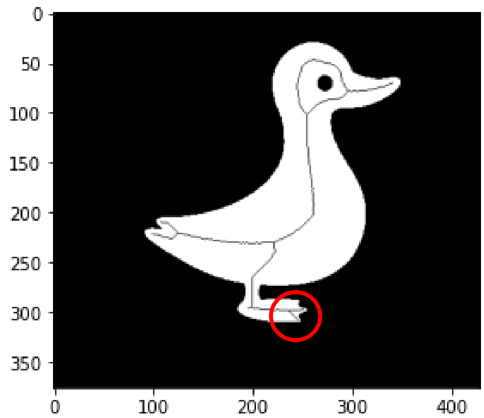
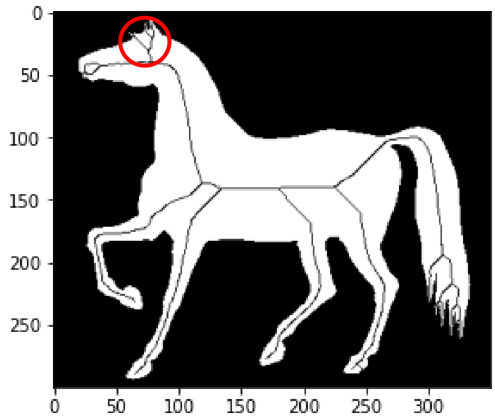


Fig. 8 Thinning result of duck by the ZS algorithm. Fig. 9 By the new thinning algorithm.



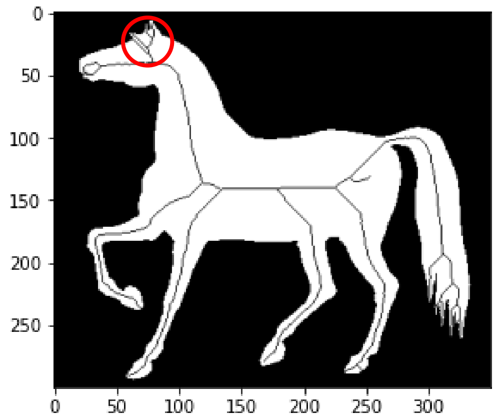


Fig. 10 Thinning result of horse by the ZS algorithm. Fig. 11 By the new thinning algorithm.

3. Handwriting arabic numbers English

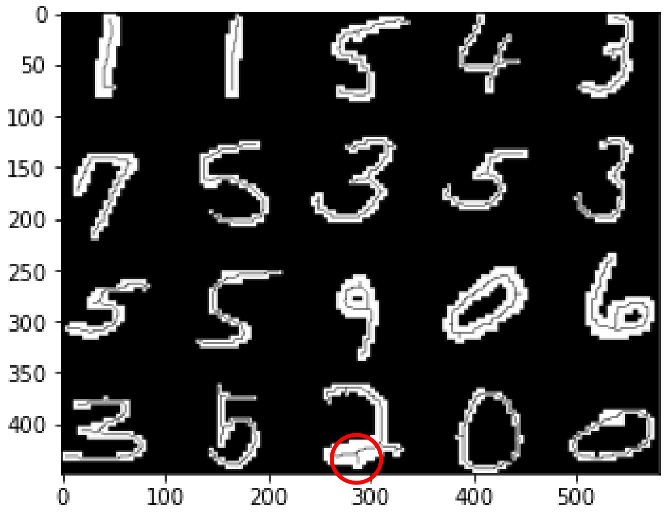


Fig. 12 Thinning result of arabic numbers by the ZS algorithm.

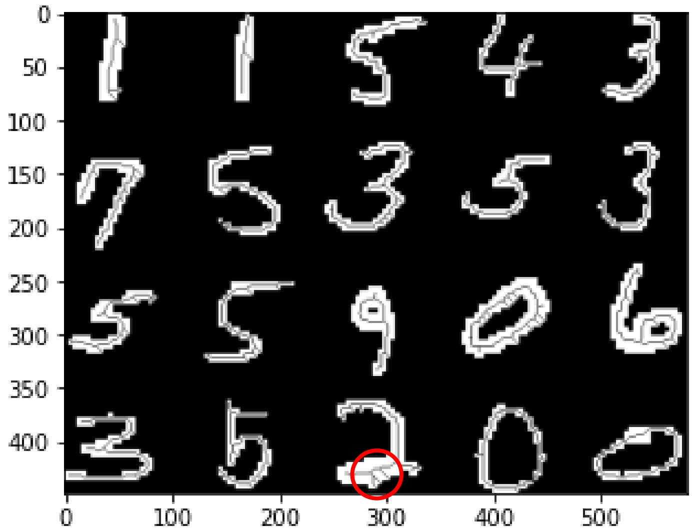


Fig. 13 Thinning result of arabic numbers by the new thinning algorithm

Fig. 14 Thinning result of English handwriting by the ZS algorithm.

Fig. 15 Thinning result of English handwriting by the new thinning algorithm.

4. Chinese

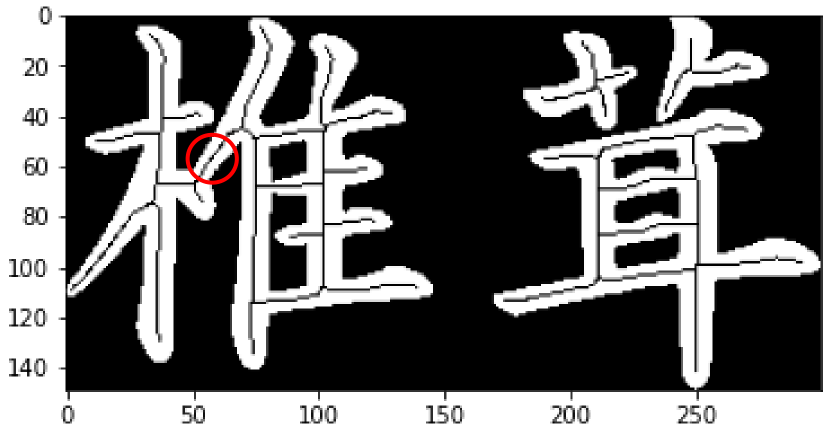


Fig. 16 Thinning result of Chinese by the ZS algorithm.

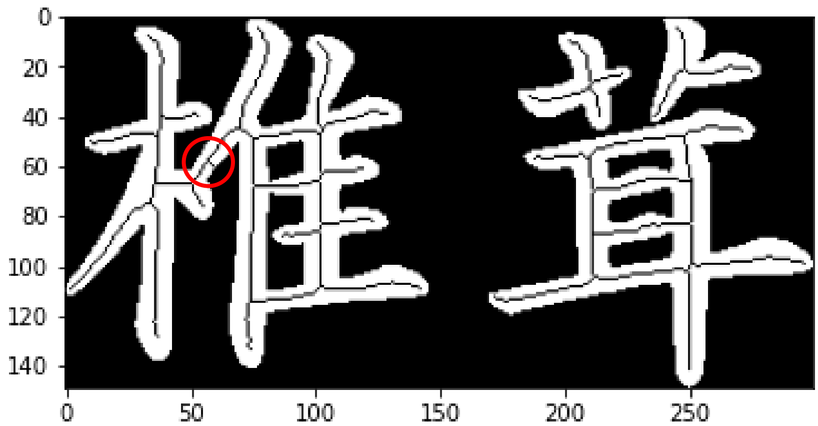
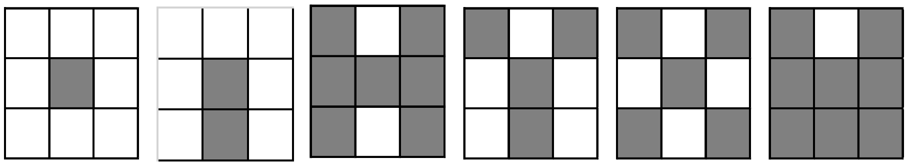


Fig. 17 Thinning result of Chinese by the new thinning algorithm.

IV. DISCUSION

1. ZS algorithm

(1) For the first condition $2 \leq B(P_1) \leq 6$, the aim is to preserve the endpoints. As in Fig.18, when $B(P_1) = 0$, it means that P_1 is an isolated point; when $B(P_1) = 1$, P_1 is an end is an end point, and the isolated point or end point is not deleted, otherwise the object will be thinned and disappear or cause line degradation. When$B(P_1) > 6$, then P_1 is an interior point and should not be deleted, otherwise there will be a hole in the thinning result.



(a) B=0 (b) B=1 (c) B=2 (d) B=3 (e) B=4 (f) B=7

Fig. 18

(2) The second condition $A(P_1) = 1$, which can prevent the deletion of points located at the end points of the skeleton, $A(P_1) > 1$, then P_1 must be one of the "bridges" of certain parts (components) in the object, and if it is deleted, it will cause a breakpoint. as follows：

0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0

Fig. 19 For the center point of the orange box P_1 ，$A(P_1) = 2$ 。

As in Fig 19., if the center point of the orange box is deleted, the line will be broken.

(3) The first iteration (3) $P_2{\times}P_4{\times}P_6 = 0$ (4)$P_4{\times}P_6{\times}P_8 = 0$ condition, the condition to be established bellow these conditions, P_4 or $P_6 = 0$ or P_2 and P_8 are zero, as follows：

p9	p2	p3
p8	p1	p4
p7	p6	p5

Fig. 20 when P_2 andP_8 are all zero.

The removed P_1 point may be the southeast boundary point or the northwest boundary point.

(4) The second iteration (3) $P_2{\times}P_4{\times}P_8 = 0$ (4)$P_2{\times}P_6{\times}P_8 = 0$ condition ,the condition to be established bellow these conditions, P_2 or $P_8 = 0$ or P_4 and P_6 are all zero, as follows：

p9	p2	p3
p8	p1	p4
p7	p6	p5

Fig. 21 when P_4 andP_6 are all zero

The removed P_1 point may be south boundary point or east boundary point or northwest corner point.

2. the new thinning algorithm

(1) Regarding the three disadvantage of the ZS algorithm, we can see that the new thinning algorithm can solve them all, because the author divide the original image into odd pixel to delete south boundary point or east boundary point and even pixel to delete the southeast boundary point or the northwest boundary point, so the skeleton can be extracted more finely.

(2) The two conditions of $P_2 \times P_4 \times P_8 = 0$ and $P_2 \times P_6 \times P_8 = 0$ in the original ZS algorithm are retained, and the $C(p)$variable is added, p is defined as the number of distinct eight-connected components of ones in p's eight-neighborhood. If $C(p) = 1$ implies p is eight-simple when p is a boundary pixel , delete p does not disconnect 3×3 neighborhoods.

(3) $2 \leq B(P_1) \leq 7$ The set of boundary points represented by inspection is larger than the set of boundary points considered by ZS, which means more boundary pixels are removed, some inner points are also removed.

3. Performance Measures

(1) Thinning Measure 1 (TM1)：If anywhere in the image is not one pixel wide (i.e. not thinned), there is a triangle of three black pixels. The thickness of the final skeleton can be measured by the number of triangles.

(2) Thinning Measure 2 (TM2)：Represents the triangle formed by the largest black pixels that an image can have.

0	0	0	0
1	1	1	1
1	1	1	1
0	0	0	0

(a)

1	0	0
0	1	0
0	1	0
0	1	0

(b)

1	1
1	1

(c)

Fig. 22 (a) This is a two-pixel wide horizontal line. TM1 = 12, TM2=4×[max(4,4)-1]^2 = 36, TM = 0.666667.The horizontal line be slightly thinned, though not completely.

(b) This is a curved line with TM1=0. Hence, TM=1. The line is completely thinned.

(c) This is a square consisting of 4 black pixels. TM1=4 and TM2=36, hence, TM = 0.The square is not thinned at all.

4. Result discusion

Table 1

Image	ZS algorithm				new thinning algorithm			
	TR	ET(s)	SM	CM	TR	ET(s)	SM	CM
Duck	0.9985342	34.235636	4	4	0.998889	18.152677	10	6
Horse	0.9951519	15.877044	14	8	0.996584	12.731954	20	9
arabic numbers	0.9952452	9.252501	61	32	0.996394	7.902656	136	178
English handwriting	0.9988424	9.284417	20	17	0.999222	7.433935	38	21
Chinese	0.9939821	1.901738	30	21	0.994686	1.473267	50	22

For Table 1, first look at the first column, you can see that the new thinning algorithm has a slightly higher TR in each image category than ZS algorithm, but both have reached 0.99 and are very close to 1, both of which are very good algorithms. For the second column, it can be seen that the new thinning algorithm is faster than the ZS algorithm in all kinds of image skeletonization, and even nearly twice as fast in the Duck image. The improved new thinning algorithm has greatly improved the thinning efficiency . For the third and fourth columns, because of the increase of TR (thinning rate), the offshoots points and discrete points of the new thinning algorithm are much higher than those of the ZS algorithm. For object contour disturbances, the new thinning algorithm is too sensitive and output more offshoots, and For noise immunity performance, the new thinning algorithm performs poorly.

So it can be seen that although the new thinning algorithm is faster and produces a better skeleton in terms of thinning rate, it also solves the problem of loss of connectivity due to the complete removal 2 × 2 square incurred in ZS and adequately avoids excessive erosion. However, there are also problems of over-sensitivity and poor noise immunity performance. I think this is why the author only mentioned in the conclusion that There are still some left two Performance Measures, CM and SM, can be used for comparison, but they are not included in the article.

5. Problems for simulation

(1) Pay attention to whether the black part is 1 or 0. The thinning algorithm used here requires the skeletonized object to be 1, and the output skeleton is white (0). Because the white of the general binary image is 255, and the black is 0, so it is decided to use the cv2.THRESH_BINARY_INV or cv2.THRESH_BINARY according to whether the background is white or black.

(2) Because there are many conditions, some need to take a lot of time to calculate separately, don't write all the conditions in the same if, it will run very, very slowly。

(3) Note that the conditions (3)(4) of the ZS algorithm and the conditions (4)(5) of the new thinning algorithm are different in the first iteration and the second iteration, One is to deleted the southeast boundary pixel and northwest corner pixel, and the other is to delete the northwest boundary pixel and south corner pixel respectively. If they are set to be the same, the output skeleton will be a straight line.

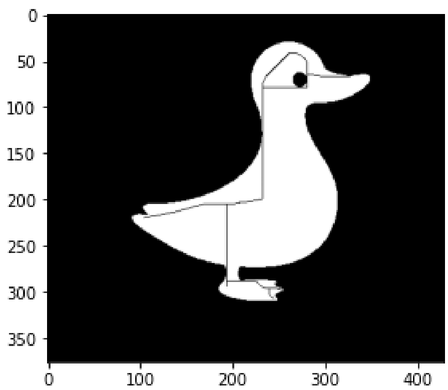


Fig. 23 Both the first and second iterations use $P_2 \times P_4 \times P_6 = 0$ and $P_4 \times P_6 \times P_8 = 0$

(4) If $A(P_1) = 1$ conditions of the ZS algorithm is missing, the algorithm will erode too many pixels and the generated skeleton will be very sparse.

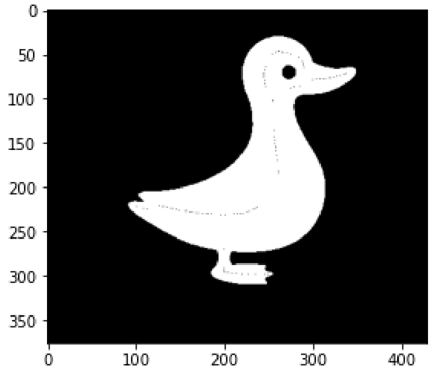


Fig. 24 with no $A(P_1) = 1$ condition.

(4) Every time the thinning algorithm is re-run, the Execution Time (ET) will be different. Sometimes there will be a difference of more than ten seconds from the last measurement result

V. THOUGHTS

在實作的過程中學到了很多關於 thinning algorithm 的原理，本來只是一個個條件公式，真正實作後更可以理解為甚麼要設這些條件，如果少了其中某項會發生甚麼事。還要去查詢有甚麼 Performance Measures，這些公式是依靠甚麼原理，是在計算骨架化的甚麼特性，再決定要使用何種 Performance Measures，並把它實作出來。最後成功只靠作者的敘述就自己實作出作者的結果覺得很有成就感。

VI. REFERENCES

[1] L. Ben Boudaoud, A. Sider and A. Tari, "A new thinning algorithm for binary images," 2015 3rd International Conference on Control, Engineering & Information Technology (CEIT), 2015, pp. 1-6, doi: 10.1109/CEIT.2015.7233099..

[2] Z. Guo and R. W. Hall, “Parallel thinning with two-subiteration algorithms,” Commun. ACM, vol. 32, no. 3, pp. 359–373, Mar. 1989.[Online]. Available: http://doi.acm.org/10.1145/62065.62074

[3] T. Y. Zhang and C. Y. Suen. 1984. “ A fast parallel algorithm for thinning digital patterns”. Commun. ACM 27, 3 (March 1984), 236–239. https://doi.org/10.1145/357994.358023

[4] Björn Þór Jónsson, Cathal Gurrin, Minh-Triet Tran, Duc-Tien Dang-Nguyen, Anita Min-Chun Hu, Binh Huynh Thi Thanh, Benoit Huet,2022, “ MultiMedia Modeling: 28th International Conference, MMM 2022, Phu Quoc, Vietnam, June 6–10, 2022, Proceedings, Part II” , Phu Quoc, Vietnam

[5] R. G.S. Ng and C. Quek, “A novel single pass thinning algorithm”, IEEE Transaction on System Man and Cybernetics, 1994.

[6] Tarabek, Peter. "Performance measurements of thinning algorithms." (2008).

[7] 对《A Fast Parallel Algorithm for Thinning Digital Patterns》一文的理解(上), Available: https://blog.csdn.net/keneyr/article/details/88944563(April 01,2019)