STUDY OF SOME THINNING ALGORITHMS AND THEIR ANALYSIS

Dissertation submitted in partial fulfilment of the requirements for the award of the Degree of Master of Technology in Computer Science

by

Saurav Bhattacharyya

INDIAN STATISTICAL INSTITUTE

203 B.T Road,Calcutta, 700035

CERTIFICATE

This is to certify that the work described in the dissertation entitled "Study of Some Thinning Algorithms and Their Analysis", has been undertaken by Saurav Bhattacharyya under my guidence and supervision. The dissertation is found worthy of acceptance for the award of the Degree of Master of Technology in Computer Science.

Dated August 31, 1992

Place : Calcutta

Srimanta Pal

(Dr. Srimanta Pal)

Indian Statistical Institute

203, B. T. Road, Calcutta

700 035

C O N T E N T S

ABSTRACT

Thinning is one of the important low level segmentation procedure. On the other hand numerous thinning algorithm have been proposed so far but the main bottleneck as given bellow :

a> Existing thinning algorithms are not generalised in the sense that a particular method may be suitable to a particular class of images but not to all kind of images

b> A generalised model for measuring time complexity of thinning algorithms is absent. As a result their performance can be compared in order to select with minimum time complexity with better skeletone.

c> Most of the thinging algorithms are iterative approximation method, when in most of the cases shape of the skeletone are not preserved by the approximation method. As a result inaccurate results are submitted to the next stage thus error propagation has occurred.One model of analysis of the most widely used template matching thinning algorithms, based on markov process method for conducting average care analysis of thinning algorithms in order to measure their performance have been proposed.

d> Also we have designed a model for the generation of random binary image which are either normally distributed or 4/8 connected uniformally distributed. We have used these binary images to study the emperical average performance of some template matching thinning algorithms.

CHAPTER I

SOME TEMPLATE MATCHING THINNING ALGORITHMS

Thinning is one of the most important operation being performed during low level segmentation. Thinning is perfomed on the edge-image which is obtained as the outcome of edge detection. An edge image is consisting of a set of edge points lying on the boundaries of the object present in the original image. Usually the boundary of the object are elongated due to the side effect of the edge detection process. Such elongated boundary of the object are thined to get skeletone of the object. In general thinning can be defined as the process of unusing points from elongated boundaries of the object until the boundaries are reduced to one-pixel wide boundaries called skeletone of the object.
Thinning plays a very important role in the preprocessing stage OfImage Processing. It deals with extracting the distinctive features known as skeletons from the images.

1.1. ALGORITHMS:

1.1.1 Zhang and Suen's thinning algorithm [Zhang. 84]

This algorithm is operated on a 3x3 window taking centre pixel as the candidate one.Also it is a two pass parallel template matching thinning algorithm whose description is given below.

P_9 (i-1,j-1)	P_2 (i-1,j)	P_3 (i-1,j+1)
P_8 (i,j-1)	P_1 (i,j)	P_4 (i-1,j+1)
P_7 (i+1,j-1)	P_6 (i+1,j)	P_5 (i+1,j+1)

fig(1.1) A 3x3 window

Algorithm

Pass-I

In parallel picture/image processing the new value given to a point at the nth iteration depends on its own value as well as those of its eight neighbour at the (n-1)th iteration so that all picture points can be processed simultaneouly. It is assume that 3x3 windows are used and each element is 8-connected. The window is scanning the image from left to right and top to bottom way .

Step Z1.1

a> A(P_1) = total number of '01' occurence in the window of figure 3.1 starting at P_2 following clockwise direction on the 8 neighbours of P_1.

b> B(P_1) = total number of '1' neghbour pixels if P_1.

Step Z1.2

An element is to be removed when it is on the edge of a stroke. This is often identified with the condition of having two to six connected neighbours, that is $2 \le B(P_1) \le 6 \Leftrightarrow P_1$ is on the edge. So we can formulate as follows if $2 \le B(P_1) \le 6$ and $A(P_1)=1$ and atleast one of the three comditions

(i)the east neighbour (i.e. P_4) is '0'.

(ii) the south neighbour (i.e. P_6) is '0'.

(iii) the north and west neighbour (i.e. P_2 and P_8) are simultaneouly '0' is true then the candidate pixel P_1 is to be deleted from the next pass II .

Step Z1.3

Scan the window by one position at a time or scan all the position simultaneouly and apply steps Z1.1 and Z1.2.

Step Z1.4

When all the pixels of the given image have been processed and at least one '1' pixel has become '0' then go to pass II else exit.

Pass II

Step Z2.1

Same as step Z1.1 in pass I.

Step Z2.2

If $2 \le B(P_1) \le 6$ and $A(P_1)=1$ and if atleast one of the conditions

(i) The north (i.e. P_2) is '0'

(ii) the west (i.e. P_8) is '0'

(iii) The east and south (i.e. P_4 and P_6) are simultaneously '0' is true then the candidate key P_1 is to be deleted for the next pass I .

Step Z2.3

Scan the window one position and apply step Z1.1 and step Z2.2

Step Z2.4

When all the pixel of the given images have been processed and atleast one pixel has become '0' then go to pass I else exit.

1.1.2 **Holt et al thinning algorithm** [Holt 87]

Holt's algorithm is the boolean representation of the modified Zhang and Suen's algorithm. It is given bellow.

In this algorithm they had assumed a 4x4 window (figure1.2.) is sliding over the binary image. It is an one pass parallel template matching thinning algorithm.

P_9	P_2	P_3	P_{13}
P_8	P_1	P_4	P_{14}
P_7	P_6	P_5	P_{15}
P_{10}	P_{11}	12	P_{16}

figure 1.2 a 4x4 window having P_1 as candidate pixel.

Definition 1.3 [Edge]

The edge of an pixel P_1 can be present provided a 3x3 window as in figure 1.1 satisfying all the conditions:

1> $P_1 = 1$

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

3> $A(P_1) = 1$

where $A(P_1)$ and $B(P_1)$ as defined in section 1.2.

Algorithm

In parallel picture processing the new value given to a point at the nth iteration depends on its own value as well as those of its eight neighbour at the (n-1)th iteration so that all picture points can be processed simultaneouly. It is assume that 4x4windows are used and each element is 8-connected. The window is scanning the image from left to right and top to bottom way .

If one of the four conditions, as shown below ,is satisfied then the pixel will not be deleted :

1> the candidate pixel will not be on the edge.

2> the value of north and south neighbors will be '1' and the east neighbor will be on edge.

3> the value of west and east neighbors will be '1' and the south neighbor will be on edge.

4> the east, south-east and south neighbours will be on edge. Repeat the process until no '1' to '0' transformation takes place.

The overall expression for servival is given bellow:

P_1 and (not (EDGE P_1) or
(EDGE P_4 and P_2 and P_6) or
(EDGE P_6 and P_8 and P_4) or
(EDGE P_4 and EDGE P_5 and EDGE P_6))

The EDGE function is defined as in [Holt 87].

1.1.3 Chin et al thinning algorithm [Chin 87]

In this algorithm they had assumed a window(figure1.3.)is sliding over the binary image. It is an one pass parallel template matching thinning algorithm.

Eight of them [figure 1.3(a)-(h)] are used for detecting bondary pixel,where as other two [figure 1.3(i)-(j)] are used for disabling the deletion of pixels if certain conditions are satisfied and the last four templates [figure 1.3(k)-(n)] are for trimmining.

0	0	0
1	1	1
x	1	x

(a)

0	1	x
0	1	1
0	1	x

(b)

x	1	x
1	1	1
0	0	0

(c)

x	1	0
1	1	0
x	1	0

(d)

x	0	0
1	1	0
x	1	x

(e)

0	0	x
0	1	1
x	1	x

(f)

x	1	x
0	1	1
0	0	x

(g)

x	1	x
1	1	0
x	0	0

(h)

x	x	x	x
0	1	1	0
x	x	x	x

(i)

(continued)

$$\begin{vmatrix} x & 0 & x \\ x & 1 & x \\ x & 1 & x \\ x & 0 & x \end{vmatrix}$$
(j)

$$\begin{vmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ p & 1 & q \end{vmatrix}$$
(k)

$$\begin{vmatrix} 0 & 1 & p \\ 0 & 1 & 1 \\ 0 & 1 & q \end{vmatrix}$$
(l)

$$\begin{vmatrix} p & 1 & q \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{vmatrix}$$
(m)

$$\begin{vmatrix} p & 1 & 0 \\ 1 & 1 & 0 \\ q & 1 & 0 \end{vmatrix}$$
(n)

(where $\overline{p \cdot q} = 1$ and x's are don't care positions)

figure 1.3 templates used in chin's algorithm.(where (a)-(h) --> thinning templates, (i)-(j)-->restoring templates,(k)-(n)--> trimming templates.)

Algorithm

Step C1 :

If an window matches with any of (i) to (j) templates then the candidate pixel can not be deleted and move to the next position and go to step c1 else goto dtep c2.

Step C2:

If an image window matches with any of (a) thru (h) templates then the candidate pixel will be deleted for the next iteration else move to the next position.

If any pixel is deleted then go to step c1 else go to dtep c3.

Step C3:

If an image window matches with any of (k) to (n) templates then the candidate pixel will be deleted.

Step C4 : STOP.

1.1.4 Pal and Bhattacharyya's thinning algorithm [Pal 89]

In this parallel thinning algorithm the binary pattern consists of those pixels those are 1's. In this algorithm a5x5 window is sliding from left to right and from top to bottom fashion over the image.

P_{99}	P_{9J}	P_{2J}	P_{3J}	P_{33}
P_{9i}	P_{9}	P_{2}	P_{3}	P_{13}
P_{8i}	P_{8}	P_{1}	P_{4}	P_{14}
P_{7i}	P_{7}	P_{6}	P_{5}	P_{15}
P_{77}	P_{7J}	P_{6J}	P_{5J}	P_{55}

figure 1.4 A 5x5 window used in Pal and Bhattacharyya's algorithm

A vertical stroke of width 2 is guarded by peeping of its edges. So a point on west edge can be preserved only if it is not on a corner and its east neighbour is on an edge. The horizontal and vertical straight lines can be preserved by if one of its four templates in figure 1.6 matches.

x	x	x	x	x
1	1	1	0	x
1	1	1	1	x
x	1	1	1	x
1	x	1	1	x

(a)

1	x	1	1	x
x	1	1	1	x
1	1	1	1	x
1	1	1	0	x
x	x	x	x	x

(b)

x	1	1	x	1
x	1	1	1	x
1	1	1	1	x
x	0	1	1	1
x	x	x	x	x

(c)

x	x	x	x	x
x	0	1	1	1
x	1	1	1	1
x	1	1	1	x
x	1	1	x	1

(d)

figure 1.5 horizontal and vertical line preserving templates.

Algorithm

*Step PB*1:

all the basic steps of Holt et al thinning algorithm except loops.

Step PB2:

If the 5x5 window matches with any of the four [figure1.5 (a) to (d)] templates then the candidate pixel will be deleted, otherwise goto step PB1 after sliding the window by one position.

Step PB3:

If any pixel is deleted in the above process (step PB1 and PB2) then go to step PB1 otherwise STOP.

1.1.5 Guo and Hall's thinning algorithm

In this algorithm they had assumed a 3x3window(figure1.1.)is sliding over the binary image. It is an two pass parallel template matching thinning algorithm.

Definition 1.1.5.1

$C(P_1)$ is defined as the number of distinct 8-connected components of 1's in P_1's eight neighbourhood. $C(P_1) = 1$ implies P_1 is 8-simple when P_1 is a boundary pixel.

Definition 1.1.5.2

$N(P_1)$ is used to detect endpoint and which can be helpful to achive thinner results where

$$N_2(P_1) = (P_9 \vee P_2) + (P_3 \vee P_4) + (P_5 \vee P_6) + (P_7 \vee P_8)$$

$N_1(P_1) = (P_2 \vee P_3) + (P_4 \vee P_5) + (P_6 \vee P_7) + (P_8 \vee P_9)$

N_1 and N_2 each break the ordered set of P's neighboring pixels into four pair of adjoining pixels and count the number of pairs which contains one or two 1's.

AlgoriΘthm

PASS I

*Step G*1.1

An element is to be removed when it is a boudary point. This often identified by $2 \le N(P_1) \le 3$ and $C(P_1) = 1$ if a pixel is on boundary point and $(P_3 \vee P_4 \vee \bar{P}_6) \vee P_5 = 0$ is there then the candidate pixel is to be deleted for the next pass 2.

*Step G*1.2

Scan the window one position at a time or scan all the positions at a time and apply G1.1.

*Step G*1.3

when all the pixels of the given image have been processed and at least one candidate pixel has become zero then goto pass 2 else exit.

*Step G*2.1

An element is to be removed when it is a boudary point. This often identified by $2 \leq N(P_1) \leq 3$ and $C(P_1) = 1$ if a pixel is on boundary point and $(P_7 \vee P_8 \vee \bar{P}_2) \vee P_9 = 0$ is there then the candidate pixel is to be deleted for the next pass 1.

Step G2.2

Scan the window one position at a time or scan all the positions at a time and apply G2.1.

Step G2.3

when all the pixels of the given image have been processed and at least one candidate pixel has become zero then goto pass 1 else exit.

2.1 TESULT AND CONCLUSION

We have studied also implememted five parallel template matching thinning algorithms of which two algorithms such as widely used Zhang and Suen's algorithm and Guo et al's algorithms are two pass algorithms other are one pass algorithms. In the one pass algorithms,Chin's algorithm explicitely describe the templates used in algorithm, and others descrfibe the property of the templets used in the thinning process. A set of output (Figure) of different algorithm for a particular set input shows the quality of thinning algorithms. Hardware implementations are very simple. But no one is a generalized thinning algorithm. So new thinning algorithm development scope is always open in the field of Image Processing/Computer Vision.

CHAPTER II

A STATISTICALLY DISTRIBUTED RANDOM BINARY IMAGE GENERATOR

INTRODUCTION

In Comuter Vision, image is a matrix of integer values which is indicating gray value at a particular position of an image. By binarization method the gray level image can be converted to a binary image. There area number of image processing operations [Stoffel 82], such as coding, contour following and skeletonization, performed on a binary images (ie, binary patterns or bilevel images/patterns). Classification of binary patterns in pattern recognition [Ullman 74, Shapro 78]. Biological or medicalimage processing [Trussel 81], Engineering Drawing [Pferd 81], map processing [Musavi 88], and other machine manipulations of imaginary data often incorporet binary image processing at some stage in their application. In the emperical study of the average performance of various image processing and classification algorithms on binary images, it is necessary to generate a statistically distributed binary image and also necessary to analyze their tolerance and sensitivity to noise [Zhou 91]. In this chpter we have presented a statistically distributed binary image generation method.

2.1 DEFINITION AND TERMINOLOGIES :

Definition 1 :

A binary image $B = [b_{ij}]$ be represented by

$$b_{ij} = \begin{cases} 1 \text{ , Object} \\ 0 \text{ , background} \end{cases}$$

where $i = 1,2,......n$; $j = 1,2,.........,m$ and $n \times m$ is the size of the binary image matrix B.

The real world image contains some natural patterns. Mostly they are connected. The connectivity of an image is defined by the following definitions.

P_9	P_2	P_3
P_8	P_1	P_4
P_7	P_6	P_5

Fig 1: The 4-neighbour and 8-neighbour of the point P.

P_1 = Central element, P_2 = North, P_4 = East, P_6 = South,
P_8 = West, P_3 = North-East, P_5 = South-East, P_7 = South-West,
P_9 = North-West.

Definition 2: In a binary image four elements, namely, top(P_2), bottom (p_6), left (P_8) and right (P_4) are said to be 4-neighbours of the elements P_1 (or 4-adjacent to P_1) (Fig.1). If two subsets U and V of S (say) there exist at least one element of U which is 4-adjacent to atleast one element of V, then U is said to be 4-adjacent to V.

Definition 3: In a binary image four diagonal elements of P_1 (Fig. 1) are said to be 8-neighbours of P_1 (or 8-adjacent to P_1). If for two subsets U & V of S there exists at least one element of U which is 8-adjacent to at least one element of V then U is said to be 8-adjacent to V.

Definition 4: An 8-path (4-path) $\Pi_n(p,q)$ of length n from p to q is a sequence of points (elements) $\langle p = p_0, p_1, \ldots\ldots p_n = q\rangle$ such that p_i is an 8-neighbour (4-neighbour) of p_{i-1} and $p_i \in B$ where B is a binary image and $1 \le i \le n$.

Definition 5: For two different points p and q of B, where B is a binary image, the point p is said to be 8-path connected (4-path conected) or simply 8-conected (4-conected) to q if there exists an 8-path (4-path), $\Pi_n(p,q)$ in B.

2.2Uniformly Distributed Binary Image

A binary image can be generated by using an uniformly distributed random number generator. We are generating a uniformaly distributed random number which lies between the ranges of gray levels, scanned or digitized through a scaning device/CCD camera for an image of a scene. On each grey values apply a thresold binaryzation. The grey values of the image can be normalised within the range 0 to 1 then choose a thresold value (0.5 say). This will give a random pattern but it does not guarantee the conectivity of the pattern. A 4-connected (8-conected) random binary pattern can be generated by the following methods.In this method a binary uniformly distributed random number (0 or 1) generator generates 0's and 1's and placing them on its 4-adjacents (8-adjacents) according to its incoming sequence then move to the upper level and find next 1 and apply the above method recursively and when it dose not find any 1 of its adjacent then come down to a step. This ultimate pattern will be bounded by 0's only.

2.3 Normally Distributed Binary Image

A normally distributed binary image can be generated by using a normally distributed random number generator. Since the image is binary so we have to select the positions on the image plane such that it will produce an edge line binary image. We can define a normally distributed binary image as follows :

Definition 6: A gray level image G is said to be normally distributed or a mixture of normally distributions if its histogram of gray values of the image is approximated by a smooth curve and the histogram looks like a normal distribution curve or a mixture of normal distribution curves (ie multi-peak) then the image is said to be normally distributed.

Definition 7: A binary image is said to be normally distributed if its original (gray value) image is normally distributed by definition 6 and also its converted binary image by edge detection method gives some natural patterns of 1's then the binary image is said to be normally distributed.

Since the binary image is normally distributed, so it contains (0,1) with natural patterns of lines and points. Natural pattern formation by using (0,1) is position dependent. We deviced a method by which we can generate the positions of 1's in a zero matrix, so that this matrix contains some natural patterns then this matrix is B, a normally distributed binary image. Now we shall generate the positions of 1's in B (initially 0). Since we can assume B as a pixel matrix so the distance between two pixels is fixed in a perticular direction. So there is no deviation on distance, only deviation may exists on the value of orientation θ (say). So we shall generate a normally distributed angle θ with respect to a center.

The parameters of a normally distributed random direction generator are standered deviation (σ_θ), mean (μ_θ) and the initial value of $\theta = \theta_0$ (say). Mathematically we can write

$$\theta = N(\theta_0, \mu_\theta, \sigma_\theta).$$

This θ is a real number, in degree or radian. So make this θ in degree to its nearest integer value take a modulus over 360^0. If θ is negative then it convert to a positive valueby $360^0+\theta$. Then choose the θ nearest of an element of the set

$\{0^{\circ}, 30^{\circ}, 45^{\circ}, 60^{\circ}, 90^{\circ}, 120^{\circ}, 135^{\circ}, 150^{\circ}, 180^{\circ}, 210^{\circ}, 225^{\circ}, 240^{\circ}, 270^{\circ}, 300^{\circ}, 315^{\circ}, 330^{\circ}, 360^{\circ}\}$.

According to the value of the angle choose corresponding pixel to be 1. This is shwon in the figure2 and table 1.

```
       6 5 4
       . . . . ..  3
   8 . . . . . .   2
   9 . . . . . .   1
  10 . . . . . .  16
  11 . . . . . .  15
    12 13 14
```

figure. 2:pixel with the corresponding number, 0 is the central element

Table1: Angle and corresponding position of pixel.

1	0°	5	90°	9	180°	13	270°
2	30°	6	120°	10	210°	14	300°
3	45°	7	135°	11	225°	15	315°
4	60°	8	150°	12	240°	16	330°

Formal Algorithm:

1> The binary image matrix $B = [b_{ij}] \; \forall \; i,j \in [0,..m;0,..n]$ where $b_{ij} = 0 \; \forall \; i,j$

2> We shall generate $w \ll n^2$ elements of 1

3> Initialize value of $\theta = \theta_0$, $\mu = \mu_\theta$ and $\sigma = \sigma_\theta$

4> Generate a normally distributed random number

$\theta = N(\theta,\mu,\sigma)$ for a seed of uniformlly distributed random number.

5> $\theta = \theta * \mathrm{VNORM}(\mu,\sigma) + \theta_0$

6> θ = Integer(θ), assume θ is in degree, if not convert it to its corresponding values of degree

7> $\theta = \theta \bmod 360$

8> If θ is negative then $\theta = \theta + 360$

9> Select θ to its nearest value of the set

$$\{0^\circ, 30^\circ, 45^\circ, 60^\circ, 90^\circ, 120^\circ, 135^\circ, 150^\circ, 180^\circ, 210^\circ, 225^\circ, 240^\circ, 270^\circ, 300^\circ, 315^\circ, 330^\circ, 360^\circ\}.$$

Also choose its corresponding position as shown in table 1

10> Fill that position by 1 and its 8-neighbour

11> $\theta_0 = \theta$

12> Repeate the steps 1 to 11 untill it generates w points.

2.4 Experimental Results and Conclusions

We have carried out a number of experiments to generate different binary images. Few results are shown in figure 3-6 contains some results of 4-connected uniformly distributed binary images.Fig.7-10 contains some results of normally distributed binary images. In the study of the average performance of template matching thinning algorithms [Pal 91] used uniformly distributed binary image. The same study can be performed by using normally distributed binary image. Alsowe can make this binary image more realistic by adding some noise [Zhou 91]. This idea can be expanded for the random natural line segment ie map line segment generation usin normally distributed random number generation for the two parameter in polar co-ordinate system (r,θ) where r is the length of the line segment and θ is the orientation.

CHAPTER III

ANALYSIS OF TEMPLATE MATCHING THINNING ALGORITHMS USING MARKOV PROCESS

In this chapter we have consider the analysis of template matching thinning algorithms for measuring their time complexity. The main objective of measuring time complexity is to measure the total amount of time requirement by the algorithm and the average number of iterations required to converge the thinning process. Here we have proposed a probabilistic model using Markov process for measuring average time complexity of the template matching thinning algorithms.

Using the proposed model it is also possible to compute a bound on the number of iterations requried and the requirement of the average time to complete a thinning process applied on an uniformly distributed binary image.

3.1Markov Process

A stochastic process can be defined quite as any collection of random variables X(t), t $\in$ T, defined on a common probability space, where T is subset of $(-\infty,\infty)$ and is thought of as the time parameter set. The process is called a continuous parameter process if T is an interval having positive length and a discrete parameter process if T is a subset of the integers.

If the random variable X(t) all takes on values from the fixed set S, then the S is called the state space of the process. Many stochastic process of theoretical and applied interest possess the property that ,given the present state of the process ,the past history does not affect conditional probability of events defined in terms of the future. such process is called Markov process.
Many systems have the property that given the present state ,the past states have no influence on the future.This property is called the Markov property , system having the property are called Markov chains. The Markov property is defined precisely by the requirement that

$$P\left(X_{n+1}= x_{n+1} \;\middle|\; X_0= x_0,\ldots\ldots X_n = x_n\right) = P\left(X_{n+1}= x_{n+1} \;\middle|\; X_n= x_n\right) \qquad \text{--------(1)}$$

for every choice of the nonnegative integer n and the numbers x_0, ...,x_{n+1}, each in S. The conditional probabilities

$$P\left(X_{n+1}= x_{n+1} \;\middle|\; X_n= x_n\right)$$

are called the transition probabilities of the chain.

3.2Markov Pure Jump Process

Let X(t) denote the state of the system at time t,defined by

$$X(t) = \begin{cases} x_0 & 0 \le t < \tau_1, \\ x_1 & \tau_1 \le t < \tau_2, \\ x_2 & \tau_2 \le t < \tau_3 \\ & \vdots \end{cases} \qquad \text{-----(2)}$$

The process defined by (2) is called a jump process.

A pure death process is defined as a birth and death process with $\lambda_x = 0$ where λ_x is the rate of birth and μ_x is known as rate of death. The forward equation for a birth and death process can be written as

$$P'_{xy}(t) = \lambda_{y-1} P_{x,y-1}(t) - (\lambda_y + \mu_y) P_{x,y}(t) + \mu_{y+1} P_{x,y+1}(t) \quad \text{-----(3)} \quad [\ t \geq 0\ ]$$

we set $\lambda_{-1} = 0$

For pure death process , from (3) we have

$$P'_{xy}(t) = \mu_{y+1} P_{x,y+1}(t) - \mu_y P_{x,y}(t)$$

$$P_{xy}(t) = 0 \quad \text{if } y > x \text{ and } t \geq 0$$

$$\Rightarrow P'_{xx}(t) = -\mu_y P_{xx}(t)$$

since $P_{xx}(0) = 1$ and $P_{xy}(0) = 0$ for $y < x$

$$\Rightarrow P_{xx}(t) = e^{-\mu_x t}$$

Thus

$$P_{xy}(t) = \mu_{y+1} \int_0^t e^{-\mu_y (t-s)} P_{x,y+1}(s)\, ds$$

=>

$$P_{x,x-1}(t) = \mu_x \int_0^t e^{-\mu_{x-1}(t-s)} P_{x,x}(s)\, ds$$

$$= \mu_x \; e^{-\mu_{x-1} t} \int_0^t e^{(\mu_{x-1} - \mu_x) s} ds$$

$$= \frac{\mu_x}{\mu_{x-1} - \mu_x} \; e^{-\mu_{x-1} t} (e^{(\mu_{x-1} - \mu_x) t} - 1)$$

$$= \frac{\mu \; x}{\mu \; (x-1-x)} \; [e^{-\mu x t} - e^{-\mu (x-1) t}]$$

considering $\mu_x = \mu . \; x$

$$= x \; e^{-\mu (x-1)t} [1- e^{-\mu t}]$$

$$P_{x,x-2}(t) = \mu_{x-1} \int_0^t e^{-\mu_{x-2} (t-s)} P_{x,x-1}(s) \; ds$$

$$= \mu \; (x-1) \; e^{-\mu (x-2)t} x \int_0^t e^{-\mu (x-1)t} [1- e^{-\mu s}] \; e^{\mu (x-2)s} ds$$

$$= \mu \; (x-1) \; e^{-\mu (x-2)t} x \int_0^t e^{-\mu s} [1- e^{-\mu s}]ds$$

$$= \frac{(x-1)}{2} x \; e^{-\mu \; (x-2)t} (1- e^{-\mu t})^2$$

$$= \binom{x}{2} e^{-\mu (x-2)t} (1- e^{-\mu t})^2 .$$

$$\Rightarrow P_{xy}(t) = \binom{x}{x-y} e^{-\mu y t}(1- e^{-\mu t})^{x-y} \quad \text{-------}(4)$$

here we consider t = total # of pixels, the delay time of the process.

Modelling the Thinning Problem

Now the thinning problem can be viewed as applying an algorithm A on the image $I_r = x_r$ to get $I_{r+1} = x_{r+1}$ where $x_{r+1} \le x_r$, r /= 0,1,...k-1. The stopping condition is when $x_{r+1} = x_r$; x_r = # of 1's after rth iteration.

$$\boxed{I_0 = x_0} \xrightarrow{A} \boxed{I_1 = x_1} \xrightarrow{A} \boxed{I_2 = x_2} \cdots\cdots\cdots$$

$$\xrightarrow{A} \boxed{I_{k-1} = x_{k-1}} \xrightarrow{A} \boxed{I_k = x_{k-1}}$$

fig3.1 thinning of an image I_0 by the algorithm A in K iteration.

As we see that the intermediate thinned image I_{r+1} depends only on the immediate past thin image I_r , not only on the past states I_0 I_1 , I_2 ,....,I_{r-1}. Again as we assign the probability of deletion of one '1', we have the randomness.

Thus we can represente the thinning algorithm by the Markov process. Thus following equation (1) we have

$P(I_k \mid I_0, I_1, \ldots, I_{k-1}) = P(I_k \mid I_{k-1})$.

Now we consider each iteration in two ways

1> Pure Death Process

2> Poisson Distribution

(1) Consider Pure Death Process, we have the distribution function

$$P_{xy}(t) = \binom{x}{x-y} e^{-\mu yt}(1- e^{-\mu t})^{(x-y)} \text{--------(4)}$$

as defined in equation (4).

Thus we can get

$P (I_0= x_0 ,I_1= x_1, \qquad ,I_{k-1}= x_{k-1},I_k= x_{k-1})$

$= P (I_0 = x_0) P (I_1=x_1 \mid I_0=x_0) P (I_2=x_2 \mid I_1=x_1)\ldots$

$\ldots P (I_{k-1}=x_{k-1} \mid I_{k-2} = x_{k-2})P (I_k =x_{k-1} \mid I_{k-1}=x_{k-1})$

$= \Pi(0)\binom{x_0}{x_0-x_1} e^{-\mu x_1 t}(1- e^{-\mu t})^{(x_0-x_1)} \binom{x_1}{x_1-x_2} e^{-\mu x_2 t}(1- e^{-\mu t})^{(x_1-x_2)}\ldots\ldots$

$\ldots\ldots\binom{x_{k-2}}{x_{k-2}- x_{k-1}} e^{-\mu x_{k-1} t}(1- e^{-\mu t})^{(x_{k-2} - x_{k-1})}$

$\binom{x_{k-1}}{0} e^{-\mu x_{k-1} t} (1- e^{-\mu t})^0$

$= \Pi(0) \text{ (factorial) } e^{-\mu t\sum^{k-1} x_r - \mu t x_{k-1}} (1 - e^{-\mu t})^{(x_0 - x_{k-1})}$

where $\Pi(0) = P(I_0= x_0)$ and factorial $= \prod_{r=0}^{k-1} \binom{x_r}{x_r - x_{r+1}}$

$P (I_0= x_0 ,I_1= x_1, \qquad ,I_{k-1}= x_{k-1},I_k= x_{k-1}) < 1$

(approximating x_r by $x_0 P_{11}^{\,r}$)

=>

$$\Pi(0)\ (x_0)\ e^{-\mu t \sum_{i} x_0 p_{11}^{r} - \mu t x_0 p_{11}^{k-1}}\ (1 - e^{\mu t})^{x_0 (1 - p_{11}^{k-1})} < 1$$

as factorial $\geq x_0$

=> $$\log \Pi(0) + \log x_0 - \frac{\mu t p_{11} x_0}{p_{10}}(1 - p_{11}^{k-1}) - \mu t x_0 p_{11}^{k-1} + x_0(1-p_{11}^{k-1})\log(1-e^{-\mu t}) < 0$$

=>

$$\log \Pi(0) + \log x_0 - \frac{\mu t p_{11} x_0}{p_{10}} - p_{11}^{k-1}\left(\mu t x_0 - \frac{\mu t p_{11} x_0}{p_{10}}\right) < 0$$

as $\log (1- e^{-\mu t})$ ---> 0 as t becomes large.

=> $$p_{11}^{k-1}\left(\mu t x_0 - \frac{\mu t p_{11} x_0}{p_{10}}\right) > \left(\log \Pi(0) + \log x_0 - \mu t x_0 \frac{p_{11}}{p_{10}}\right)$$

=>

$$p_{11}^{k-1} < \frac{\left(\log \Pi(0) + \log x_0 - \mu t x_0 \frac{p_{11}}{p_{10}}\right)}{p_{11}^{k-1}\left(\mu t x_0 - \frac{\mu t p_{11} x_0}{p_{10}}\right)} = kk$$

=> $$\log p_{11}\ (k-1) < \log kk$$

=>

$$k-1 > \frac{\log kk}{\log p_{11}}$$

=>

$$k > 1 + \frac{\log kk}{\log p_{11}}$$

(2) Now we consider the Poisson distribution. It is because the # of deletion of '1' in first few iterations is very large where as that of the last few iterations is very low. We consider the Poisson distribution of constant parameter λ.

The probability distribution function for i-th iteration ,

$$p_i = \frac{\lambda^{i}\ e^{-\lambda}}{i!} \quad \text{where } \lambda = p_{10}\ x_0$$

Thus

$$P\ (\ I_0 = x_0\ , I_1 = x_1,\qquad , I_{k-1} = x_{k-1}, I_k = x_{k-1}) < 1$$

$$= \Pi(0) \prod_{i=1}^{k} \frac{\lambda^{i}\ e^{-\lambda}}{i\ !} < 1$$

$$\Rightarrow \qquad \frac{\lambda^{\sum i}\ e^{-\lambda\ k}}{\Pi\ i\ !} \qquad < 1 \text{ [as each } p_i \le 1 \text{]}$$

$$\text{as } \Pi(0)\ \ \lambda^{\sum i}\ e^{-\lambda\ k} > 0 \text{ and}$$

$$i\ ! < \Pi\ i^{i} < \Pi\ k^{i} = k^{\sum i}$$

$$\Rightarrow \frac{1}{\Pi\ i\ !} > \frac{1}{k^{\frac{k(k+1)}{2}}}$$

$$\Rightarrow \qquad \frac{\lambda^{\frac{k(k+1)}{2}}\ e^{-k\lambda}}{k^{\frac{k(k+1)}{2}}} < 1$$

$$\Rightarrow \quad \frac{k(k+1)}{2} \log \lambda - k\lambda < \frac{k+1}{2}\,)\, \log k$$

$$\Rightarrow \quad k + 1 < \frac{2\lambda}{\log \lambda - \log k}$$

since (k+1) is a positive number , we have

$$\log \lambda - \log k > 0$$

$$\Rightarrow \quad k < \lambda$$

3.3 Time Complexity Analysis

There are two important measures of any algorithm,namely

1> time requirement

2> space requirement

Generally the time and space complexity is measured as the function of size of the input. If for a given size the coplexcity is chosen as the maximum one then it is called worst case complexity.

If the complexity is taken as average requirement of time for a given size ,then it is called the average time coplexcity.

The main objective of measuring time complexity is to measure the performance of the template matching thinning algorithm. A large amount of thinning algorithms have been proposed so far, but no serious attempt ,except in [Pal91] is to measure average time complexity.

We have proposed a stochastic model for measuring average complexity of template matching thinning algorithms. Using the proposed model it is possible to compute a bound on the number of iteration required.

Table 1 : The # of templates and probability of matching

algorithm used	# of elements in a template	# of templates used to delete	probanility of template matching
Zhang&Suen	9	34	0.06640625
Chin	9	48	0.09375000
Holt	16	4823	0.07359314
Pal&Bhattacharyya	25	2608640	0.07774353
Guo & Hall	9	32	0.0625

As shown in table 2 the average number of iterations required for thinikg have been obtained experimentally for images of different sizes.

Table 2: avetage # of iterations required for images with different sizes.

size of image --> / algorithm used ↓	32 x 32	64 x64	100 x 100
Zhang&Suen	2.5	3.16	5.25
Chin	5.5	13.1	15.5
Holt	3.833	6.4	7.5
Pal&Bhattacharyya	3.33	6.33	7.0
Guo & Hall	3.5	6.11	8.0

Table 3 displays the percentage of matched elements encountered at different iterations.

Table 3 : Average % of matched elements in the iterations (32x32)

iterations --> / algorithm used ↓	1	2	3	4	5	6
Zhang&Suen	57.8	7.01	0.066	-	-	-
Chin	39.69	21.32	6.809	0.925	0.238	-
Holt	54.299	10.3	0.198	0.0	-	-
Pal&Bhattacharyya	57.02	7.206	0.1983	-	-	-
Guo & Hall	54.145	12.858	1.75	0.429	-	-

REFERENCES :

[Chin 87]

Chin R. T, Wan H K , Stover D L , Iverson R D. " A one pass thinning algorithm and its parallel implementation".Computer Vision,Graphics and Image Processing,vol 40 , no 1, pp-30-40, oct, 1987.

[Guo 89]

Guo Z , Hall R W." Parallel Thinning with Two Subiteration Algorithm",CACM,Vol 32, No 3, March 1989, pp 359- 373.

[Holt 87]

Holt C M ,Stewart A, Clint M and Perrott R H. " An Improved Parallel Thinning Algorithm". CACM,Vol 30, No 2, Feb ,1987, pp- 156-160.

[Musavi 88]

Musavi M T,Shirvaikar M V, Ramanathan E and Nekovei A R. "A Vision Based Method To Automatic Map Processing".Pattern Recognition,Vol.21,No.4,PP.319-326,1988.

[Pal 89]

Pal S and Bhattacharyya P."A Shape Preserving One Pass Parallel Thinning Algorithm".Indian Institute of Management,Calcutta, working papaer series No 123(89),1989.

[Pal 91]

Pal S and Bhattacharyya P. "Analysis of Template Matching Thinning Algorithms".Pattern Recognition.(Accepted)

[Pferd 81]

Pferd W and Stocker G. "Optical Fibers for Scanning Digitizers",Bell Sys. Tech. J. 60 , 1981, 523-534.

[Shapiro 78]

Shapiro L G , "Inser Matching of Line Drawings in a Syntactic Pattern Recognition System", Pattern Recognition,10, 1978,313-321.

[Stoffel 82]

Stoffel J C (Ed) Graphical and Binary Image Processing and Applications.Artech House,Dedham,M A ,1982

[Trussell 81]

Trussell H J ,Processing of X-ray Images,Proc.IEEE 69,1981,615-627

[Ullman 74]

Ullman J R ,'Binarization Using Associate Addressing , Pattern Recognition,6,1974, 127-135.

[Zhang 84]

Zhang T Y and Suen C Y." A FastParallel Algorithmfor Thinning Digital Patterns".CACM, Vol 27, No 3, pp 236 - 239,March 1984.

[Zhou 91]

Zhou X and Gordon R. "Generation of Noise in Binary Images".C V G I P: Graphical Models and and Image Processing. Vol. 53, no. 5, Sept.1991, 476-478.

BY ZHANG AND SUEN

BY GUO AND HALL

BY CHIN ET AL

ORIGINAL THINNED

CONTD......

BY HALT ET AL

BY PAL AND BHATTACHARYYA

ORIGINAL THINNED

FIGURE - I

BY CHIN ET AL

BY GUO AND HALL

BY HALT ET AL

THINNED

ORIGINAL

CONTD....

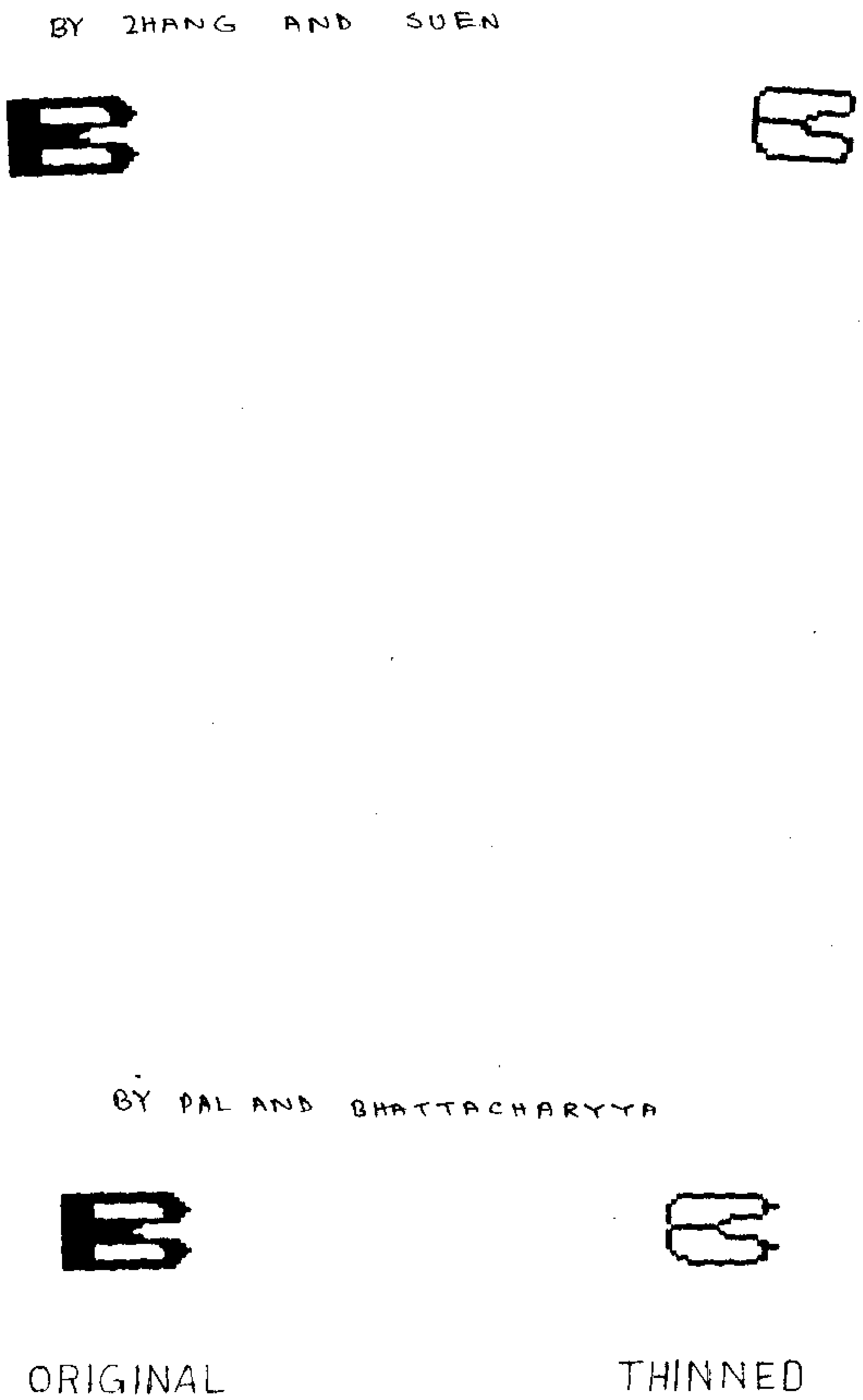
BY ZHANG AND SUEN
BY PAL AND BHATTACHARYYA
ORIGINAL
THINNED

FIGURE-II

BY HALT ET AL

BY PAL AND BHATTACHARYYA

BY GUO AND HALL

ORIGINAL

THINNED

CONTD.....

BY ZHANG AND SUEN

BY CHIN ET AL

ORIGINAL THINNED

FIG-III

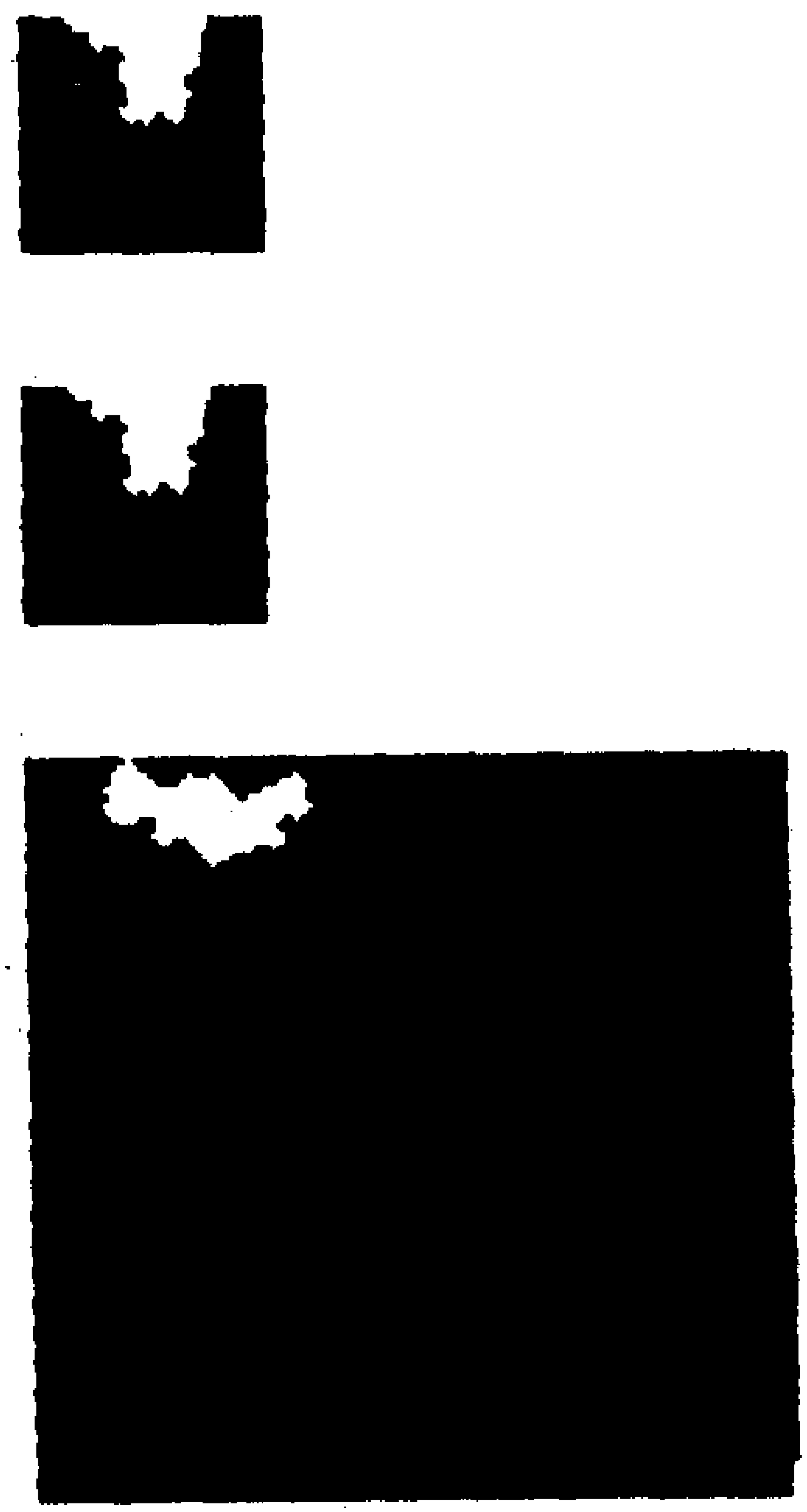

FIGURE -3-6

FOUR CONNECTED IMAGES

FIGURE - 7 - 10

NORMALLY DISTRIBUTED IMAGES