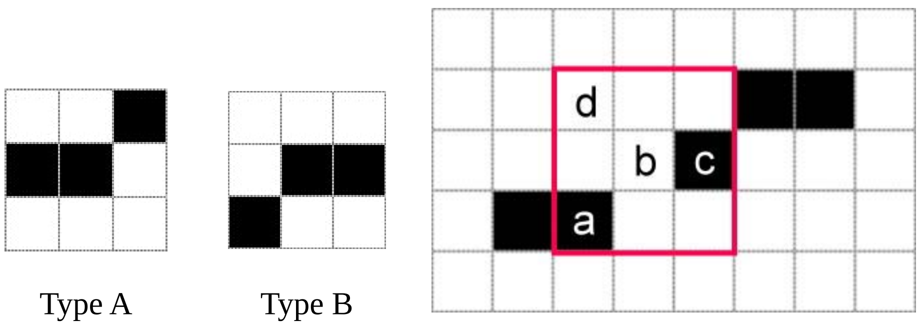
Chapter 2

Connectionist Model with 1-neighborhood for Line Linking

2.1 Introduction

In this chapter, we apply the connectionist model on line linking. In the line linking implementation, we consider a point whether it is a part of a line by the feature of the neighboring points. If the neighboring points have line feature, then we test the neighboring points whether they can become a line with interesting point. We increase the strength to a point which has ability to become a line with its neighbors and decrease if not. For example, suppose there are two 3×3 pattern types that shown in Figure 2.1(a) can compose of any line. Figure 2.1(b) shows a line example. Consider the point b, interesting point, and its neighboring points in the Figure 2.1(b). Clearly, the point b with its eight neighbors match the type B, point a and c match the type A, and point d does not match any type. Here, point b will get the reinforcement from point a and c because the type of point a and c can match a line with type of point b. The point d without give any reinforcement to point b. If point b gets enough reinforcement, then the point b is regard as a line point.



(a) (b)

Figure 2.1. (a) Two pattern types, (b) A line example.

Because of this idea, we use connectionist model to achieve our goal. Jayanta

Basak, Bhabatosh Chanda, and Dwijesh Dutta Majumder in 1994 [4] proposed a connectionist model to solve line linking problem. The model is designed based on cellular neural network. We follow the network and modify some formulas to suit our problem.

2.2 **Line Linking**

In the implementation of the connectionist model for line linking, the model is considered to consist of $m \times n$ processing elements (PEs) if the image contains $m \times n$ pixels. Each PE only connects with its eight neighbors, 1-neighborhood [5]. Figure 2.2 shows a 4×4 1-neighborhood network. In order to know the line feature of each PE, Figure 2.3 shows sixteen 3×3 patterns as the line feature. The line detection operator is used to calculate strength of each point for every pattern. A point belongs to some pattern if it gets larger strength in that pattern than other patterns. We regard the pairs of strength and corresponding pattern type of all PEs as the initial state of the network. Then, network adjusts until network convergence, i.e. network output is not changed. After network convergence, the network output is the result of line linking.

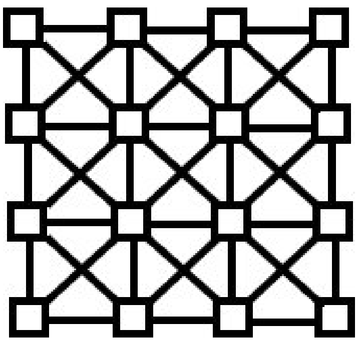


Figure 2.2. A 4×4 1-neighborhood network connection.

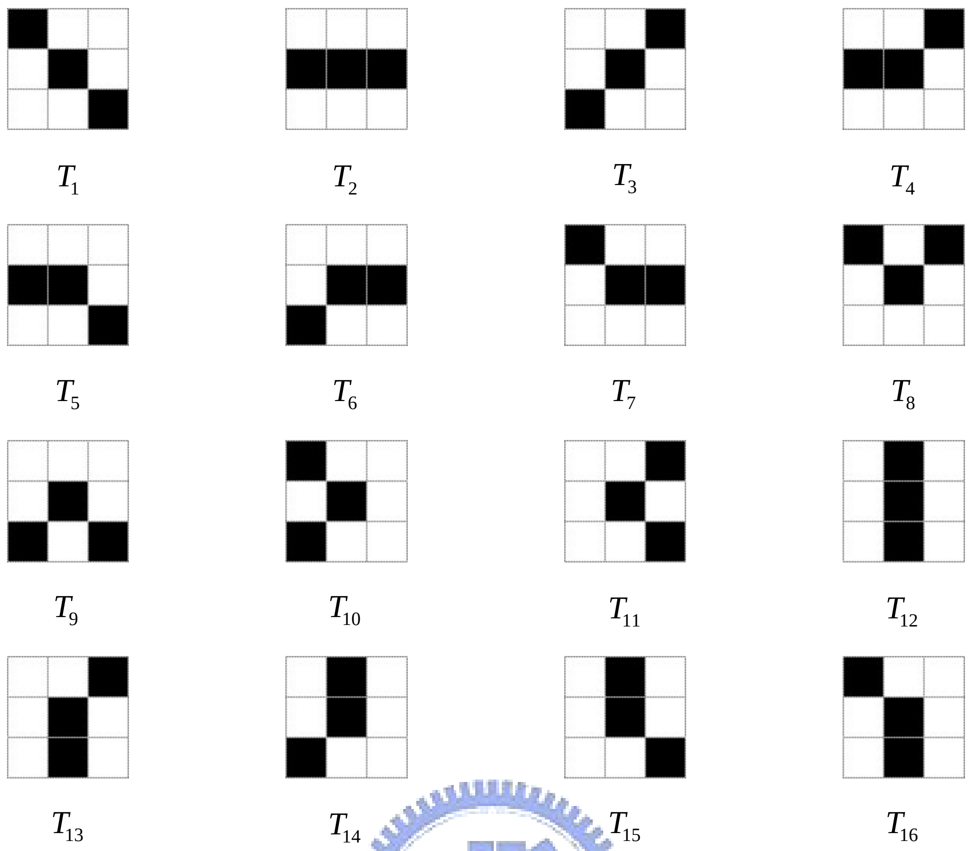


Figure 2.3. Feature patterns in line feature detection.(Basak [4])

2.2.1 **Line Detection Operator**

For each point, we have to know what pattern the point similar to and set the largest strength for that pattern. Figure 2.4 shows the coordinate relationship between a point and its eight neighbors.

$(x-1, y-1)$	$(x-1, y)$	$(x-1, y+1)$
$(x, y-1)$	(x, y)	$(x, y+1)$
$(x+1, y-1)$	$(x+1, y)$	$(x+1, y+1)$

Figure 2.4. A point and its eight neighbors.

There are sixteen formulas to calculate strength for each pattern shown in Table 2.1 [6]. We use $f(x, y)$ to represent the gray value of the point (x, y). We set $f(x, y)$ equal to 1 if the point is black and set to 0 if the point is white.

Let R_i , $i = 1,\ldots,16$, denote the response of the masks from T_1 to T_{16} . Suppose that the sixteen masks are run individually through an image. For a certain point in the image, if $R_i > R_j$, $\forall i \neq j$, that point is said to be more likely associated with the pattern T_i . The responses are all in the interval [-1 1].

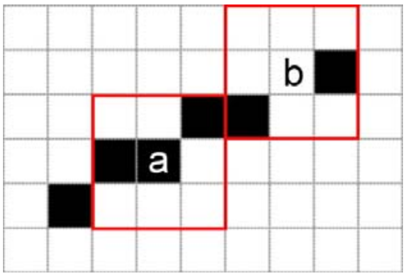
Table 2.1. Sixteen line detection operator for strength calculation.

type	mask	Pattern matching formula
T_1 :	$\begin{array}{\|c\|c\|c\|}\hline 1/3 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & 1/3 \\ \hline\end{array}$	$R_1 = \frac{1}{3}(f(x-1,y-1)+f(x,y)+f(x+1,y+1)) -$ $\frac{1}{6}\times(f(x-1,y)+f(x-1,y+1)+f(x,y+1)+f(x,y-1)+f(x+1,y-1)+f(x+1,y))$
T_2 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & -1/6 \\ \hline 1/3 & 1/3 & 1/3 \\ \hline -1/6 & -1/6 & -1/6 \\ \hline\end{array}$	$R_2 = \frac{1}{3}(f(x,y-1)+f(x,y)+f(x,y+1)) -$ $\frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)+f(x+1,y-1)+f(x+1,y)+f(x+1,y+1))$
T_3 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & 1/3 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline 1/3 & -1/6 & -1/6 \\ \hline\end{array}$	$R_3 = \frac{1}{3}(f(x+1,y-1)+f(x,y)+f(x-1,y+1)) -$ $\frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x,y-1)+f(x,y+1)+f(x+1,y)+f(x+1,y+1))$
T_4 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & 1/3 \\ \hline 1/3 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & -1/6 \\ \hline\end{array}$	$R_4 = \frac{1}{3}(f(x,y-1)+f(x,y)+f(x-1,y+1)) -$ $\frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x,y+1)+f(x+1,y-1)+f(x+1,y)+f(x+1,y+1))$
T_5 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & -1/6 \\ \hline 1/3 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & 1/3 \\ \hline\end{array}$	$R_5 = \frac{1}{3}(f(x,y-1)+f(x,y)+f(x+1,y+1)) -$ $\frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)+f(x+1,y-1)+f(x+1,y)+f(x,y+1))$
T_6 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & 1/3 \\ \hline 1/3 & -1/6 & -1/6 \\ \hline\end{array}$	$R_6 = \frac{1}{3}(f(x+1,y-1)+f(x,y)+f(x,y+1)) -$ $\frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)+f(x,y-1)+f(x+1,y)+f(x+1,y+1))$
T_7 :	$\begin{array}{\|c\|c\|c\|}\hline 1/3 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & 1/3 \\ \hline -1/6 & -1/6 & -1/6 \\ \hline\end{array}$	$R_7 = \frac{1}{3}(f(x-1,y-1)+f(x,y)+f(x,y+1)) -$ $\frac{1}{6}\times(f(x,y-1)+f(x-1,y)+f(x-1,y+1)+f(x+1,y-1)+f(x+1,y)+f(x+1,y+1))$

T_8 :	$\begin{array}{\|c\|c\|c\|}\hline 1/3 & -1/6 & 1/3 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & -1/6 \\ \hline \end{array}$	$R_8 = \frac{1}{3}(f(x-1,y-1)+f(x,y)+f(x-1,y+1)) - \frac{1}{6}\times(f(x,y-1)+f(x-1,y)+f(x,y+1)+f(x+1,y-1)+f(x+1,y)+f(x+1,y+1))$
T_9 :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline 1/3 & -1/6 & 1/3 \\ \hline \end{array}$	$R_9 = \frac{1}{3}(f(x+1,y-1)+f(x,y)+f(x+1,y+1)) - \frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x-1,y+1)+f(x,y-1)+f(x+1,y)+f(x,y+1))$
T_{10} :	$\begin{array}{\|c\|c\|c\|}\hline 1/3 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline 1/3 & -1/6 & -1/6 \\ \hline \end{array}$	$R_{10} = \frac{1}{3}(f(x-1,y-1)+f(x,y)+f(x+1,y-1)) - \frac{1}{6}\times(f(x+1,y+1)+f(x-1,y)+f(x-1,y+1)+f(x,y-1)+f(x+1,y)+f(x,y+1))$
T_{11} :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & 1/3 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & 1/3 \\ \hline \end{array}$	$R_{11} = \frac{1}{3}(f(x-1,y+1)+f(x,y)+f(x+1,y+1)) - \frac{1}{6}\times(f(x-1,y-1)+f(x-1,y)+f(x+1,y-1)+f(x,y-1)+f(x+1,y)+f(x,y+1))$
T_{12} :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline \end{array}$	$R_{12} = \frac{1}{3}(f(x-1,y)+f(x,y)+f(x+1,y)) - \frac{1}{6}\times(f(x+1,y-1)+f(x-1,y-1)+f(x-1,y+1)+f(x,y-1)+f(x+1,y+1)+f(x,y+1))$
T_{13} :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & -1/6 & 1/3 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline \end{array}$	$R_{13} = \frac{1}{3}(f(x-1,y+1)+f(x,y)+f(x+1,y)) - \frac{1}{6}\times(f(x+1,y+1)+f(x-1,y)+f(x-1,y-1)+f(x,y-1)+f(x+1,y-1)+f(x,y+1))$
T_{14} :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline 1/3 & -1/6 & -1/6 \\ \hline \end{array}$	$R_{14} = \frac{1}{3}(f(x-1,y)+f(x,y)+f(x+1,y-1)) - \frac{1}{6}\times(f(x+1,y+1)+f(x-1,y-1)+f(x-1,y+1)+f(x,y-1)+f(x+1,y)+f(x,y+1))$
T_{15} :	$\begin{array}{\|c\|c\|c\|}\hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & -1/6 & 1/3 \\ \hline \end{array}$	$R_{15} = \frac{1}{3}(f(x-1,y)+f(x,y)+f(x+1,y+1)) - \frac{1}{6}\times(f(x+1,y-1)+f(x-1,y-1)+f(x-1,y+1)+f(x,y-1)+f(x+1,y)+f(x,y+1))$
T_{16} :	$\begin{array}{\|c\|c\|c\|}\hline 1/3 & -1/6 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline -1/6 & 1/3 & -1/6 \\ \hline \end{array}$	$R_{16} = \frac{1}{3}(f(x-1,y-1)+f(x,y)+f(x+1,y)) - \frac{1}{6}\times(f(x+1,y+1)+f(x-1,y)+f(x-1,y+1)+f(x,y-1)+f(x+1,y-1)+f(x,y+1))$

After calculating strength of all patterns for a point, we can choice the pattern with the largest strength or *h*-best match, the first *h* large strength, patterns as input type. Because we maybe get the same strength in different pattern and can't decide which pattern type will be better for this point in the initial state. If we use *h*-best match pattern type as input type, the strength of the best pattern type will be increased and other pattern will be decreased at adjusting stage. Furthermore, we choice *h*-best match in our approach.

For example, Figure 2.5(a) shows a line example. The response of point a and b are shown in Figure 2.5(b) and (c). If the *h* is equal to 7, then we choice the first seven large responses and corresponding pattern type as initial input.



(a)

	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}	T_{16}
R_i	0	0.5	0.5	1	0.5	0	0	0.5	0	0	0.5	0	0.5	0	0	0

(b)

	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}	T_{16}
R_i	-0.33	0.17	0.17	-0.33	-0.33	0.67	0.17	-0.33	0.17	0.17	-0.33	-0.33	-0.33	0.17	-0.33	-0.33

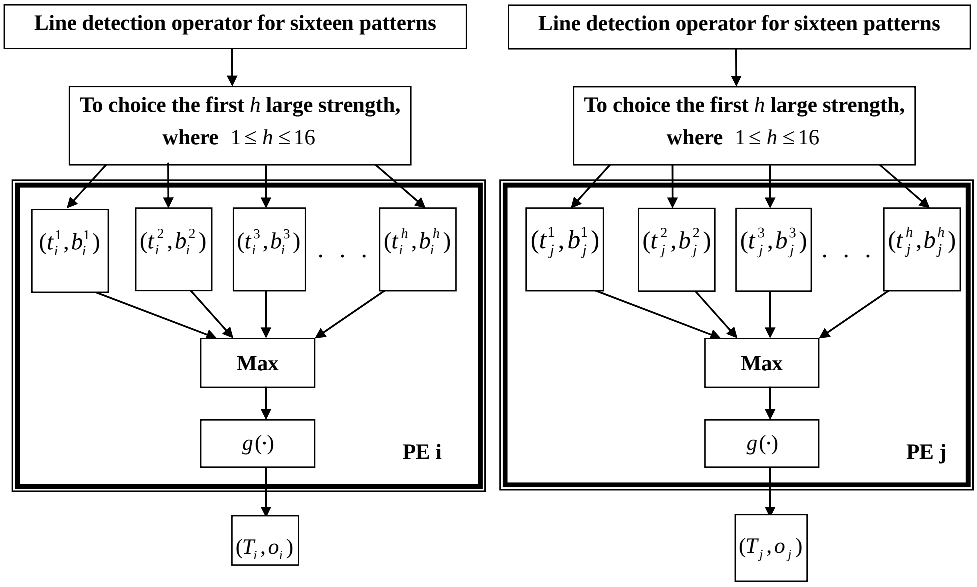
(c)

Figure 2.5.(a)A line example, (b)Response of point a, (c) Response of point b.

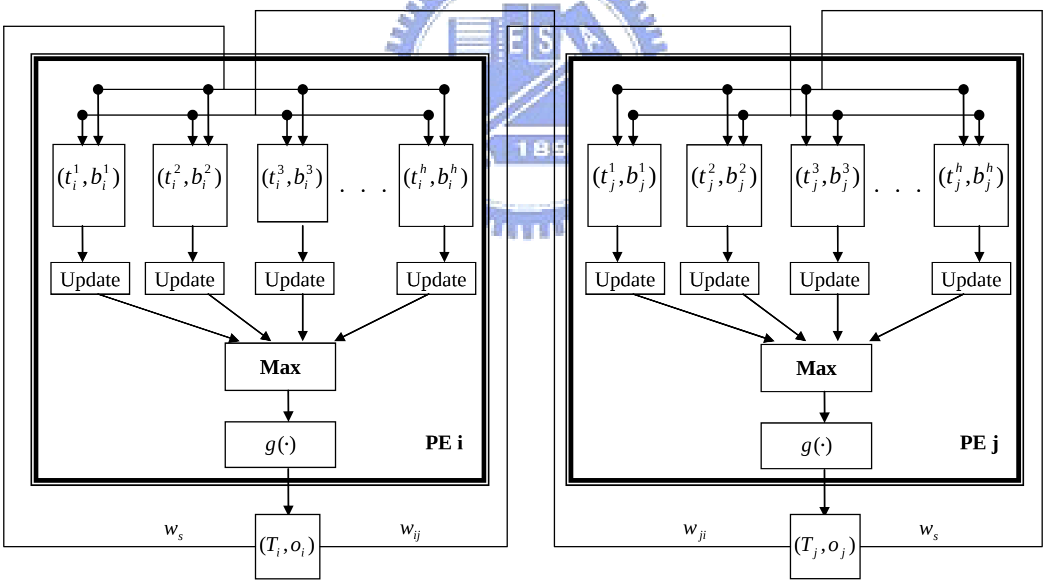
The boundary PEs which without eight neighbors, the gray value of nonexistent neighbors is set to 0.

2.2.2 **Network Architecture**

For an $m \times n$ image, it denotes there are $m \times n$ PEs in the network. Each PE connects with its eight neighboring PEs and a self-feedback. They operate individually in iteration. The Figure 2.6(a) shows the initial state of two PEs. Each PE has *h* states, where $1 \leq h \leq 16$. We calculate response of every PE for sixteen patterns and choices best h-match, the first h large strength, as the initial input. Figure 2.6(b) shows the adjusted state. The update block shows in Figure 2.6(b) is the update step which we discuss later. The response calculation is only used in initial state. When update process beginning, the initial input does not affect the states of the processing element anymore.



(a)



(b)

Figure 2.6. (a) Initial input (b) Adjusted state. (Basak [4])

The state of PE i can be written as

$$(b_i^1\ t_i^1), (b_i^2\ t_i^2), \ldots, (b_i^h\ t_i^h)$$

where $b_i^1, b_i^2, \ldots, b_i^h$ are the strength of each state of PE i and $t_i^1, t_i^2, \ldots, t_i^h$ are the type of each state of PE i.

The output of PE i can be written as

$$(T_i, o_i)$$

where o_i denote the output strength of processing element i (PE i) and T_i is the output type of PE i. The transfer function maps the states to the output. We can write as

$$T_i = t_i^r \ when \ b_i^r \geq b_i^k \ \forall k \quad 1 \leq k \leq h$$

$$o_i = g(\max_{1 \leq k \leq h}(b_i^k))$$

Where h is the number of state in PE i.

The state strength of a point represents the similar degree to a line segment. In the previous section, we discuss the line detection operator and fine out that the response is equal to 1 if a point is fully match one of the sixteen pattern types. Hence, the output strength tend to 1 if the state strength greater than 1 and tend to -1 if state strength less than 1. So the transfer function $g(\cdot)$ is defined as

$$g(x) = \frac{1}{2}(|x+1| - |x-1|)$$

The Figure 2.7 shows the transfer function.

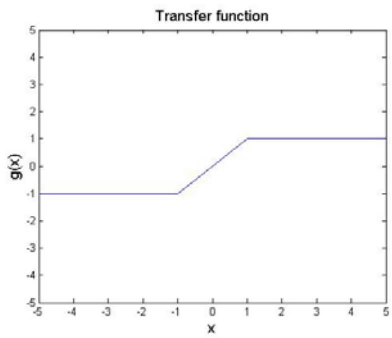


Figure 2.7. Transfer function.

A. Update Formula

Basak propose a update formula to update the strength of state. The strength of the state k of PE i is updated as

$$b_i^k(t+1) = b_i^k(t) + \sum_{j \in N(i)} w_{ji} \Delta_{ji}^k o_j(t)$$

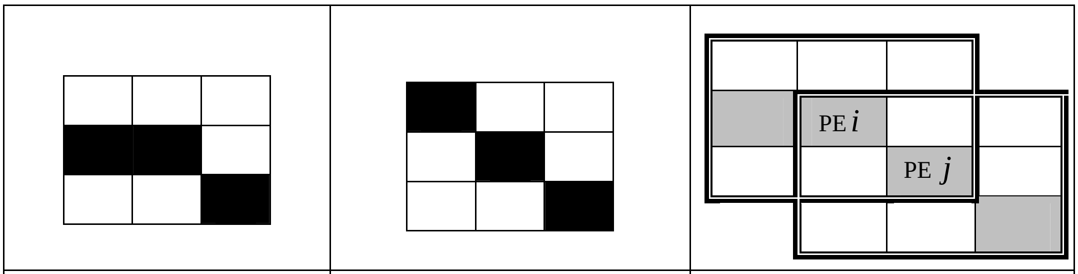
where $N(i)$ represent the neighbors of PE i, w_{ji} is the connection weight of the link from PE j to PE i. It is assume that all interconnection in the network are symmetric, namely $w_{ij} = w_{ji}$.

Δ^k_{ji} is either 0 or 1 which determines whether the kth state of PE i would receive activation from PE j or not. We consider two conditions. First condition, we consider the type of the kth state of PE i. If the part of kth state type of PE i is match the part of output type of PE j, then consider second condition. In the second condition, we consider the position problem. Two types can match a line only when two PEs have some position relationship, so position must be considered. If the kth state type of PE i and the output type of PE j corresponding to two conditions, the Δ^k_{ji} set to 1, else set to 0.

1	2	3
4	PE i	5
6	7	8

Figure 2.8. Neighboring position number of PE i.

Figure 2.8 defines the neighboring position number of the PE i. Suppose that the kth state type of PE i is T_5 shown in Figure 2.9(a), and the output type of PE j is T_1 shown in Figure 2.9(b). These two types can match a line when only PE j in position 8, so Δ^k_{ji} set to 1 only when PE j in position 8 and set to 0 when PE j in other position. Figure 2.9(c) shows the linking condition when PE j in position 8.



(a)	(b)	(c)

Figure 2.9. (a) The kth state type of PE i, T_5, (b) The output type of PE j, T_1, (c) Linking condition when PE j in position 8.

All association conditions of sixteen types are listed in Table 2.2. The pth row and the qth column in Table 2.2 can be written as $M(p,q)=l$. It represents that when the kth state type of PE i is T_p, the output type of PE j is T_q and PE j locates in the position l, we set Δ_{ji}^{k} to 1 else set to 0. From previous example, PE j is at the position 8 of PE i, the kth state type of PE i is T_5, the output type of PE j is T_1, so $M(5,1)$ is equal to 8.

TABLE 2.2

States relation $M(p,q)=l$ between PE i and PE j. (Basak [4])

p \ q	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1,8				1		8	8	1	8	1				1	8
2		4,5		5	5	4	4									
3			3,6	6		3		6	3	3	6		6	3		
4		4	3			3,4	4		3	3				3		
5	8	4				4	4,8	8		8						8
6		5	6	5,6	5			6			6		6			
7	1	5		5	1,5				1		1				1	
8	1		3		1	3			1,3	3	1			3	1	
9	8		6	6			8	6,8		8	6		6			8
10	1		6	6	1			6	1		1,6		6		1	
11	8		3			3	8	8	3	3,8				3		8
12												2,7	2	7	7	2
13			3			3			3	3		7		3,7	7	
14			6	6				6			6	2	2,6			2
15	8						8	8		8		2	2			2,8
16	1				1				1		1	7		7	1,7	

B. Remove Noise

Some points which without receiving any activation from its eight neighbors at update process are regarded as noise. In the previous updated formula, we can observe that the output of the noise is unchanged at update stage. In order to remove these noise points, we add a negative self-feedback to remove them. The updated formula can be rewritten as

$$b_i^k(t+1) = b_i^k(t) + \sum_{j \in N(i)} w_{ji} \Delta_{ji}^k o_j(t) - w_s\left(1 - o_i(t)\right)$$

where w_s is a constant value. It represents the strength of self-feedback which is the same for all PEs in the network. Usually, a pixel on a line, the activation get form its neighboring PEs is greater than the self-feedback. Due to the output value of a PE can express the similar degree to some type. If the output value of a PE is close to 1, we can say that the PE with its eight neighbors is similar to some line segment of sixteen types. If the output value of a PE is close to 0, we can say that the PE similar to noise. Hence, we let the negative self-feedback larger when the output value closed to 0 than closed to 1.

C. Converge of Network

In this section, we prove the convergence and stability of the network. It is sufficient to prove that both output type and output value converge and stable. First, we prove the stability of output type.

Proof:

For any two neighboring PE i and j, if $T_i = t_i^k$ and $T_j = t_j^s$, then $\Delta_{ji}^k = 1 \Leftrightarrow \Delta_{ij}^s = 1$. This represents that if the PE i supports the output type of PE j, then the PE j must support the output type of PE i.

First, we consider $h = 2$. Hence, for PE i, the state can be written as

$(t_i^1, b_i^1), (\mathrm{t}_\mathrm{i}^2, \mathrm{b}_\mathrm{i}^2)$, and let the output be (T_i, o_i). Suppose the initial state is $T_i = t_i^1$, i.e. $b_i^1 > b_i^2$ and $o_i = g(b_i^1)$. T_i changes to t_i^2 only if b_i^2 becomes greater than b_i^1. If T_i changes to t_i^2, that means that the state element (t_i^2, b_i^2) get more support than the state element (t_i^1, b_i^1). On the other hand, state element (t_i^1, b_i^1) would not receive support from its neighboring PEs. In this situation, the support to the state element (t_i^2, b_i^2) would increase further. This ensures that b_i^2 increase recursively and T_i remains the same as t_i^2. Hence there will be at most one interchange between these two state elements. For $h > 2$, there can be at most $h-1$ change in the output type. So the output type of all PEs will stabilize depending on the neighboring information.

Next, we prove the output value of network will stabile.

Proof:

We define the energy function $E(t)$ as

$$E(t) = -\frac{1}{2}\sum_i \sum_{j \in N(i)} w_{ji}\Delta_{ji} o_j(t) o_i(t) + \frac{1}{2}\sum_i w_s (1 - o_i(t)) o_i(\mathrm{t})$$

$$\text{where} \quad \Delta_{ji} = \begin{cases} 1 & \text{if } \Delta_{ji}^k = 1 \text{ and } T_i = t_i^k \\ 0 & \text{otherwise} \end{cases}$$

In previous section, the output of PE i can be written as

$$o_i = g(\max_{1 \le k \le h}(b_i^k))$$

The output of the PE i is the maximum strength of k state elements. Because the output type of all PEs are stable. Hence, the output strength will be the corresponding strength of the stable output type. Therefore, when the output type is stable, the output strength can be written as

$$o_i = g(b_i)$$

where b_i represents the strength of the mostly activated state element of PE i. Form the update formula, the output activation can be written as

$$\frac{db_i}{dt} = \sum_{j \in N(i)} w_{ji} \Delta_{ji} o_j - w_s (1 - o_i)$$

where b_i represents the mostly activated state element of PE i, the state element which get the largest strength, i.e. $b_i = g^{-1}(o_i)$.

It is clear that the energy function is bounded for some finite network structure because the value of o_i and o_j are between 0 and 1 and the other parameters are constant. From energy function, the differential equation of energy function can be written as

$$\begin{aligned}
\frac{dE}{dt} &= -\frac{1}{2} \sum_i \left(\sum_{j \in N(i)} w_{ij} D_{ij} o_j - w_s (1 - o_i) \right) \left(\frac{do_i}{dt} \right) \\
&= -\frac{1}{2} \sum_i \frac{db_i}{dt} \left(\frac{do_i}{dt} \right) \\
&= -\frac{1}{2} \sum_i \frac{dg^{-1}(o_i)}{dt} \left(\frac{do_i}{dt} \right) \\
&= -\frac{1}{2} \sum_i g^{-1'}(o_i) \left(\frac{do_i}{dt} \right)^2
\end{aligned}$$

where $g^{-1'}(.)$ is the first derivative of the inverse of $g(.)$, which is an increasing function (we define $g^{-1'}(x) = \infty$ when $|x| = 1$). Hence, $g^{-1'}(.)$ is nonnegative.

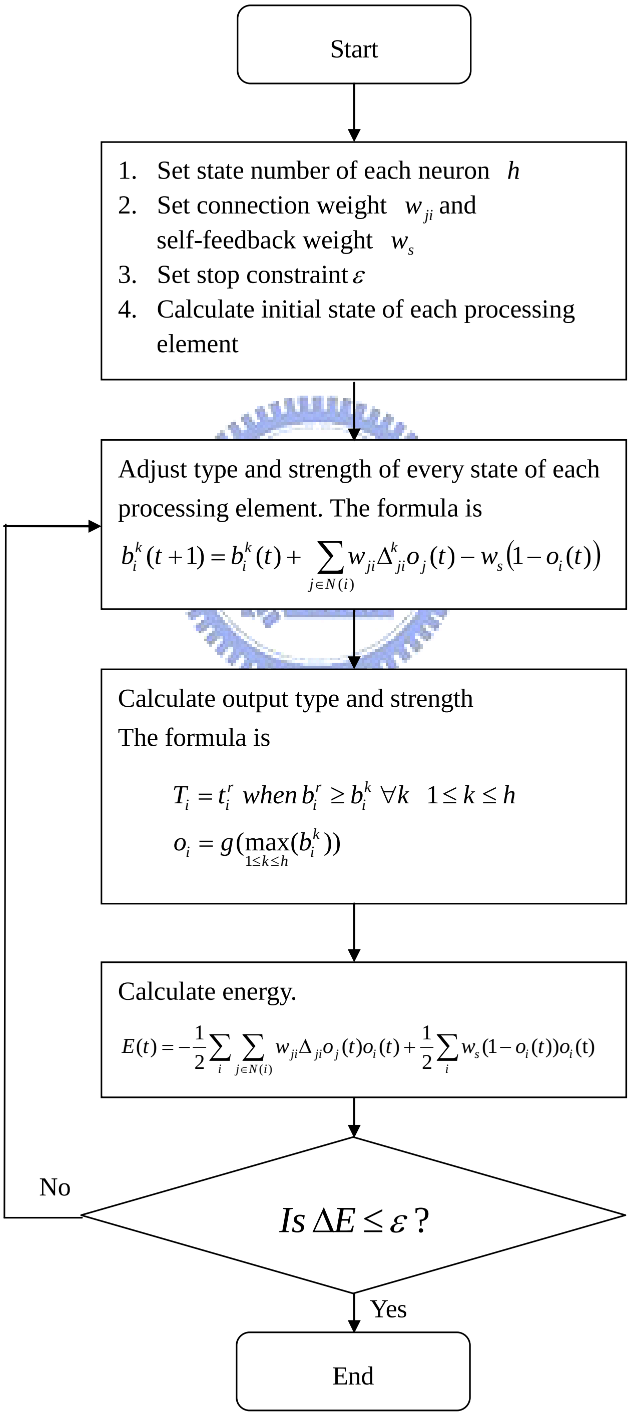
The $\left(\frac{do_i}{dt} \right)^2$ is always positive, so we can get

$$\frac{dE}{dt} \leq 0 \quad \forall t > 0$$

Because $E(t)$ is bounded and is a decreasing function, i.e. $dE/dt \to 0$ as $t \to \infty$ and therefore $do_i/dt \to 0$ as $t \to \infty$. Hence, the output values of all PEs converge.

We calculate the energy after an adjustment. When energy is not changed, the network achieves equilibrium, and then stops the update process and output the linking result. Because the value of output strength is between 0 and 1, we set the gray value to 1if the output strength is greater than 0.5 and set the gray value to 0 if the output strength is less than 0.5.

2.3 Programming Flowchart



2.4 Examples

Next, we shows an 5×5 images to explain. We set the parameters $w_{ji} = 1 \forall i, j$ and $w_s = 1$ in this example.

The input image is shown in Figure 2.10

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

Figure 2.10. Input image.

1. Set the number of state of a processing element $h = 3$ and stop constraint $\varepsilon = 0$
2. Calculate initial state of every processing element. Figure 2.11 shows the initial state of input data. A thick border box is a PE. There are four states and an output in a PE.

   For example, we consider the center point of input image. We use line detection operator to calculate initial state. The result of calculation shows in Table 2.3. In this example, we choice $h = 3$, and then the first three largest strength will be chosen as initial state of considered point.

TABLE 2.3

The result of calculation by line detection operator.

Type	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}	T_{15}	T_{16}
strength	-0.33	-0.33	0.67	0.17	-0.33	0.17	-0.33	0.17	0.17	0.17	0.17	-0.33	0.17	0.17	-0.33	-0.33

The maximum strength and corresponding type of initial state is the initial output. We get output strength by transfer function.

$$g(0.67) = 0.67$$

So the output strength is 0.67 and output type is T_3.

h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	0	0	0	0	0	0	0.33	0.33	0.33	0.17	0.17	0.17	0.17	0.17	0.17
Output strength	0			0			0.33			0.17			0.17		
Output type	T_1			T_1			T_1			T_1			T_3		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	0	0	0	0	0	0	0.33	0.33	0.33	0.67	0.67	0.67	0.67	0.67	0.67
Output strength	0			0			0.33			0.67			0.67		
Output type	T_1			T_1			T_2			T_2			T_2		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	0.17	0.17	0.17	0.17	0.17	0.17	0.67	0.17	0.17	0.17	0.17	0.17	0.17	0.17	0.17
Output strength	0.17			0.17			0.67			0.17			0.17		
Output type	T_1			T_3			T_3			T_3			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	0.67	0.67	0.67	0.67	0.67	0.67	0.33	0.33	0.33	0	0	0	0	0	0
Output strength	0.67			0.67			0.33			0			0		
Output type	T_2			T_2			T_2			T_1			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	0.17	0.17	0.17	0.17	0.17	0.17	0.33	0.33	0.33	0	0	0	0	0	0
Output strength	0.17			0.17			0.33			0			0		
Output type	T_3			T_1			T_1			T_1			T_1		

Figure 2.11. Initial state.

3. Fed initial state to the network to adjust

   Iteration 1:

   The result of first update is shown in figure 2.12.

h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	-1	-1	-1	-1	-1	-1	-0.33	-0.33	-0.33	-0.67	-0.67	-0.67	-0.67	-0.67	-0.67
Output strength	-1			-1			-0.33			-0.67			-0.67		
Output type	T_1			T_1			T_1			T_1			T_3		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	-1	-1	-1	-2	-0.67	-1	-0.33	0.5	-0.67	1.5	1.67	0.67	1.33	1.33	1.33
Output strength	-1			-0.67			0.5			1			1		
Output type	T_1			T_2			T_6			T_6			T_2		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	-0.67	-0.67	-0.67	-0.17	-0.67	-0.67	1.33	0.83	-0.17	-0.67	-0.67	-0.67	-0.67	-0.67	-0.67
Output strength	-0.67			-0.17			1			-0.67			-0.67		
Output type	T_1			T_3			T_3			T_3			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	1	1	1	1.67	2.33	1.67	-0.33	-1	-0.33	-1	-1.33	-1	-1	-1	-1
Output strength	1			1			-0.33			-1			-1		
Output type	T_2			T_4			T_2			T_1			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	-0.67	-0.67	-0.67	-0.67	-0.67	-0.67	-0.33	-0.33	-0.33	-1	-1	-1	-2	-1	-1
Output strength	-0.67			-0.67			-0.33			-1			-1		
Output type	T_3			T_1			T_1			T_1			T_2		

Figure 2.12. First update result.

Energy $E(1) = -0.625$

Iteration 2:

The result of second update is shown in figure 2.13.

h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	-3	-3	-3	-3	-3	-3	-1.67	-1.67	-1.67	-2.33	-2.33	-2.33	-2.33	-2.33	-2.33
Output strength	-1			-1			-1			-1			-1		
Output type	T_1			T_1			T_1			T_1			T_3		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	-3	-3.67	-3	-4.67	-2.33	-2.67	-1.83	-0.17	-2.17	2.33	3.67	0.67	2.33	2.33	2.33
Output strength	-1			-1			-0.17			1			1		
Output type	T_1			T_2			T_6			T_6			T_2		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	-2.33	-2.33	-2.33	-1.5	-0.83	-1.83	3.33	1.83	0.83	-2.33	-2.33	-2.33	-2.33	-2.33	-2.33
Output strength	-1			-1			1			-1			-1		
Output type	T_1			T_6			T_3			T_3			T_1		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	2	2	2	2.33	4.33	2.33	-1.67	-3.33	-2.67	-3	-4.33	-3	-3	-3	-3
Output strength	1			1			-1			-1			-1		
Output type	T_2			T_4			T_2			T_1			T_1		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	-2.33	-2.33	-2.33	-2.33	-2.33	-2.33	-1.67	-1.67	-1.67	-3	-4	-3	-5	-3	-3
Output strength	-1			-1			-1			-1			-1		
Output type	T_3			T_1			T_1			T_1			T_2		

Figure 2.13. Second update result.

Energy $E(2) = \text{-}16.056$

Calculate $\Delta E = -0.625 - (-16.056) = 15.43 > 0$, adjust.

Iteration 3:

The result of update is shown in figure 2.14.

h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	-5	-5	-5	-5	-5	-5	-3.67	-3.67	-3.67	-4.33	-4.33	-4.33	-4.33	-4.33	-4.33
Output strength	-1			-1			-1			-1			-1		
Output type	T_1			T_1			T_1			T_1			T_3		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	-5	-6.67	-5	-7.67	-4.33	-4.67	-4	-2.33	-4.33	2.33	5.67	0.67	3.33	3.33	3.33
Output strength	-1			-1			-1			1			1		
Output type	T_1			T_2			T_6			T_6			T_2		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	-4.33	-4.33	-4.33	-4.5	-.383	-3.83	5.33	1.83	1.83	-4.33	-4.33	-4.33	-4.33	-4.33	-4.33
Output strength	-1			-1			1			-1			-1		
Output type	T_1			T_6			T_3			T_3			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	3	3	3	2.33	6.33	2.33	-3.67	-6.33	-5.67	-5	-7.33	-5	-5	-5	-5
Output strength	1			1			-1			-1			-1		
Output type	T_2			T_4			T_2			T_1			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	-4.33	-4.33	-4.33	-4.33	-4.33	-4.33	-3.67	-3.67	-3.67	-5	-7	-5	-8	-5	-5
Output strength	-1			-1			-1			-1			-1		
Output type	T_3			T_1			T_1			T_1			T_2		

Figure 2.14. Third update result.

Energy $E(3) = -23.181$

Calculate $\Delta E = -16.056 - (-23.181) = 7.125 > 0$, adjust.

Iteration 4:

The result of update is shown in figure 2.15.

h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	-7	-7	-7	-7	-7	-7	-5.67	-5.67	-5.67	-6.33	-6.33	-6.33	-6.33	-6.33	-6.33
Output strength	-1			-1			-1			-1			-1		
Output type	T_1			T_1			T_1			T_1			T_3		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	-7	-9.67	-7	-10.7	-6.33	-6.67	-7	-4.33	-7.33	2.33	7.67	0.67	4.33	4.33	4.33
Output strength	-1			-1			-1			1			1		
Output type	T_1			T_2			T_6			T_6			T_2		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	-6.33	-6.33	-6.33	-7.5	-5.83	-5.83	7.33	1.83	2.83	-6.33	-6.33	-6.33	-6.33	-6.33	-6.33
Output strength	-1			-1			1			-1			-1		
Output type	T_1			T_6			T_3			T_3			T_1		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	4	4	4	2.33	8.33	2.33	-5.67	-9.33	-8.67	-7	-10.3	-7	-7	-7	-7
Output strength	1			1			-1			-1			-1		
Output type	T_2			T_4			T_2			T_1			T_1		
h	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**	**1**	**2**	**3**
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	-6.66	-6.33	-6.33	-6.66	-6.33	-6.33	-5.67	-5.67	-5.67	-7	-10	-7	-11	-7	-7
Output strength	-1			-1			-1			-1			-1		
Output type	T_3			T_1			T_1			T_1			T_2		

Figure 2.15. Forth update result.

Energy $E(4) = -24$

Calculate $\Delta E = -23.181 - (-24) = 0.8194 > 0$, adjust.

Iteration 5:

The result of update is shown in figure 2.16.

h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_5	T_9	T_1	T_5	T_9	T_3	T_6	T_9
State strength	-9	-9	-9	-9	-9	-9	-7.67	-7.67	-7.67	-8.33	-8.33	-8.33	-8.33	-8.33	-8.33
Output strength	-1			-1			-1			-1			-1		
Output type	T_1			T_1			T_1			T_1			T_3		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_2	T_3	T_1	T_2	T_3	T_2	T_6	T_7	T_2	T_6	T_7	T_2	T_4	T_5
State strength	-9	-12.7	-9	-13.7	-8.3	-8.67	-10	-6.33	-10.3	2.33	9.67	0.67	5.33	5.33	5.33
Output strength	-1			-1			-1			1			1		
Output type	T_1			T_2			T_6			T_6			T_2		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_1	T_5	T_9	T_3	T_6	T_9	T_3	T_4	T_6	T_3	T_4	T_8	T_1	T_7	T_8
State strength	-8.33	-8.33	-8.33	-10.5	-7.83	-7.83	9.33	1.83	3.83	-8.33	-8.33	-8.33	-8.33	-8.33	-8.33
Output strength	-1			-1			1			-1			-1		
Output type	T_1			T_6			T_3			T_3			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_2	T_6	T_7	T_2	T_4	T_5	T_2	T_4	T_5	T_1	T_2	T_3	T_1	T_2	T_3
State strength	5	5	5	2.33	10.33	2.33	-7.67	-12.3	-11.7	-9	-13.3	-9	-9	-9	-9
Output strength	1			1			-1			-1			-1		
Output type	T_2			T_4			T_2			T_1			T_1		
h	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
State type	T_3	T_4	T_8	T_1	T_7	T_8	T_1	T_7	T_8	T_1	T_2	T_3	T_1	T_2	T_3
State strength	-8.33	-8.33	-8.33	-8.33	-8.33	-8.33	-7.67	-7.67	-7.67	19	-13	-9	-14	-9	-9
Output strength	-1			-1			-1			-1			-1		
Output type	T_3			T_1			T_1			T_1			T_2		

Figure 2.16. Fifth update result.

Energy $E(5) = -24$

Calculate $\Delta E = -24 - (-24) = 0 \leq 0$, stop adjust.

4. Output image

Figure 2.17 shows the output result and Figure 2.18 shows the energy curve.

-1	-1	-1	-1	-1
-1	-1	-1	1	1
-1	-1	1	-1	-1
1	1	-1	-1	-1
-1	-1	-1	-1	-1

Figure 2.17.Output result.

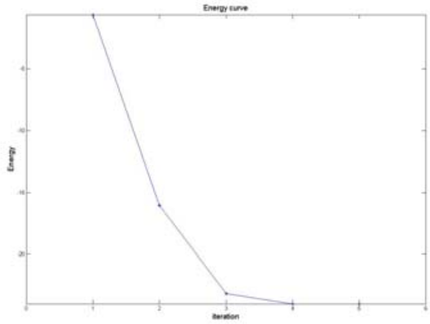


Figure 2.18.Energy curve.

2.5 Experiments

Experiment I：Circle

In experiment I, we use a circle image to do line linking experiment which shows in Figure 2.19. The size of image is 50×50. We set the parameters as $w_{ij} = 1 \, \forall i, j$, $w_s = 1$, state number of each PE $h = 16$ and stop constraint $\varepsilon = 0$ in this experiment.

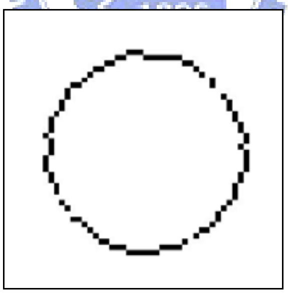
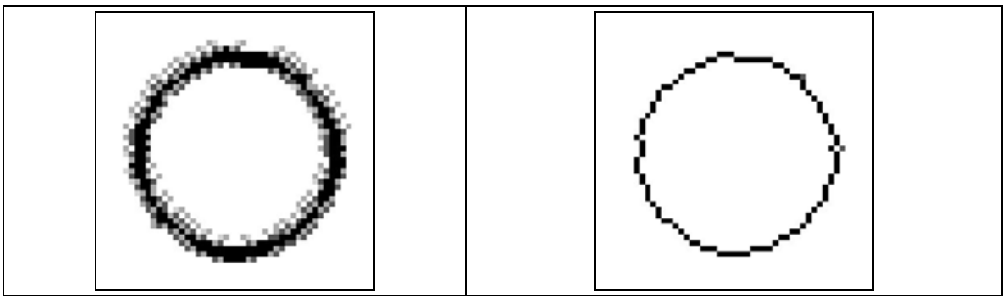


Figure 2.19.Input data.

Four iterations have run in this experiment. Figure 2.20 show output image of iteration. Table 2.4 shows energy of iteration. The energy curve is shown in Figure 2.21.



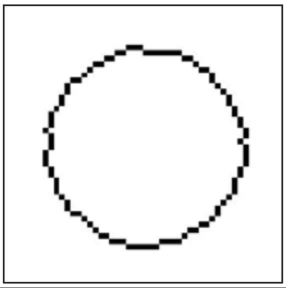
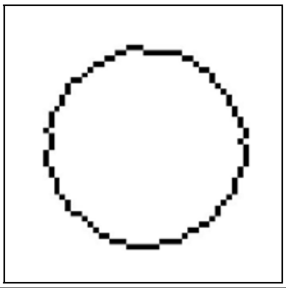
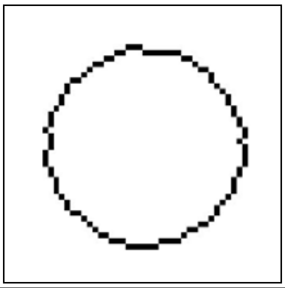
Iteration 1	Iteration 2
	
Iteration 3	Iteration 4
	
Iteration 5	

Figure 2.20. Output images of each iteration.

TABLE 2.4
Energy of iteration

Iteration	1	2	3	4	5
Energy	-61.375	-2251.6	-2495.3	-2500	-2500

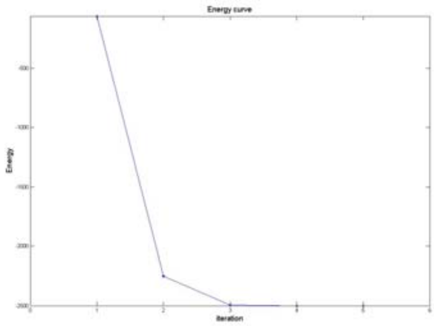


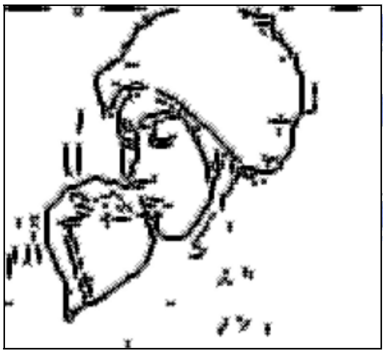
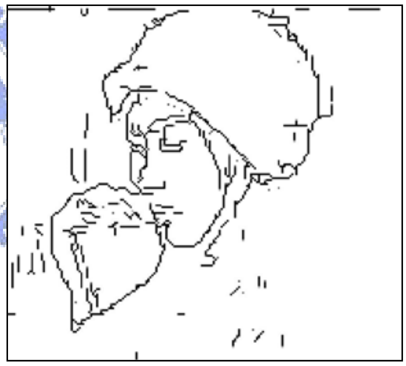
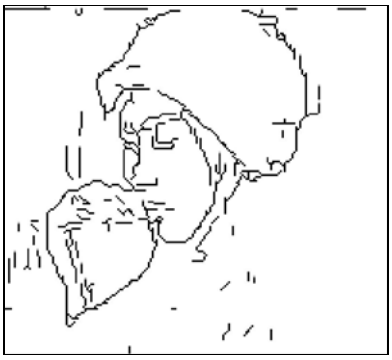
Figure 2.21.Energy curve of experiment I.

Experiment II：Personal face

In experiment II, we use a face image to do line linking experiment which shows in Figure 2.22. The size of image is 200×182. We set the parameters as $w_{ij} = 1 \, \forall i, j$, $w_s = 1$, state number of each PE $h = 16$ and stop constraint $\varepsilon = 0$ in this experiment.

Figure 2.22.Input data.

Ten iterations have run in this experiment. Figure 2.23 show output image of iteration. Table 2.5 shows energy of iteration. The energy curve is shown in Figure 2.24.

	
Iteration 1	Iteration 2
	
Iteration 3	Iteration 4

	
Iteration 5	Iteration 6
	
Iteration 7	Iteration 8

Figure 2.23. Output images of each iteration.

TABLE 2.5
Energy of each iteration

Iteration	1	2	2	4	5	6	7	8
Energy	-609.1	-33033	-36010	-36059	-36086	-36088	-36089	-36089

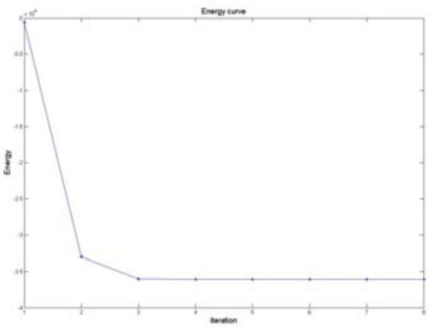


Figure 2.24.Energy curve of experiment II.