An Unsupervised Learning Algorithm For Character Recognition

C.K. Lee, P.F. Sum, K.S.Tam

Dept. of Electronic Eng.

Hong Kong Polytechnic.

Abstract

An unsupervised learning algorithm is presented for the character recognition. Basically the idea is based on electrostatics phenomenon. Though the algorithm is very simple, it illustrates the application of a law of Physics for learning. Here, we treat the normalized pattern vectors as the position vectors of positive charges, while the normalized weight vectors as the position vectors of negative charges. The positive charges are fixed in location, and we let the negative charges to move freely in the space. A step of movement of the negative charge indicates the change of the corresponding weight vector. Hence, according to the Inverse Square Law, $F = k/r^2$, where r is the distance between two charges, we can evaluate each of the forces acting on a single charge. The direction of the change of a weight vector is along the direction of the resultant force. For simplicity, we make the movement of the charge as $\Delta\mathbf{W} = \mu\mathbf{F}$. It is found that this network can be self organized to give various responses to different input patterns. Compared with other competitive learning algorithms, this algorithm can lift the limitation due to different initial settings.

1. Introduction

The develpment of unsupervised learning algorithm has been long in history[1-4]. It can be traced back to 1958, the Gamma Perceptron[1]. Competitive learning algorithm is one of the simplest paradigm in this area. Though competitive learning can be applied in clustering and categorization of patterns, its application in pattern recognition is very poor, as mentioned in [5]. It is mainly due to the inital setting constraint. In the last decade, Stephen Grossberg reclaimed the Adaptive Resonance Theory and he also proved ART can eliminate this limitation[5]. Grossberg and his colleague implemented this model in adaptive pattern recognition[6]. However, in this paper, we are not using the approach of ART, instead we propose a new one. First, we review the limitation of competitive learning. Secondly, for completeness, we state the criteria for the unsupersived learning algorithm in pattern/character recognition. Thirdly, we elucidate the unsupervised algorithm that we have investigated. Simulation results are included to illustrate the success of this approach.

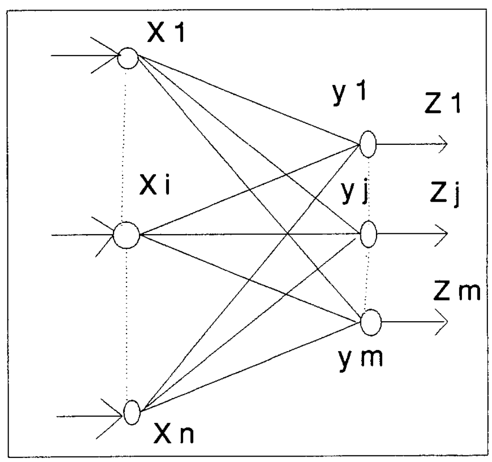

Fig. 1 Simplified competitive learning model

2. Competitive learning

For simplicity, we define a neural network as a two layers of neurons, Fig. 1. Each of the neurons in the first layer is connected to all the neurons in the second layer. Response of the first layer is in an *all-or-none* principle. That is, when the neuron *i* in the first layer receives signal greater than a threshold, this neuron will deliver an impulse, $x_i = 1$. All impulses from the first layer will then pass to the second layer through the synaptic connections, and the strength of the synapse is denoted by w_{ji} where *j* represents the *j*th neuron in the second layer.

The response of the second layer obeys the *winner-takes-all* rule. Hence, there is only one neuron will be active. Once a neuron wins, say neuron *j*, its corresponding weight will be updated as Equ. (1)[4]. If there are *m* input patterns, $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_m$, we have

$$\Delta w_{ji} = g\frac{C_{ik}}{n_k} - gw_{ji} \tag{1}$$

where k = 1, 2, ..., m. C_{ik} and n_k are constants such that the summation of the C_{ik}/n_k is equal to 1. The parameter g is the step size. Since the norm of W_j ($|W_j| = \sum_{i=1}^{n} w_{ji}$) is equal to unity, the change of W in each step is zero. Accordingly, the weight vector is moving towards the pattern vector, Fig 2. Qualitatively speaking, we can state the principle of competitive learning as the following:

The winner will be given to the one which is the closest to the pattern, otherwise it is a loser. For a winner, it will be attracted towards the pattern; otherwise, as a loser, it will remain unchanged at all.

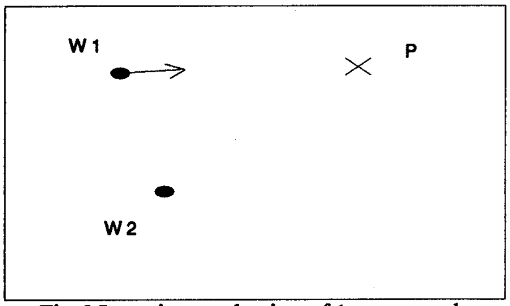

Fig. 2 Learning mechanism of 1 pattern only

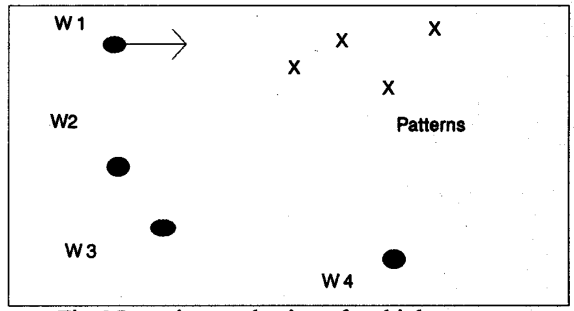

Fig. 3 Learning mechanism of multiple patterns

In this case, it is observed that the movement of the neurons is highly determined by the initial values of both the weight vector and the pattern vector. In Fig 3, we have 4 output neurons and 4 input patterns. If, initially there is one weight vector, say W_1, very close to all the 4 pattern vectors, it will win all the four patterns. Therefore, it will be the only weight vector getting updated. As a result, the system cannot give out 4 different responses to the four patterns. (In fact, this is the idea of clustering.) Hence, it is the limitation that competitive learning algorithm cannot be implemented for pattern recognition.

3. Criteria of unsupervised learning for pattern recognition

The limitation of the competitive learning algorithm is briefly described as above. In order to make a description of this learning model more complete, we here define the criteria of an unsupervised learning algorithm in pattern recognition as:

No matter which algorithm the neural network is used, the network must achieve (i) an equlibrium state, (ii) the equlibrium state must be stable and (iii) the output response must be observable.

Criteria (i) and (ii), in fact, have been considered as the basic conditions for checking the network performance. Criterion (iii) is actually a common understanding criterion in pattern recognition. In mathematical term, it means that: For each of the input patterns, there must exist one neuron in the second layer for the response. Besides, for each of the output neurons, it can only respond to one input pattern. That is, the mapping is one-to-one. These three conditions govern the way for us to derive an unsupervised learning algorithm.

4. Learning model

In principle, the structure of this learning model is the same as Fig. 1. The only difference is the weight updating rule.

Organization

1. The pattern **P** is impinged onto the first layer of neurons which are of response in an *all-or-none* basis. The output of the first layer is denoted by $X = \{x_1, .., x_i, .., x_n\}$, and is given by Equ. (2), where θ_i is the threshold of the *i*th neuron.

$$x_i = \begin{cases} 1 & \text{if } p_i \geq \theta_i \\ 0 & \textit{otherwise} \end{cases} \quad (2)$$

2. When the signal is delivered from neuron *i* in the first layer to the neuron *j* in the second layer, the value is decreased. The value received is $w_{ji}x_i$. Hence the total signal received by *j*th neuron, y_j, in the second layer is given by Equ. (3).

$$y_j = \sum_{i=1}^{n} w_{ji} x_i \qquad (3)$$

3. The response of the second layer neurons will be in a *winner-takes-all* basis, and it is given as Equ. (4).

$$z_j = \begin{cases} 1 & \text{if } y_j \text{ is a maximum} \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

Self Organization As Charge Distribution

The idea of the learning, in fact, comes from the phenomenon of electric charge distribution,[7]. Suppose the normalized pattern vectors to be the position vector of positive charges, and the nomalized weight vector to be the position vector of negative charges. As the patterns are fixed, it means that the position of positive charges will also be fixed. Hence, only the negative charges will move freely. On another point of view, you can imagine that there are only weight vectors moving in the space. Moreover, the motion is governed by the resultant external force acting on it at that position. To evaluate the change of the weight vector at that position, we simply find out the total force acting on that charge. Then we make the change of the weight vectors along the direction of the resultant force, i.e. change of weight is proportional to the net external force. Fig 4 shows the above idea.

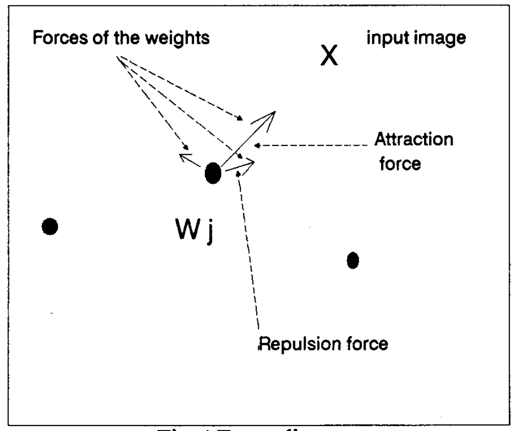

Fig. 4 Force diagram

In Fig.4, there are three negative charges in the space which positions are given by $\mathbf{W}_j$ where j = 1,2,3. There is a positive charge , the cross sign, which position is given by the pattern vector. Consider the neuron *j*, there are three external forces acting on it, indicated by three arrows. The arrow pointing towards the input image is the attraction force while the other two are repulsion force. According to the Inverse Square Law, the attraction force is given by Equ. (4).

$$\mathbf{F}_A = q_1 \frac{\mathbf{X} - \mathbf{W}_j}{||\mathbf{X} - \mathbf{W}_j||^3} \qquad (4)$$

where $q_1 = Q_1Q_2/(4\pi\varepsilon)$, Q_1 and Q_2 are the magnitudes of the positive and negative charges respectively. Here the meaning of $||.||$ is not the same meaning as norm, as stated in section 2, instead, *symbol* $||.||$ *is the Elucidean norm,* i.e the magnitude or the length of the vector. The repulsion force between *j* neuron and the *r* neuron is given by Equ.(5).

$$\mathbf{F}_R = q_2 \sum_r \frac{\mathbf{W}_j - \mathbf{W}_r}{||\mathbf{W}_j - \mathbf{W}_r||^3} \qquad (5)$$

where $q_2 = Q_2Q_2/(4\pi\varepsilon)$, and r denotes the rest of the weight vectors. Hence the resultant force acting on the negative charge, corresponding to $\mathbf{W}_j$ can be given by Equ.(6).

$$\mathbf{F}_j = \mathbf{F}_A + \mathbf{F}_R \qquad (6)$$

Let us have a simple example. Consider the case when there is only one neuron in the second layer and there is only one input pattern. That is to say there is only one negative charge and one positive charge, the negative charge will move towards the positive charge, i.e. the weight vector is equal to the pattern vector. When there are more than one neuron in the second layer and more than one input pattern, the steps of updating is given as below.

Learning Algorithm (Weight Updating Rule)

Step 1. Normalize all the input patterns and on each of the dimensions of the normalized patterns, add a small value, say 0.05, to it.

Step 2. Normalize all the weight vectors, so the magnitude is equal to unity.

Step 3. Select randomly one of the normalized patterns.

Step 4. Calculate the total force acting on each of the negative charges. The resultant force is given by Equ. (6), and let it move along the direction of the resultant force.

Step 5. The movement of charges indicates the change of the weight vectors. For each of the weight vectors, it is updated with a step size, μ, as given by Equ. (7).

$$\Delta \mathbf{W}_j = \mu \mathbf{F}_j \qquad (7)$$

Repeat Step 2-5 for N times, where N is no. of iterations.

Step 6. The weight value is fixed and the network is tested by the Equ. (1) to Equ. (3).

The inclusion of Step 1 is used to prevent the situation when $\mathbf{W}_j = \mathbf{X}$. In step 5, it is clear that the update of weight vector is indenpendent of z_j which is the output of the second layer neurons.

The above algorithm has been simulated by a computer program and the result is compared with the comptitive learning algorithm [4]. The result will be stated in the next section.

5. Simulation

We use a network which consists of 48 neurons in the first layer and 8 neurons in the second layer, Fig 5. We set the positive charge is of 25 units and the negative charge is of 1 unit. Hence $Q_1 = 25$ and $Q_2 = 1$. So the attraction force is larger than the repulsion force. The step size of the weight update is 0.0001. The thresholds of the each of neurons in the first layer are set to 0, i.e., $\theta_i = 0$. The update takes over 4096 iterations. Two sets of patterns, selected from Fig 6, are investigated.

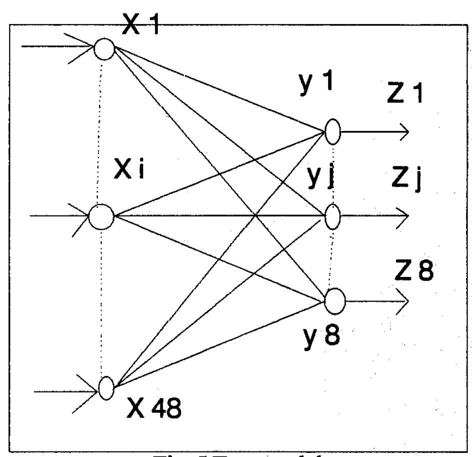

Fig. 5 Test model

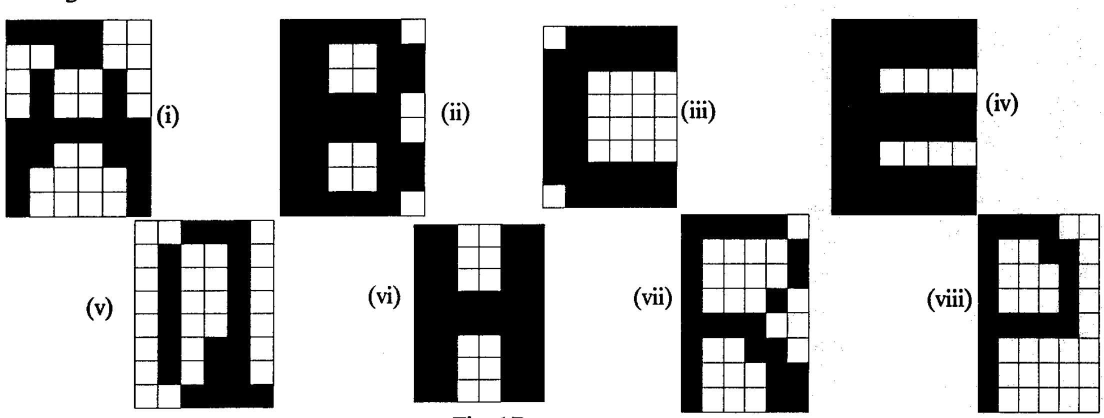

Fig. 6 Pattern vector

Test 1

In the test, we choose pattern (i) to (vi). Initially, the 2nd neuron in the second layer is the winner of pattern 1 to 4. After 512 iterations, the network can distinguish more than 1 pattern. After 4096 iterations, the network can completely differentiate all the 6 patterns.

No. of iterations	Pattern no.	z_j	max. value of y_j
0	Image 1 :	0 1 0 0 0 0 0 0	0.976141
	Image 2 :	0 1 0 0 0 0 0 0	0.955053
	Image 3 :	0 1 0 0 0 0 0 0	1.016840
	Image 4 :	0 1 0 0 0 0 0 0	1.116477
	Image 5 :	1 0 0 0 0 0 0 0	1.023682
	Image 6 :	0 0 0 0 0 0 1 0	1.125428
512	Image 1 :	0 0 0 0 1 0 0 0	1.159309
	Image 2 :	0 1 0 0 0 0 0 0	1.157658
	Image 3 :	0 0 0 1 0 0 0 0	1.146984
	Image 4 :	1 0 0 0 0 0 0 0	1.253260
	Image 5 :	0 0 0 0 0 0 1 0	1.194579
	Image 6 :	0 0 0 0 0 0 1 0	1.290241
4096	Image 1 :	0 0 0 0 1 0 0 0	1.259999
	Image 2 :	0 1 0 0 0 0 0 0	1.241957
	Image 3 :	0 0 0 1 0 0 0 0	1.260759
	Image 4 :	1 0 0 0 0 0 0 0	1.307193
	Image 5 :	0 0 0 0 0 0 1 0	1.271767
	Image 6 :	0 0 0 0 0 0 0 1	1.320562

The same pattern set is input to the competitive learning network[4], where C_{ik} is set to unity, n_k is set to be the no. of the active neurons in the first layer due to the impaining patterns, and g is set to be 0.0001. The following results are obtained. It is observed that the final/intermediate results of the network are totally determined by the initial conditions.

No. of iterations	Pattern no.	z_j	max. value of y_j
0	Image 1 :	0 1 0 0 0 0 0 0	0.022365
	Image 2 :	0 1 0 0 0 0 0 0	0.024384
	Image 3 :	0 0 0 1 0 0 0 0	0.023920
	Image 4 :	1 0 0 0 0 0 0 0	0.022417
	Image 5 :	1 0 0 0 0 0 0 0	0.022807
	Image 6 :	0 0 0 0 0 0 0 1	0.022316
512	Image 1 :	0 1 0 0 0 0 0 0	0.022594
	Image 2 :	0 1 0 0 0 0 0 0	0.024700
	Image 3 :	0 0 0 1 0 0 0 0	0.024065
	Image 4 :	1 0 0 0 0 0 0 0	0.022464
	Image 5 :	1 0 0 0 0 0 0 0	0.022897
	Image 6 :	0 0 0 0 0 0 0 1	0.022340

4096	Image 1 :	0 1 0 0 0 0 0 0	0.024251
	Image 2 :	0 1 0 0 0 0 0 0	0.026696
	Image 3 :	0 0 0 1 0 0 0 0	0.025121
	Image 4 :	1 0 0 0 0 0 0 0	0.022746
	Image 5 :	1 0 0 0 0 0 0 0	0.023439
	Image 6 :	0 0 0 0 0 0 0 1	0.022494

Obviously, the neuron 1 and 2 respond for more than one image pattern all the time.

Test 2 (Chaotic Approach)

In this test, we select the pattern (i) to (iv) and (vii) to (viii). The network parameters are the same except the no. of iterations is reduced to 1024. The result obtained is not as good as that of Test 1. It is found that the network can only give different responses to the first four patterns. It gives the same response to the last two. Scanning through the results, it is sure that the weight vectors have attained stable values. In order to solve this not one-to-one mapping problem, we add a concept - **Chaos** - to the simulation program. After every 128 iterations, the program will randomly select one connection and set its value to zero. We use the same network parameters but the total no. of iterations is 2048. The results illustrate that it can successfully distinguish the 6 patterns.

6. Conculsion

In the paper, we mainly try to illustrate a new approach in the unsupervised learning inspired from electrostatics phenomenon. As a conclusion we summarize our investigation as the following:

1. Clarification of the mechanism and the limitation of competitive learning.

2. Application of a simple criterion for an unsupervised learning system in pattern recognition.

3. Establishment an algorithm of unsupervised learning based on simple electrostatics phenomenon.

4. The implementation of the concept - chaos - for pattern recognition.

Meanwhile , we are trying to find out a mathematical proof to show that this learning algorithm can in fact achieve a stable state.

References

1. Rosenblatt F., "The Perceptron: The Probabilistic Model for Information Storage and Organization in the Brain," **Psychological Review**, 65, 386-408, 1958.
2. McCleland J.L and Rumelhart D.E., "An Interactivation Model of Context Effects in Letter Perception: Part 1. An Account of Basic Finding," **Psychological Review**, 88, 357-407,1981.
3. Rumelhart D.E. and McCleland J.L., "An Interactive Activation Model of Context Effects in Letter Perception: Part 2. The Contextual Enhancement Effect and Some Tests and Extensions of the Model," **Psychological Review**, vol 89, 6094,1982.
4. Rumelhart D.E. and Zipser D., "Feature Discovery by Competitive Learning," **Cognitive Science** 9, 75-112, 1985.
5. Grossberg S., "Competitive Learning: From Interactive Activation to Adaptive Resonance," **Cognitive Science** 11, 23- 63, 1987.
6. Carpenter G.A. and Grossberg S., "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine," **Computer Vision, Graphics, and Image Processing**, 37, 54-115, 1987.
7. Kraus J., **Electromagnetics.** 3rd Eds. McGraw Hill, 1984.