1000-9825/2001/12(08)1128-12 ©2001 Journal of Software 软件学报 Vol.12, No.8

A Novel Recurrent Neural Network for Face Recognition*

ZHAO Jie-yu

(*The IST Research Institute, Ningbo University, Ningbo* 315211, *China*)

E-mail: zhao_jieyu@sina.com

http://www.nbu.edu.cn

Received September 22, 2000; accepted March 9, 2001

Abstract: A novel stochastic neural network is proposed in this paper. Unlike the traditional Boltzmann machine, the new model uses stochastic connections rather than stochastic activation functions. Each neuron has very simple functionality but all of its synapses are stochastic. It is shown that the stationary distribution of the network uniquely exists and it is approximately a Boltzmann-Gibbs distribution. It is also revealed there exists a strong relationship between the model and the Markov random field. New efficient techniques are developed to implement simulated annealing and Boltzmann learning. The model has been successfully applied to a large-scale face recognition task in which face images are dynamically captured from a video source. Learning and recognizing processes are carried out in real time. The experimental results show the new model is not only feasible but also efficient.

Key words: stochastic binary network; incremental Boltzmann learning; Markov random field; simulated annealing

Stochastic computing utilizes what is generally regarded as a waste product — random noise. It is expected to provide a powerful technology for machine learning and pattern recognition. In the 1960's, Rosenblatt noticed that the noise and randomness present in the nervous system were not merely inconvenience because of poor design and construction, but were essential to the kind of computation brains performed. In a realistic situation it is necessary for a computing system to learn to make association with randomness since the environment is noisy and variable.

The neural network of a stochastic nature is very promising for global optimization. It is able to avoid the system being trapped in local minima. In a stochastic neural network the information is usually fuzzy and noisy and each neuron does not have to make very fine discriminations on its inputs. The output of the network should not be greatly affected by a single neuron but only determined by the global state of the network. This is achieved by building a totally distributed representation into the network. The distributed representation will make the network very robust in a noisy environment and improve the generalization performance[1].

In general a stochastic neural network can be built by introducing a random mechanism into the traditional deterministic network of sigmoid neurons. The well-known model is the Boltzmann machine which uses a stochastic

* Supported by the National Natural Science Foundation of China under Grant No. 69805002 (国家自然科学基金); the Natural Science Foundation of Zhejiang Province of China (浙江省自然科学基金); and the Education Grant for Excellent Youth of Ministry of China (教育部优秀青年教师基金)

ZHAO Jie-yu was born in 1965. He is an associate professor at the Department of Computer Science, Ningbo University. He received a Ph.D. degree in the Computer Science from Royal Holloway College, University of London in 1995. He was a post doctoral research fellow at IDSIA, Switzerland from 1995 to 1998. His research interests are machine learning, neural computation and biometrics.

sigmoidal activation function. Other techniques such as the 'pulse stream approach'[2], 'stochastic logic neural network'[3], and 'spiking neurons'[4~6] are also widely used.

In this paper we investigate a novel network which uses stochastic connections rather than stochastic activation functions. It is shown that there is a deviation of its functionality from that of the standard sigmoid neural network. The behavior of the network is similar to a Boltzmann machine when the size of the net is large. New techniques for simulated annealing and Boltzmann learning have been developed for the new model.

Training a neural network for face recognition is a very challenging task. Much of the present literature on face recognition with neural networks presents results with only a small size network and a simple structure. The mostly adopted network is of feed-forward multi-layer structure[7]. Some methods use a hybrid structure[8,9]. We present a large-size recurrent neural network solution using the proposed model. The network has 4827 neurons and 129951 connections. All neurons except the input units are fully connected. Face images are directly captured from a video source. The system carries out incremental learning and on-line recognition in real time.

1 Definition

A recurrent stochastic binary network is a parallelly distributed system in the form of a graph with the following restrictions:

• The edges of the graph are stochastic connections associated with certain values called weights. Each connection functions as a data path. The passing rate of the data is determined by the value of the corresponding weight. The connections can be excitatory or inhibitory. The former does not change the sign of the data passed on it, and the latter does change the sign.

• The nodes of the graph are two-state units. They have a dynamic updating mechanism which allows the units to change their states according to those of their neighbours. If the net input to a unit is greater than zero then it outputs 1; if the net input is less then zero it outputs −1; if the net input is exactly equal to zero then it randomly chooses 1 or −1 with probability 1/2 each.

We now give a formal definition of the network we have described above.

Definition 1. (**RSBN**) A recurrent stochastic binary network $N(V,W,F)$ is a pseudo graph with vertex set V having state $S\in\{-1,+1\}^V$ and edge set W which is a $n\times n$ matrix of random variables W_{ij}. Each W_{ij} has the following probability density function:

$$f_{ij}(x)=\begin{cases}(1+\beta w_{ij})/2 & x=1\\ (1-\beta w_{ij})/2 & x=-1\\ 0 & \text{otherwise}\end{cases} \quad (1)$$

where $-1\leqslant w_{ij}\leqslant+1$ is the weight value of W_{ij}, and $0\leqslant\beta\leqslant1$ is a control parameter. F is a dynamic updating mechanism of the network. It selects the vertices and changes their states with updating rule

$$S_i=\text{sgn}\Big(\sum_{j\neq i}W_{ij}S_j\Big)\quad \text{if } \sum_{j\neq i}W_{ij}S_j\neq 0.$$

If $\sum_{j\neq i}W_{ij}S_j=0$, then set S_i to 1 or −1 with probability 1/2.

With the different ways of updating RSBN can have the following two models:

• sequential RSBN, in which units change their states one by one.

• parallel RSBN, in which units change their states simultaneously.

We say that connections in a recurrent stochastic binary network are symmetric if $W_{ij}=W_{ji}$ for all i and j.

It can be seen that all the weights and the output values can be represented by stochastic binary sequences if we use a bipolar Bernoulli sequence representation[10]. The representation of a real value by a stochastic binary sequence is the central idea of 'stochastic computing'[11]. Given a real value r in $[-1,1]$, we may represent it by a stochastic binary sequence with probability $p=(r+1)/2$. With a bipolar representation the multiplication of two real values in $[-1,1]$ can be easily implemented by an Exclusive-Nor gate[10]. The highly efficient multiplication is one of the main advantages of the model.

2 The Stationary Distribution

In statistical mechanics a fundamental result tells us that when a physical system reaches thermal equilibrium each of the possible states α of the system occurs with probability

$$P_\alpha=\frac{1}{Z}e^{-H_\alpha/k_B T},$$

where the normalizing factor

$$Z=\sum_\alpha e^{-H_\alpha/k_B T}.$$

H_α is the Hamiltonian function, k_B is a constant and T is the temperature. This is the well-known Boltzmann-Gibbs distribution.

In a sequential or partial parallel Boltzmann machine it can be proved that the steady state of the network obeys the Boltzmann-Gibbs distribution[12].

In a sequential RSBN the steady state of the network also approximately obeys the Boltzmann-Gibbs distribution if the size of the network is not too small.

We first define the energy function in the recurrent stochastic binary network.

Definition 2. The Energy Function in a recurrent stochastic binary network is given by

$$H=-\frac{1}{2}\beta\sum_{ij}w_{ij}S_iS_j, \tag{2}$$

where β is the control parameter, w_{ij} is the weight value associated with the stochastic connection $W\,ij$ and S_i,S_j the states of neuron i and j.

Theorem 1. The stationary distribution of a sequential RSBN uniquely exists when the control parameter $\beta\in[0,1]$ and the stationary distribution is approximately a Boltzmann-Gibbs distribution if the connections are symmetric and the size of the network is not too small.

Proof. The proof is achieved by using the theory of Markov chains. First we prove the existence of the stationary distribution.

For a sequential RSBN $N(V,W,F)$ which randomly chooses a unit k and then updates it with the defined rule, the transition probability p_{ij} is given by

$$p_{ij}=\frac{1}{n}\left(Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l<0\right)+\frac{1}{2}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)\right)$$

and

$$p_{ii}=1-\sum_{j\neq i}p_{ij}.$$

Here the state i is denoted by the state vector $\{S_1,S_2,\ldots,S_k,\ldots,S_n\}$ and the state j denoted by $\{S_1,S_2,\ldots,-S_k,\ldots,S_n\}$. If the control parameter $\beta\in[0,1]$, we have $-1<\beta*w_{uv}<1$ for every connection W_{uv}. Thus

$$p_{ij}\geq\frac{1}{2n}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)>0.$$

Because the unit to be updated is randomly chosen from all n units, for each pair of the states $a,b\in S$ there is a positive probability of reaching b from a in finite steps. i.e. the Markov chain is irreducible.

The transition probability p_{ij} can be rewritten as

$$p_{ij}=\frac{1}{n}\left(1-Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l>0\right)-\frac{1}{2}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)\right)$$
$$\leqslant\frac{1}{n}\left(1-\frac{1}{2}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)\right)<\frac{1}{n}.$$

Hence,

$$p_{ii}=1-\sum_{j\neq i}p_{ij}>1-\frac{1}{n}(n-1)=\frac{1}{n}>0.$$

Thus the Markov chain is aperiodic.

The stationary distribution uniquely exists since the Markov chain is irreducible and aperiodic.

Now we prove the stationary distribution is approximately a Boltzmann-Gibbs distribution if the connections are symmetric and the size of the network is not too small.

If the network has symmetric connections we have the following energy function

$$H=-\frac{1}{2}\beta\sum_{uv}w_{uv}S_uS_v.$$

The difference between the energy of states j and i is

$$H_j-H_i=2\sum_{l\neq k}\beta w_{kl}S_kS_l.$$

The sum $\sum_{l\neq k}\beta w_{kl}S_kS_l$ is exactly the mean of the sum of the random variables $\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l$.

For a network with a large number of units the distribution of the sum of weighted inputs to each unit approaches a Gaussian distribution. The transition probability p_{ij} can be estimated by a cumulative Gaussian distribution function.

$$p_{ij}=\frac{1}{n}\left(Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l<0\right)+\frac{1}{2}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)\right)\approx\int_{-\infty}^{\mu/\sigma}\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}dx,$$

where μ is the mean and σ is the standard deviation,

$$\mu=\sum_{l\neq k}\beta w_{kl}S_kS_l,\qquad \sigma=\sqrt{\sum_{l=1,l\neq k}^{n}(1-\beta^2w_{kl}^2)}.$$

According to the result of Zhao[10], we have

$$p_{ij}\approx\frac{1}{1+e^{-1.699\mu/\sigma}}=\frac{1}{1+e^{-0.8495(Hj-Hi)/\sigma}}.$$

Similarly we can obtain

$$p_{ji}\approx\frac{1}{1+e^{-0.8495(Hi-Hj)/\sigma}}.$$

If we choose $q_i=\frac{e^{0.8495H_i/\sigma}}{Z}$ and $q_j=\frac{e^{0.8495H_j/\sigma}}{Z}$, where Z is the normalizing factor, the detailed balance equation holds. Therefore the stationary distribution is approximately a Boltzmann-Gibbs distribution. □

3 Simulated Annealing

Simulated annealing in a stochastic neural network is implemented by selecting units at random and updating them according to the probability

$$p_{ij}=\frac{1}{1+e^{(H_j-H_i)/T}},$$

where H is the energy function and T the temperature. By gradually lowering the T from a high initial value to a low point with a certain cooling schedule, the energy function in the network will hopefully reach a global minimum.

In a recurrent stochastic binary network, simulated annealing is controlled by the standard deviation σ which has exactly the same function as the temperature. The larger is the σ, the larger the degree of the randomness in the network. When σ reaches its minimum 0 the network reduces to the deterministic case.

In order to implement the simulated annealing in a RSBN we use the parameter β to control the standard deviation σ. The initial value of β is set to be zero. Thus the standard deviation

$$\sigma=\sqrt{\sum_{l=1,l\neq k}^{n}(1-\beta^2 w_{kl}^2)}=\sqrt{n-1}$$

reaches its maximum. The network becomes completely random. Then we gradually increase the value of β from 0 to 1 with a certain schedule. This will decrease the randomness in the network. Hopefully the process will minimize the energy function of the network.

Simulated annealing process brings a stochastic system to a stable state. It can be used in a variety of aspects such as the Boltzmann learning and the combinatorial optimization. The Boltzmann learning rule is usually applied when the system reaches a stable state via a simulated annealing process. To find the optimal solution of an optimization problem we need to map its cost function to the energy function of the network. A simulated annealing process minimizes the energy function and gives out the solution. Simulated annealing can also be used to optimize the gradient descent learning process[13].

4 RSBNs and Markov Random Fields

There is a very strong relationship between the stochastic recurrent binary networks and the Markov Random Field models. It was shown in Section 2 that the stationary distribution of a sequential recurrent stochastic binary network with symmetric connections uniquely exists and it is approximately a Boltzmann-Gibbs distribution. Thus, a recurrent stochastic binary network can be practically treated as a Markov Random Field. Now we consider the recurrent stochastic binary network from a viewpoint of Markov Random Fields. This will give us some useful insight into the model.

The energy function we defined in a RSBN has the following form

$$H=-\frac{1}{2}\beta\sum_{i=1}^{n}\sum_{j=1}^{n}w_{ij}S_iS_j,$$

where β is the control parameter, w_{ij} is the weight value associated with the stochastic connection W_{ij} and S_i, S_j the states of neuron i and j.

With the above energy function, the RSBN approximately obeys the Boltzmann-Gibbs distribution. It is very likely that the energy function is the limit case of the exact energy function (which still remains unknown) when the size of the network gets large.

We have pointed out that the Markov chain associated with a sequential RSBN is irreducible and aperiodic. Therefore, the stationary distribution $\boldsymbol{q}=(q_1,q_2,\ldots q_m)$ uniquely exists, where m is the number of total global states of the network. According to the result of Markov theory, we have the following equations

$$\sum_{j=1}^{m}q_jp_{ji}=q_i,\ \text{for } i=1,2,\ldots,m, \tag{3}$$

where p_{ji} is the transition probability from global state j to global state i. For a sequential RSBN, p_{ji} is non-zero only when the states j and i are neighbours, i.e., only one unit in the network is in the reversed state for i and j. If we denote the global state j by $\{S_1,S_2,S_3,\ldots,S_k,\ldots,S_n\}$ and the global state i by $\{S_1,S_2,S_3,\ldots,-S_k,\ldots,S_n\}$, then

$$p_{ji}=\frac{1}{n}\left(Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l<0\right)+\frac{1}{2}Prob\left(\sum_{l=1,l\neq k}^{n}W_{kl}S_kS_l=0\right)\right)$$

and the probability that the network stays unchanged is

$$p_{ii}=1-\sum_{i\neq j}p_{ji}.$$

Obviously, it is too difficult to solve the equations exactly since the number of the global states m is usually very large. However, we have also shown that the detailed balance equation roughly holds for the energy function of (2) when the size of the network is large. If the detailed balance equation holds in the real case, that is, the unique solution of the Eq. (3) q satisfies the equation

$$q_i p_{ij}=q_j p_{ji}\quad \text{for all } i \text{ and } j,$$

then we can prove a RSBN is exactly a MRF.

Theorem 2. A sequential Recurrent Stochastic Binary Network is a Markov Random Field if the detailed balance equation holds for its stationary distribution.

Proof. We denote the unit i in the network as v_i and its state as S_i, $i=1,2,\ldots,n$. The conditional probability

$$P(S_i|S_j;v_j\neq v_i)=\frac{P(S_1,S_2,\ldots,S_n)}{P(S_1,\ldots,S_i=1,\ldots,S_n)+P(S_1,\ldots,S_i=-1,\ldots,S_n)}.$$

For any $v_u\notin Q(v_i)$, where $Q(v_i)$ is the set of neighbours of v_i, let a,b,c,d,e,f denote the global states $(S_1,S_2,\ldots,S_u=1,\ldots,S_n)$, $(S_1,S_2,\ldots,S_u=-1,\ldots,S_n)$, $(S_1,S_2,\ldots,S_i=1,\ldots,S_u=1,\ldots,S_n)$, $(S_1,S_2,\ldots,S_i=-1,\ldots,S_u=1,\ldots,S_n)$, $(S_1,S_2,\ldots,S_i=1,\ldots,S_u=-1,\ldots,S_n)$, and $(S_1,S_2,\ldots,S_i=-1,\ldots,S_u=-1,\ldots,S_n)$ respectively, the conditional probability

$$P(S_i|S_j;\ v_j\neq v_i \text{ and } v_j\neq v_u)=\frac{q_a+q_b}{q_c+q_d+q_e+q_f}.$$

If the detailed balance equation holds, we have

$$q_a p_{ab}=q_b p_{ba};\ q_c p_{ce}=q_e p_{ec};\ q_d p_{df}=q_f p_{fd}.$$

Because $v_u\notin Q(v_i)$, and a,c and d are the neighbouring states of b, e and f respectively, the transition probabilities

$$p_{ab}=p_{ce}=p_{df}=t_1>0;$$

and

$$p_{ba}=p_{ec}=p_{fd}=t_2>0.$$

Therefore,

$$\frac{q_a+q_b}{q_c+q_d+q_e+q_f}=\frac{(1+t_1/t_2)q_a}{(1+t_1/t_2)q_c+(1+t_1/t_2)q_d}=\frac{q_a}{q_c+q_d}$$

and similarly,

$$\frac{q_a+q_b}{q_c+q_d+q_e+q_f}=\frac{q_b}{q_e+q_f}.$$

Because of the bipolar state of the unit, the global state $(S_1,S_2,\ldots,S_n)$ is either a or b; correspondingly, $(S_1,S_2,\ldots,S_i=1,\ldots,S_n)$ is either c or e, and $(S_1,S_2,\ldots,S_i=-1,\ldots,S_n)$ is either d or f. Hence,

$$P(S_i|S_j;\ v_j\neq v_i)=P(S_i|S_j;\ v_j\neq v_i \quad \text{and } v_j\neq v_u).$$

Since v_u was any unit not a neighbour of v_j, we have that the conditional probability $P(S_i|S_j;v_j\neq v_i)$ only depends on the units in $Q(v_i)$. Hence,

$$P(S_i|S_j;\ v_j\neq v_i)=P(S_i|S_j;\ v_j\in Q(v_i)).$$

So the RSBN is a Markov Random Field. □

5 RSBNs and Boltzmann Machines

The stochastic sigmoid neuron used in the Boltzmann Machine is governed by the update rule

$$Prob(S=1)=\frac{1}{1+e^{-(\sum_j w_j v_j-\theta)/T}}$$

and

$$Prob(S=0)=1-Prob(S=1)=\frac{1}{1+e^{(\sum_j w_j v_j-\theta)/T}},$$

where S is the state of the neuron, $T>0$ is the so-called temperature.

By making the neuron behave stochastically the network is capable of escaping from the local minima. Taking the zero-temperature limit will reduce the stochastic neuron to a deterministic binary version, but the finite temperature extension will prove very useful, such as in the simulated annealing process where T acts as a crucial control parameter.

The main similarities of a RSBN and a Boltzmann machine are listed as follows.

- Neurons in both networks are stochastic and have two discrete states.
- Stochastic behavior of both networks can be controlled by certain parameters.
- The connections within the networks may be excitatory or inhibitory, i.e. the weights can be positive or negative.
- The strength of the connection determines the degree of the influence from the input. Both networks have adaptable connections.
- The two networks have a very similar functionality if the size of the network is large.

In addition to the above similarities, a number of differences can be identified between these two networks.

- The stochastic behavior in a RSBN comes from its connections, but in a Boltzmann machine it is generated by the neuron according to the net contributions from the inputs.
- The temperature T in a Boltzmann machine is an abstract concept derived from a real physical system, which is used to control the degree of randomness of a unit; but in a RSBN the values of its weights naturally determine the degree of randomness.
- All the weights of a RSBN are limited in the range $[-1,1]$; but in a Boltzmann machine every weight can become arbitrarily large. The limitation of the weight values in a Boltzmann machine will cause a change of the range in its output value and severely affect its learning process.

Generally speaking, the functionality of a RSBN and a Boltzmann machine appears quite different if the size of the network is small, their behavior become similar with each other when the number of the neuron and the connection increases.

6 Incremental Boltzmann Learning

The ability to learn is one of the most essential characteristics of intelligent behavior. It is also the most important aspect of neural networks. One useful definition of learning for recurrent stochastic networks involves matching probabilities between the environment and the network. During the learning process we attempt to adjust the weights w_{ij} to give the states of the visible units a particular desired probability distribution.

We have shown that the stationary distribution of RSBNs is approximately a Boltzmann-Gibbs distribution in the previous section. Therefore the following Boltzmann learning rule can be directly applied to RSBN which has some remarkable features.

$$\Delta w_{ij}=\varepsilon(\langle\overline{S_iS_j}\rangle_c-\langle S_iS_j\rangle_f). \tag{4}$$

The term $\langle\overline{S_iS_j}\rangle_c$ is essentially a Hebb term, with the visible units clamped, while the second term $\langle S_iS_j\rangle_f$ corresponds to Hebbian unlearning with the system free running. ε is the learning rate. The learning involves first letting the system run freely. The probabilities of the states taken by each unit can be estimated. Then the visible units are clamped, that is, forced to take appropriate values. Again, values of the probabilities of the states of the units are estimated. Then each local weight change is proportional to the difference in the probabilities of the units connected by the weight. The process converges when the difference of these values becomes zero for each pair of the units in the network.

The estimation of the correlations $\langle S_iS_j \rangle$ in both free running and clamped phases should be done in equilibrium. In the absence of appropriate hardware the system has to be studied by Monte Carlo simulation. Unfortunately this procedure takes a very long time to come to equilibrium at low temperature T. The solution is to use a simulated annealing procedure with a gradual lowering of the temperature from a high initial value to the desired value. This procedure is very computationally intensive.

The Boltzmann learning rule (4) does not completely determine the learning algorithm. It is still necessary to decide how long to collect the statistics for the correlation estimation, and what temperature schedule to use in the annealing procedure. A commonly encountered difficulty with Boltzmann Machine is that the weights grow too large since there is nothing that can effectively stop such growing. Large weights would create such high-energy barriers that the network would not reach equilibrium in the arranged time. Once this happens, the statistics required for the estimation of correlations will not be the equilibrium statistics required. Thus the whole learning process will be damaged.

However, all the weights in a RSBN are between -1 and 1, it overcomes a main difficulty encounter in the general Boltzmann machine. The limitation of the weight value also leads to a complete distribution of the information over the whole network. This is expected to improve the generalization performance.

The design of our RSBN is finally aimed at an efficient hardware implementation with on-chip learning. We use a simplified incremental learning method. A brief description of the learning process is as follows.

(1) When there is an input pattern the system is in a clamped phase. That is, all the input and output units are clamped to particular values; when there is no input pattern, the system is in a free-running phase.

(2) Begin a simulated annealing process in which the randomness is controlled by the parameter β. Set β to 0, then gradually increase the value of β from 0 to 1 with a certain schedule. At the end of simulated annealing process, collect the states of all neurons and estimate the correlations $\langle S_iS_j \rangle$ between each pair of the neurons.

(3) During the clamped phase, we add a small term $\varepsilon * \langle S_iS_j \rangle$ to the weight w_{ij} associated with the random connection W_{ij}.

(4) During the free-running phase, we subtract a small term $\varepsilon * \langle S_iS_j \rangle$ from w_{ij}.

(5) Alternatively set the system to the clamped phase and the free-running phase, repeat steps (1)～(4) until the stop criterion is reached.

The efficiency of such incremental learning can be explained by the energy distribution over the whole configuration space. In the clamped phase, each learning step decreases the energy corresponding to the particular learning pattern, and this decrease is always the greatest among all the changes of the energy for other patterns. Since the probability corresponding to the particular pattern is exponentially proportional to the minus energy, this pattern gets more chance to appear in future. During the free-running phase, the learning is simply a de-correlation process. This is helpful for continuous learning and may increase the number of the patterns stored in the network. It is obvious that the incremental learning is entirely local and appropriate for on-chip implementation in hardware.

7 Human Face Recognition

There is increased interest in biometrics for reliable personal identification. The computerized access control has a wide variety of applications. Human face recognition has the benefit of being a passive, non intrusive system for verifying personal identity. Many researchers treat the face recognition as a high-level task with many stages of processing. However, some evidence which show the face recognition process may also based on a low-level image processing. Physiological experiments in monkey cortex reveal there are isolated neurons responding selectively to faces[8]. Artificial neural network is believed to have certain basic features of the human brain. It may deal sensibly with the face recognition problem which is still very badly defined. Our ultimate aim is to develop a novel and unified neural computing framework for face recognition.

We first apply our recurrent stochastic binary network to a static face recognition problem to test its learning and generalizing ability. Then the model is used to learn and recognize faces from a video data stream.

7.1 Static face recognition

The image database we used for static face recognition was a part of the FERET database at the MIT Media Lab. There are 200 frontal human images of 100 distinct subjects. Each image is in the size of 128×192 pixels and with 256 grey scales. 100 images taken from different individuals are used as learning samples and the other 100 are test samples which are the images of the individuals appeared in the learning samples with some variations in expressions (smiling/non-smiling, open/closed eyes), appearances (hairstyle, glass/no glass), illumination and poses. All of the test images are contaminated with 5% uniform noise. Typical image samples are shown in Fig. 1.

Fig. 1 Some learning samples (top) and test samples (bottom)

The recurrent stochastic binary network has a structure of 54×72 input neurons, 15 hidden neurons and 7 output neurons. The network is fully connected except the weights between input neurons. It can distinguish $2^7=128$ different patterns in total but only 100 of them are used in this experiment. The network is trained with the incremental learning algorithm and the simulated annealing technique proposed in the previous sections. Each learning pattern is normalized before it is used as the inputs of the network. That is, the size of the image is reduced to 54×72 pixels, all the input grey scales between 0 and 255 are mapped into the range $[-1,1]$.

The network successfully learned all the 100 input images after 12376 learning cycles. The learning rate is 0.003. The result shows the trained network is able to recognize 91 percent of the test images.

7.2 Dynamic face recognition

We are primarily interested in the case of identifying particular people in real-time. There are a small group of people need to be recognized but for each one it allows varying facial detail, expression, pose, rotation, illumination etc. Multiple images per person are available for training and real-time recognition is required. The face image is directly obtained from a video source.

A preliminary system has been developed with Windows 2000 platform SDK. All programmes are written in Visual C++ 6.0. The system is a combination of three main parts—face detection, face tracking, and face recognition. There are many elaborate methods for face detection and face tracking[14,15]. Most of them are far too computationally expensive. We need a system which is able to detect and track objects in the presence of noise, other faces and hand movements. Moreover, it must run fast and efficiently so that the performance can be carried out in real time (30 frames per second). The face detector firstly checks the moving objects in the video scene and then finds out the flesh color distribution. The largest moving area with flesh color will be used as the candidate for tracking at the beginning. The face tracker continuously calculates the mean and the area of the flesh color distribution and always shifts the center of the tracking window to the mean location of the distribution[16,17]. The size of the tracking window is scaled according to the area of the flesh color. The system works as follows:

(1) Find out the moving objects in the video scene according to the difference between the successive video frames.

(2) Check whether there exists an moving object with flesh color. If there are more than one such object, choose the largest one as the tracking candidate.

(3) If the user would like to track and recognize a non-facial moving object or there are no moving objects with flesh color in the video scene, set the tracking target manually.

(4) If the neural network is well trained, take the current tracking candidate as the input of the network, check whether the candidate is a face. Otherwise, skip step 4.

(5) Calculate the flesh color distribution including the mean and the size.

(6) Rescale the tracking window size and track the object with mean shift algorithm.

(7) Take the image within the tracking widow as the input, start enrollment or learning or recognition.

The user interface of the system is shown in Fig. 2. The top frame in the window is used to show the enrolled people. The video image is shown in the left frame of size 640×400. The frame at the right side is for the information exchange between the user and the system. A user should first enroll itself before using the recognition system. The user's name and a face snap are stored into the database.

Fig. 2 The user interface of the recognition system

The face image in the tracking window (the white frame) is normalized and changed into grey scales before it is inputted into the neural network. The size of the image is rescaled to 60×80. The brightest point in the image is set to $+1$ and the darkest set to -1. All the other points are linearly mapped into the range $[-1,1]$. The recurrent neural network contains 4 800 input units, 20 hidden units and 7 output units. It has 4827 neurons and 129951 connections in total. The network is fully connected except the input neurons. Each neuron has no connection to itself. The learning rate is 0.002 in this experiment.

The supervised learning process is as follows. The user should first set the learning target by clicking the corresponding name in the name list. Then start learning while the target is correctly tracked. The system takes the face image in the tracking window, normalizes and rescales the image, and uses it as the inputs of the neural network. If the network gives out the wrong output, calls an incremental Boltzmann process; if the output of the network is right, skip the current learning cycle and begin the next cycle. The supervised learning process is stopped either by the user or by a stopping criterion——for a certain number of the consecutive facial images the network gives the right answers.

TrueFace is a competitive face recognition product developed by the eTrue company. It is featured with fast face recognition, good accuracy and accommodating to the variations of the face image. Compared to this commercial product, our system has some unique features:

(1) The recognition process in both systems is carried out in real time. Face detection in TrueFace makes use of the location of the flesh color and eyes. It is usually difficult to enroll for a user wearing glasses. This problem doesn't exist in our system.

(2) TrueFace only records four face snaps from each user. It is only able to recognize the user in few poses. The learning mechanism in our system, which learns from a sequence of face images, is more powerful. Our system is able to recognize a face with wider variations.

(3) TrueFace only works with a facial image. Our system is developed to recognize not only a human face but also a moving object with a uniform color distribution.

8 Conclusions

A novel model called the recurrent stochastic binary network has been proposed in this paper. It has been proved that the stationary distribution of the network is approximately a Boltzmann-Gibbs distribution if the size of the network is not too small. A very strong relationship between the sequential recurrent stochastic binary network and the Markov random field has been revealed. The simulation results on human face recognition demonstrate the power of the model.

References:

[1] Rumelhart, D.E., McClelland, J.L., and the PDP Research Group. Parallel Distributed Processing. The MIT Press, 1986.

[2] Murray, A., Tarassenko, L. Analogue Neural VLSI-a Pulse Stream Approach. Chapman & Hall, 1991.

[3] Kondo, Y., Sawada, Y. Functional abilities of a stochastic logic neural network. IEEE Transactions on Neural Networks, 1992,3:434~443.

[4] Maass, W. Computation with spiking neurons. The Handbook of Brain Theory and Neural Networks. 2nd Edition, MIT Press, 2001.

[5] Banerjee, A. On the phase-space dynamics of systems of spiking neurons. I: Model and Experiments. Neural Computation, 2001,13:161~193.

[6] Banerjee, A. On the phase space dynamics of systems of spiking neurons. II: Formal Analysis. Neural Computation, 2001,13:195~225.

[7] Mao, K.Z., Tan, K.C., Ser, W. Probabilistic neural-network structure determination for pattern classification. IEEE Transactions on Neural Networks, 2000,11(4):1009~1016.

[8] Lawrence, S., Giles, C.L., Tsoi, A.C., *et al.* Face recognition: a convolutional neural network approach. IEEE

Transactions on Neural Networks, 1997,8(1):98~113.

[9] Gutta, S., Huang, J.R.J., Jonathon, P., *et al*. Mixture of experts for classification of gender, ethnic origin, and pose of human faces. IEEE Transactions on Neural Networks, 2000,11(4):948~960.

[10] Zhao, J. Stochastic bit stream neural networks [Ph.D. Thesis]. London University, 1995.

[11] Poppelbaum, W.J., Dollas, A., Glickman, J.B., *et al*. Unary processing. Advances in Computers, 1987,26:47~92.

[12] Aarts, E., Korst, J. Simulated Annealing and Boltzmann Machines. John Wiley & Sons, 1989.

[13] Chuang, C., Su, S., Hsiao, C. The annealing robust backpropagation (ARBP) learning algorithm. IEEE Transactions on Neural Networks, 2000,11(5):1067~1077.

[14] Sung, K., Poggio, T. Learning human face detection in cluttered scenes. Computer Analysis of Images and Patterns, 1995. 432~439.

[15] Turk, M., Pentland, A. Eigenfaces for recognition. Journal of Cognitive Neuroscience, 1991,3:71~86.

[16] Cheng, Y. Mean shift, mode seeking, and clustering. IEEE Transactions on Pattern Analysis Machine Intelligence, 1995, 17:790~799.

[17] Fukunaga, K. Introduction to Statistical Pattern Recognition. Academic Press, Boston, 1990.

可用于人脸识别的反馈型二元神经网络

赵杰煜

(宁波大学 信息科学与技术研究所,浙江 宁波 315211)

摘要:提出和分析了一种新型的反馈型随机神经网络,并将其用于解决复杂的人脸识别问题.该模型采用随机型加权联接,神经元为简单的非线性处理单元.理论分析揭示该网络模型存在唯一的收敛平稳概率分布,当网络中神经元个数较多时,平稳概率分布逼近于Boltzmann-Gibbs分布.网络模型与马尔可夫随机场之间存在密切关系.在设计了一种新型模拟退火和渐进式Boltzmann学习算法后,系统被成功地应用于难度较大的静态和动态人像识别,实验结果证实了系统的可行性和高效率.

关键词:反馈型随机二元神经网络;渐进式Boltzmann学习;马尔可夫随机场;模拟退火

中图法分类号:TP391 **文献标识码**:A