

Image Restoration Technology Based on Discrete Neural network

Duoying Zhou

College of Computer and Control Engineering, Nankai University, Tianjin, China

ABSTRACT: With the development of computer science and technology, the development of artificial intelligence advances rapidly in the field of image restoration. Based on the MATLAB platform, this paper constructs a kind of image restoration technology of artificial intelligence based on the discrete neural network and feedforward network, and carries out simulation and contrast of the restoration process by the use of the bionic algorithm. Through the application of simulation restoration technology, this paper verifies that the discrete neural network has a good convergence and identification capability in the image restoration technology with a better effect than that of the feedforward network. The restoration technology based on the discrete neural network can provide a reliable mathematical model for this field.

Keywords: artificial intelligence; mathematical model; discrete neural network; image restoration

1 PREFACE

In human life and production activities, the majority of information acquired is visual information, such as the plate number, auto logo or model required to be identified in daily traffic. However, the role of nature often leads to incomplete information. For example, the note fonts might fade after several years, thus resulting in difficulty in obtaining commercial evidence or other activities. How to restore the incomplete information becomes a problem to be solved by modern human beings. Since the advent of the neural network, it is widely used and promoted to the image restoration technology. This paper implements the theory based on the classical mathematical theory, applies the discrete neural network in the field of image restoration, and compares the identification effect of the feedforward network.

2 MODELING

2.1 Theory of BP neural network

BP neural network refers to a kind of multilayer feedforward bionic algorithm. This algorithm has two main features: The first feature is the forward transmission of information; the second feature is the reverse transmission of error. There is no interaction between neurons. The change of values has an inherited effect, which is repeatedly recycled until reaching the desired error, and training a matrix that is in line with the expected rate of specific gravity^[3].

In Figure 1, X_1, X_2, \dots, X_n are input values of forward transmission; Y_1, Y_2, \dots, Y_m are predicted values of reverse transmission; ω_{ij} and ω_{jk} are specific gravity of the reverse transmission.

As shown in Figure 1, the reverse-transmission neural network is essentially a nonlinear function; the independent variable is an input value of the network; the dependent variable is an output value of the network, thereby building a function relation from the dimension (n) to the dimension (m).

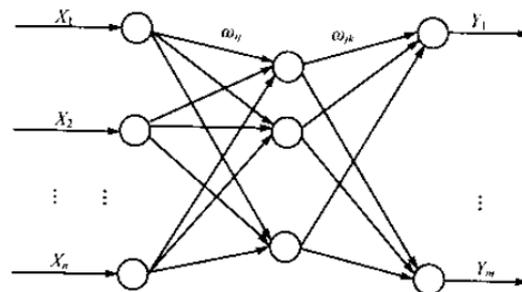


Figure 1. Structure chart of feedforward neural network

The training network can make data become standard and network more intelligent. The training steps are as follows:

First step: initialize the network. Determine the number of nodes at the typing layer (n), the number of nodes at the hidden layer (l) and the number of nodes at the printing layer (m) according to the typing and printing matrix(X,Y). Initialize the specific gravity connected between the neurons at the typing layer and the printing layer (ω_{ij} and ω_{jk}), the range at the hidden layer (a), the range at the printing layer (b) and a given acquisition rate and agitation function.

Second step: output at the hidden layer. Determine the number of nodes at the typing layer (n), the specific gravity connected between the hidden layers (ω_{ij}) and the range (a) according to the matrix(X,Y), so as to calculate the output at the hidden layer (H).

This is an Open Access article distributed under the terms of the Creative Commons Attribution License 4.0, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

$$H_j = f\left(\sum_{i=1}^n \omega_{ij}x_i - a_j\right) \quad j=1,2,\dots,l \quad (1)$$

In the Formula (1), l is the number of nodes at the hidden layer; f is an agitation function.

Third step: output at the printing layer. The predicted output (O) of bionic algorithm can be calculated by the output at the hidden layer (H), the specific gravity of connection (ω_{jk}) and the range (b).

$$O_k = \sum_1^l H_j \omega_{jk} - b_k \quad k=1,2,\dots,m \quad (2)$$

Fourth step: error calculation. The prediction error (e) can be calculated by the predicted printing (O) and expected printing (Y).

$$e_k = Y_k - O_k \quad k=1,2,\dots,m \quad (3)$$

Fifth step: update of specific gravity. The specific gravity of connection (w_{ij}, w_{jk}) can be updated according to the prediction error (e).

$$w_{ij} = w_{ij} + \eta H_j (1 - H_j) x(i) \sum_{k=1}^m w_{jk} e_k \quad j=1,2,\dots,n \quad j=1,2,\dots,l \quad (4)$$

In the Formula (4), η is the learning rate.

Sixth step: update of range. The range (a, b) can be updated to the prediction error (e) in the algorithm.

$$a_j = a_j + \eta H_j (1 - H_j) \sum_{k=1}^m w_{jk} e_k \quad j=1,2,\dots,l$$

$$b_k = b_k + e_k \quad k=1,2,\dots,l \quad (5)$$

Seventh step: determine whether it is finished or not. If it cannot meet the standard, then return to the second step^[4].

2.2 Working principle of Hopfield network

Hopfield proposed this brand new network in 1982 and named it as Hopfield network. The network is characterized by full connection. Its architectural feature and algorithm learning is different from the past. The network structure at earlier stage is the class type, while this network is the binary type. Hopfield neural network has the function of association, speculation, memorization and logic judgment via imitation of the cerebral nerve, which has obtained success in various industry applications. Moreover, numerous engineering practices have verified the feasibility of this algorithm.

The neural network has only two output values, namely 1 (activated state) and -1 (depressed state). These two different states can also be represented by other figures. Therefore, the Hopfield network is discrete. And this paper adopts Hopfield binary neurons.

The architecture of Hopfield network is different

from that of other networks. Its architecture is shown in Figure 2:

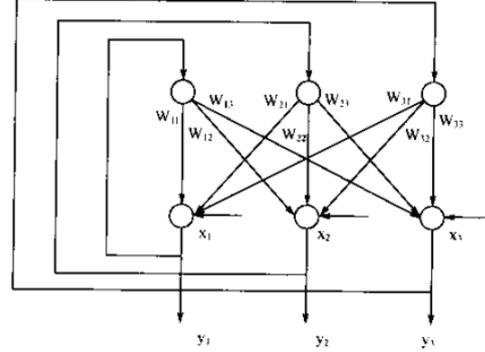


Figure 2. Structure chart of Hopfield network

The Hopfield network has one output layer and only two output values (1 or -1). The base layer shown in the figure is the network typing, which is not designed for calculation, so it is not a real neuron^[2]. The neuron at the summation layer (the second layer) is a real neuron. The continued product of the typed value and weight, and the cumulative sum can be calculated. The print value is generated after processing by the non-linear conversion $T(u)$; $T(u)$ is a permission function, which can judge the print value. If it is greater than 0, then the output value which is the set value is 1; if it is less than 0, then the output value is -1. The formula is:

$$u_j = \sum_i w_{ij} y_i + x_j \quad (6)$$

In Formula (6), x_j is the external typed value, and satisfies:

$$\begin{cases} y_j = 1, u_j \geq 0 \\ y_j = -1, u_j \leq 0 \end{cases} \quad (7)$$

Hopfield network output is related to the tense. If an n -dimensional vector is output at the moment of t , this vector is an integration of meta-information.

$$Y(t) = [y_1(t), y_2(t), y_3(t), \dots, y_n(t)]^T \quad (8)$$

The node state of the Hopfield network changes over the time t . In $y_i(t) (i=1,2,\dots,n)$, 1 and -1, or other figures can be used to represent the state. In short, there are 2^n kinds of state. Therefore, the next moment state can be inferred from the meta-state of the moment. The formula is:

$$y_j(t+1) = f[u_j(t)] = \begin{cases} 1, u_j(t) \geq 0 \\ -1, u_j(t) < 0 \end{cases} \quad (9)$$

$$u_j(t) = \sum_{i=1}^n w_{ij} y_i(t) + x_j - \theta_j \quad (10)$$

2.3 Working mode of the network

The evolution process of the Hopfield network is a dynamic mode, and the meta-state has the dynamic characteristics. Its process is in line with Lyapunov function, which is similar to the nature and moves toward energy reduction. When the network reaches an equilibrium value, then the state is the print value.

The working modes of the network are respectively serial and parallel with some differences. The operating steps are as follows:

First step: initialize the Hopfield network.

Second step: randomly select the meta i .

Third step: calculate meta-output ($u_i(t)$) at the moment of t .

Fourth step: calculate the output of the meta i ($v_i(t-1)$) at the moment of $t-1$, and keep the print value of other meta unchanged.

Fifth step: give a judgment of the steady state, and then do cycle. The steady state refers to unchanged output after the moment of t_i , that is:

$$v(t + \Delta t) = v(t) \quad \Delta t > 0 \quad (11)$$

2.4 Learning rules of the discrete Hopfield network

To make the Hopfield network stable, symmetrical and converging, there is generally a need of rule-based design, such as the commonly-used exterior product method and interpolation method.

1. Exterior product method

Assuming a given sample matrix $\{t^1, t^2, \dots, t^N\}$, if the value of t^k is ± 1 , then its link value can be learnt by the exterior product method. The formula is as follows:

$$W = \sum_{k=1}^N [t^k (t^k)^T - I] \quad (12)$$

The steps of Hopfield can be summarized as follows:

First step: compile the weight matrix based on the sample matrix and following (12).

Second step: The test matrix $p_i (i=1, 2, \dots, n)$ is the initial print value $y_i(0) = p_i (i=1, 2, \dots, n)$. Then we set the iteration number.

Third step: calculation. The formula is:

$$y_i(k+1) = f\left(\sum_{j=1}^N w_{ij} y_j\right) \quad (13)$$

Fourth step: determine the steady condition, and then cycle.

2. Orthogonalization method

This algorithm is roughly as follows:

First step:

print the input matrix $t = \{t^1, t^2, \dots, t^{n-1}, t^n\}$ and parameters τ and h .

Second step:

calculate $A = \{t - t^n, t^2 - t^n, \dots, t^{n-1} - t^n\}$.

Third step: Make the singular value decomposition ($A = USV^T$) of the above matrix, and calculate its rank ($K = \text{rank}(A)$).

Fourth step: calculate $T^p = \sum_{i=1}^N u^i (u^i)^T$ and

$T^m = \sum_{i=K+1}^N u^i (u^i)^T$ through $U^p = \{U^1, U^2, \dots, U^k\}$ and

$u^m = \{u^{K-1}, u^{K-2}, \dots, u^N\}$.

Fifth step:

calculate $W^t = T^p - \tau T^m, b^t = t^N - W^t t^N$.

Sixth step: calculate $W = \exp(hW^t)$

Seventh step: calculate

$$b = U \times \begin{bmatrix} C_1 \times I(k) & 0(K, N-K) \\ 0(N-K, K) & C_2 \times I(N-K) \end{bmatrix} \times U^T \times b^t$$

Where: $C_1 = \exp(h) - 1$; $C_2 = -[\exp(-\tau \times h) - 1] / \tau$

3 MODEL SOLUTION

3.1 Construction of image restoration technology

There is a first need to establish a steady-state matrix. This paper needs to reconstruct a digital image with ten matrices from 0 to 10 as an ideal domain. The dimension of the matrix is 10×10 , and its discretization is done by 1 and -1. The filling domain is 1, while the blank domain is -1. The standard image is shown in Figure 4. The network has memory and identification capability for the steady-state matrix. To connect with the discrete matrix of the image to be restored and the network, it can realize the identification, associative memory, and correct judgment of the matrix, so as to achieve the purpose of correct identification.

The flow chart of network design is shown in Figure 3.

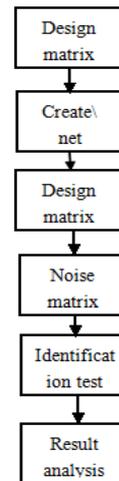


Figure 3. Flow chart of network design

Table 1. Matrix of Figure 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	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	-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	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	-1

Table 2. Matrix of Figure 2

-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	-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
-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	-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

The ideal image can be discretized through MATLAB. The digital matrix of Figures 1 and 2 is respectively shown in Table 1 and Table 2.



Figure 4. Standard image

The discrete network is constructed after the establishment of the digital matrix, and the image to be processed is discretized, and then imported to MATLAB database. After parameter testing, the image can be restored. The simulation steps are as follows:

First step: import the training matrix:

$T = [array_one; array_two]^T$, and test the matrix.

Second step: create the discrete network: $net = newhop(T)$, and connect with the matrix and network for simulation.

Third step: simulation test: *for i to n*

$noisy_i = (noisy_array_i)^T$

$identity_i = sim(net, \{10, 10\}, \{\}, noisy_i)$

$n = 0, 2 \dots 9$

Fourth step: result analysis.

The simulation results are shown in Figure 5-simulation result diagram.

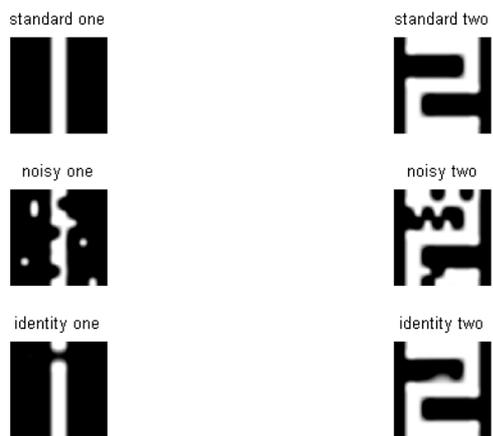


Figure 5. Simulation result diagram

The simulation steps of the feedforward network are similar to that of this network. The simulation results are shown in Figure 6:

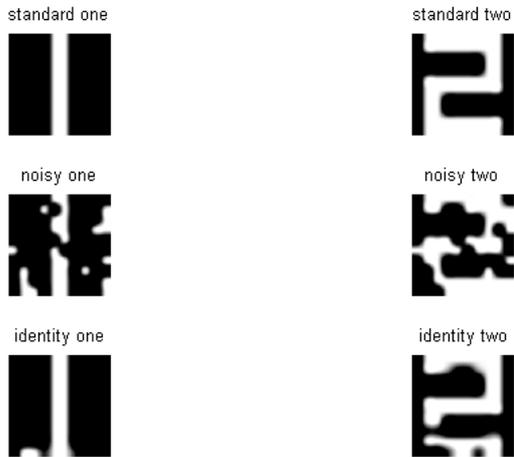


Figure 6. Simulation result diagram of feedforward network

The diagram indicates that the discrete network has a good restoration capacity for the images with missing information, and enables to handle complex images. The restoration capacity of the feedforward neural network is inferior to that of the discrete network, and the convergence rate of the feedforward neural network is slow.

4 CONCLUSION

The discrete Hopfield neural network and feedforward network are respectively used to restore the images with missing information, so as to establish a model of

image restoration technology based on artificial intelligence, realize the restoration simulation, and verify the feasibility of this principle and commercial value. The following conclusions can be obtained by contrast:

1. The discrete network has a good restoration capacity for the images with missing information.
2. The discrete network enables to handle complex images with a fast convergence rate.
3. The artificial intelligence has a broad application prospect in the image restoration technology.

REFERENCES

- [1] Abdul S Shigtullah, Tillal Eldabi, Jasna Kuljis. 2006. A Proposed Multi-agent Model for Bus Crew Scheduling. Proceedings of the 2006 Winter Simulation Conference.
- [2] BAI E W, LI Q Y, & ZHANG Z Y. 2005. Blind source separation/channel equalization of nonlinear channels with binary inputs. *IEEE Transactions on Signal Processing*.
- [3] Yunshan Sun, Liyi Zhang, Ting Liu, & Yanqin Li. 2006. Study on Fuzzy Neural Network Classifier Blind Equalization Algorithm. 2006 IEEE International Conference on Information Acquisition.
- [4] J K Tugnait. 1995. On blind equalization of multi path channels using fractional sampling and second order cyclostationary statistics. *IEEE Transactions on Information Theory*.
- [5] Yingbo Hua. 2001. Blind identification and equalization of channels driven by colored signals. *Signal Processing Advances in Wireless & Mobile Communications*.
- [6] J·J· Hopfield. 1982. Neural Networks and physical Systems with Emergent Collective Computational Abilities. *proc Natl. sci.*