Hamilton neural-network model: recognition of the color patterns

Jianwei Shuai, Zhenxiang Chen, Ruitang Liu, and Boxi Wu

A 16-state Hamilton neural-network model is discussed. The storage capacity of the model is analyzed through theory and through a computer numerical simulation. The storage-capacity ratio of the presented model equals that of the Hopfield model. This 16-state neural network can be applied to the recognition of 16-level color patterns, and some examples are discussed.

Key words: Hamilton number, neural network, pattern recognition.

1. Introduction

During the past few years, much attention has been focused on neural-network models.[1–3] Some research work in the field of pattern recognition has been done with neural-network models.[4–9] Some multistate neural-network models are discussed to process gray-level patterns. Rieger[7] suggested that, in a Q-state neural-network model,[11] multistate signals are represented by Q integers. Noest[8] and Sirat[9] independently proposed another technique, called the discrete-state phasor neural network, in which they mapped the gray level to the phase domain. However, colors are the basic information carriers of any natural scene. When making a gray version of a colored pattern, we not only lose some information but also some beauty of the scene.

In this paper, a discrete Hamilton–number neural network is discussed. The model is especially appropriate to application of recognition of 16-level color patterns. In the model, each neuron is assumed to be a Hamilton number $(\pm 1, \pm i, \pm j, \pm k)$. The signal-to-noise theory and computer numerical simulations are used to analyze the storage stability and the storage capacity of the model. Then its application to the recognition of a 16-level color pattern is discussed.

2. Hamilton Number and the Hamilton Neural Network

We are familiar with the natural, integral, real, and complex numbers. However, a type of multidimensional numbers are defined, i.e., 2^n-element numbers,[10] in mathematics. For $n = 0, 1$, a multidimensional number stands for the real and the complex numbers; for $n = 2, 3, 4$, it stands for the Hamilton number, the Cayley number, and the Clifford number, respectively. Let us assume that the Hamilton number $Q(R) = \{\alpha\colon\ \alpha = a + b\hat{i} + c\hat{j} + d\hat{k};\ a, b, c,\ d \in R\}$, where $\hat{i}$, $\hat{j}$, $\hat{k}$ are the three basic unit vectors of the Hamilton number. The addition among the Hamilton numbers is defined as usual, as is the multiplication between the real number and the Hamilton number. The multiplication between the basic unit vectors is defined as

$$\hat{i}^2 = \hat{j}^2 = \hat{k}^2 = -1, \qquad \hat{i}\hat{j} = -\hat{j}\hat{i} = \hat{k},$$
$$\hat{j}\hat{k} = -\hat{k}\hat{j} = \hat{i}, \qquad \hat{k}\hat{i} = -\hat{i}\hat{k} = \hat{j}. \tag{1}$$

So the multiplication between two Hamilton numbers is to expand the expression $(a_0 + a_1\hat{i} + a_2\hat{j} + a_3\hat{k})(b_0 + b_1\hat{i} + b_2\hat{j} + b_3\hat{k})$ by means of the distribution law. From Eq. (1) we know that the Hamilton numbers don't obey the exchange law of multiplication, but they do obey the combination law, i.e., $(\alpha\beta)\gamma = \alpha(\beta\gamma)$. A Hamilton number $\alpha = a + b\hat{i} + c\hat{j} + d\hat{k}$ has a conjugate number that is denoted by an asterisk and is defined as $\alpha^* = a - b\hat{i} - c\hat{j} - d\hat{k}$. So $\alpha\alpha^* = a^2 + b^2 + c^2 + d^2$.

The Hamilton number is introduced into the neural network so as to form a discrete Hamilton neural-network model. In the model, each neuron is to be a 16-state one that has one of the following values:

$$\pm 1, \pm\hat{i}, \pm\hat{j}, \pm\hat{k}.$$

Suppose there are N neurons and M patterns S^μ $(\mu = 1, 2, \ldots, M)$ stored in the network. The Hamilton connection matrix is given by the extended Heb-

The authors are with the Department of Physics, Xiamen University, Xiamen, Fujian, 361005 China.

Received 23 June 1994; revised manuscript received 7 March 1995.

0003-6935/95/296764-05$06.00/0.

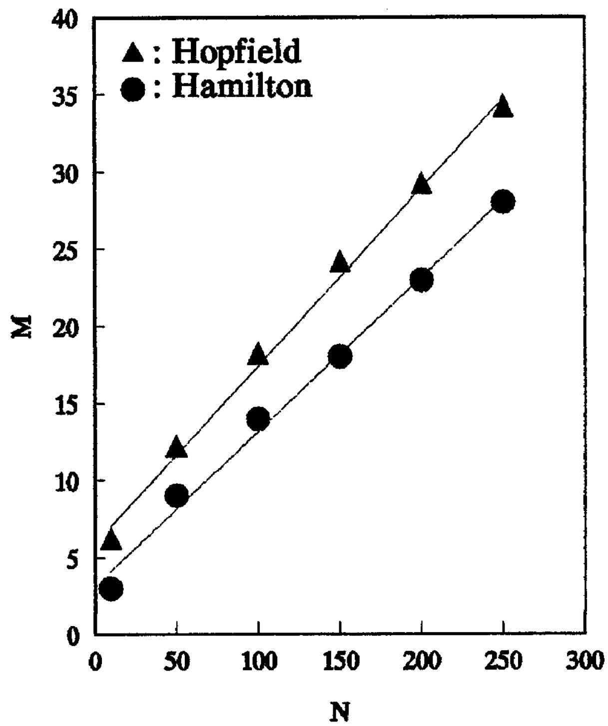

Fig. 1. Statistical curves of the stored pattern number M by means of the neuron number N under the correct retrival percentage (50%).

bian learning rule:

$$J_{mn} = (1 - \delta_{mn}) \sum_{\mu=1}^{M} S_m^{\mu} (S_n^{\mu})^*. \tag{2}$$

The dynamics of the system are defined by the generalized Hamilton step function sgn:

$$S_m(t + 1) = \mathrm{sgn}\left[\sum_{n=1}^{N} J_{mn} S_n(t)\right]. \tag{3}$$

The Hamilton operating rule of the sgn(α) is as follows: whenever a real or imaginary component of α is not smaller than zero, a positive unit is drawn out for the corresponding component of sgn(α); otherwise, a negative unit is drawn out. For example,

$$\mathrm{sgn}(-2 + 5\hat{i} - 3\hat{k}) = -1 + \hat{i} + \hat{j} - \hat{k}.$$

3. Theory Analysis and Numerical Simulation

Here, an analysis of the stability and the storage capacity of the Hamilton dynamic system is presented on the basis of the signal-to-noise theory. With the assumption that the stored patterns are $S^{\mu} = a^{\mu} + b^{\mu}\hat{i} + c^{\mu}\hat{j} + d^{\mu}\hat{k}$, ($a^{\mu}, b^{\mu}, c^{\mu}, d^{\mu} = \pm 1$ and $\mu = 1, \ldots, M$), Eqs. (2) and (3) can be rewritten in terms of each real and imaginary part, respectively. If the pattern $S(0) = S^{\nu}$ is put into the dynamic system, the dynamic equation $S^{\nu'} = \mathrm{sgn}(JS^{\nu})$ can be expanded as follows:

$$\begin{aligned}
a_m^{\nu'} &= \mathrm{sgn}[4(N-1)a_m^{\nu} + \Delta_0],\\
b_m^{\nu'} &= \mathrm{sgn}[4(N-1)b_m^{\nu} + \Delta_i],\\
c_m^{\nu'} &= \mathrm{sgn}[4(N-1)c_m^{\nu} + \Delta_j],\\
d_m^{\nu'} &= \mathrm{sgn}[4(N-1)d_m^{\nu} + \Delta_k],
\end{aligned} \tag{4}$$

where

$$\begin{aligned}
\Delta_0 = \sum_{\mu\neq\nu}\sum_{n\neq m} [& a_m^{\mu}(a_n^{\mu}a_n^{\nu} + b_n^{\mu}b_n^{\nu} + c_n^{\mu}c_n^{\nu} + d_n^{\mu}d_n^{\nu})\\
&+ b_m^{\mu}(b_n^{\mu}a_n^{\nu} - a_n^{\mu}b_n^{\nu} + c_n^{\mu}d_n^{\nu} - d_n^{\mu}c_n^{\nu})\\
&+ c_m^{\mu}(c_n^{\mu}a_n^{\nu} - a_n^{\mu}c_n^{\nu} + d_n^{\mu}b_n^{\nu} - b_n^{\mu}d_n^{\nu})\\
&+ d_m^{\mu}(d_n^{\mu}a_n^{\nu} - a_n^{\mu}d_n^{\nu} + b_n^{\mu}c_n^{\nu} - c_n^{\mu}b_n^{\nu})],\\
\Delta_i = \sum_{\mu\neq\nu}\sum_{n\neq m} [& b_m^{\mu}(b_n^{\mu}b_n^{\nu} + c_n^{\mu}c_n^{\nu} + d_n^{\mu}d_n^{\nu} + a_n^{\mu}a_n^{\nu})\\
&+ c_m^{\mu}(c_n^{\mu}b_n^{\nu} - b_n^{\mu}c_n^{\nu} - d_n^{\mu}a_n^{\nu} + a_n^{\mu}d_n^{\nu})\\
&+ d_m^{\mu}(d_n^{\mu}b_n^{\nu} - b_n^{\mu}d_n^{\nu} - a_n^{\mu}c_n^{\nu} + c_n^{\mu}a_n^{\nu})\\
&+ a_m^{\mu}(a_n^{\mu}b_n^{\nu} - b_n^{\mu}a_n^{\nu} - c_n^{\mu}d_n^{\nu} + d_n^{\mu}c_n^{\nu})].
\end{aligned} \tag{5}$$

The terms Δ_j and Δ_k can also be written out easily but are ignored here for simplicity. Clearly, for every Eq. (4), the first term on the right side is the signal, whereas the second, Δ, is the noise. One finds that the value $4(N - 1)$ is the nonnormalized probability that the state S^{ν} is a fixed point. The noise terms Δ expressed by Eq. 5 are sums of $4(M - 1)$ components of the stored patterns, each of which has a weight equal to the inside summation of $4(N - 1)$ terms. Owing to the random character of the stored patterns and their independence from each other, it is reasonable to suppose that the noise term Δ is governed by a

Table 1. Computer Codes and Neuron Codes of the 16-Level Colors

Level	Color	Codes: Computer	Codes: Neuron	Level	Color	Codes: Computer	Codes: Neuron
0	Black	0000	$-1-i-j-k$	8	Gray	1000	$1-i-j-k$
1	Blue	0001	$-1-i-j+k$	9	Light blue	1001	$1-i-j+k$
2	Green	0010	$-1-i+j-k$	10	Light green	1010	$1-i+j-k$
3	Cyan	0011	$-1-i+j+k$	11	Light cyan	1011	$1-i+j+k$
4	Red	0100	$-1+i-j-k$	12	Light red	1100	$1+i-j-k$
5	Magenta	0101	$-1+i-j+k$	13	Light magenta	1101	$1+i-j+k$
6	Brown	0110	$-1+i+j-k$	14	Yellow	1110	$1+i+j-k$
7	White	0111	$-1+i+j+k$	15	Intense white	1111	$1+i+j+k$

Gaussian distribution with the expectation value's being zero and the standard deviation's value being $[16(N-1)(M-1)]^{1/2}$. So the signal-to-noise ratio σ of the real part or of the three imaginary parts of each component of S^{ν} is $[(N-1)/(M-1)]^{1/2}$.

If $N \gg M$, then $\sigma \gg 1$, and so the neural network converges to the stored pattern S^{ν}. When a pattern S that is close to the stored pattern S^{ν} is put into the network, the above conclusion essentially holds true, and the pattern S automatically converges to the pattern S^{ν} after one or more retrieval processes. In conclusion, the stored pattern S^{ν} is a stable attractor of the Hamilton neural network.

Now the storage capacity ratio $\rho = M/N$ of the presented model, with the limit in which $N, M \to \infty$ and ρ is finite. Without a loss of generality, let us assume that the real component $\mathrm{Re}(S_m^{\nu}) = 1$; then the probability that the $\mathrm{Re}(S_m^{\nu'}) = 1$ can be given on the basis of Gaussian distribution of the noise term Δ is

$$p = \frac{1}{\sqrt{2\pi}} \int_{-\sqrt{1/\pi}}^{\infty} \exp\left(-\frac{x^2}{2}\right) \mathrm{d}x. \tag{6}$$

Here, $\sigma \approx \sqrt{1/\rho}$. Thus, compared with the pattern S^{ν}, the expected number of errors in the real and imaginary components in the pattern $S^{\nu'}$ is approximately

$$E(\rho) = \frac{4N}{\sqrt{2\pi}} \int_{\sqrt{1/\pi}}^{\infty} \exp\left(-\frac{x^2}{2}\right) \mathrm{d}x. \tag{7}$$

If the number of error components in $S^{\nu'}$ is approximately a Poisson distribution, it follows that the probability of correct components, i.e., the probability that S^{ν} is indeed a stable attractor, is given approximately by the expression

$$P = \exp[-E(\rho)]. \tag{8}$$

Suppose that the probability P is a fixed number very near 1. We can invert the preceding expression for $0 < \rho \ll 1$ and obtain the following result:

$$\rho \propto \frac{1}{2 \ln 4N} \approx \frac{1}{2 \ln N}. \tag{9}$$

This result is equal to that of the Hopfield neural-network model analyzed by Bruce *et al.*[2] and Mceliece *et al.*[3] So the storage capacities of the Hamilton–number neural-network model and of the Hopfield model are almost at the same level.

Now the numerical simulation of the storage capacity of the model for a finite-size system $N \leq 300$ is performed. The statistical curve of the maximum number of stored patterns M by means of the neuron number N under the correct retrieval percent (50%) is obtained (see Fig. 1). The statistical curve $M - N$ of the Hopfield model is also plotted. From Fig. 1 one can see that the curves for the two models are similar; in another words, the storage capacity ratios of the

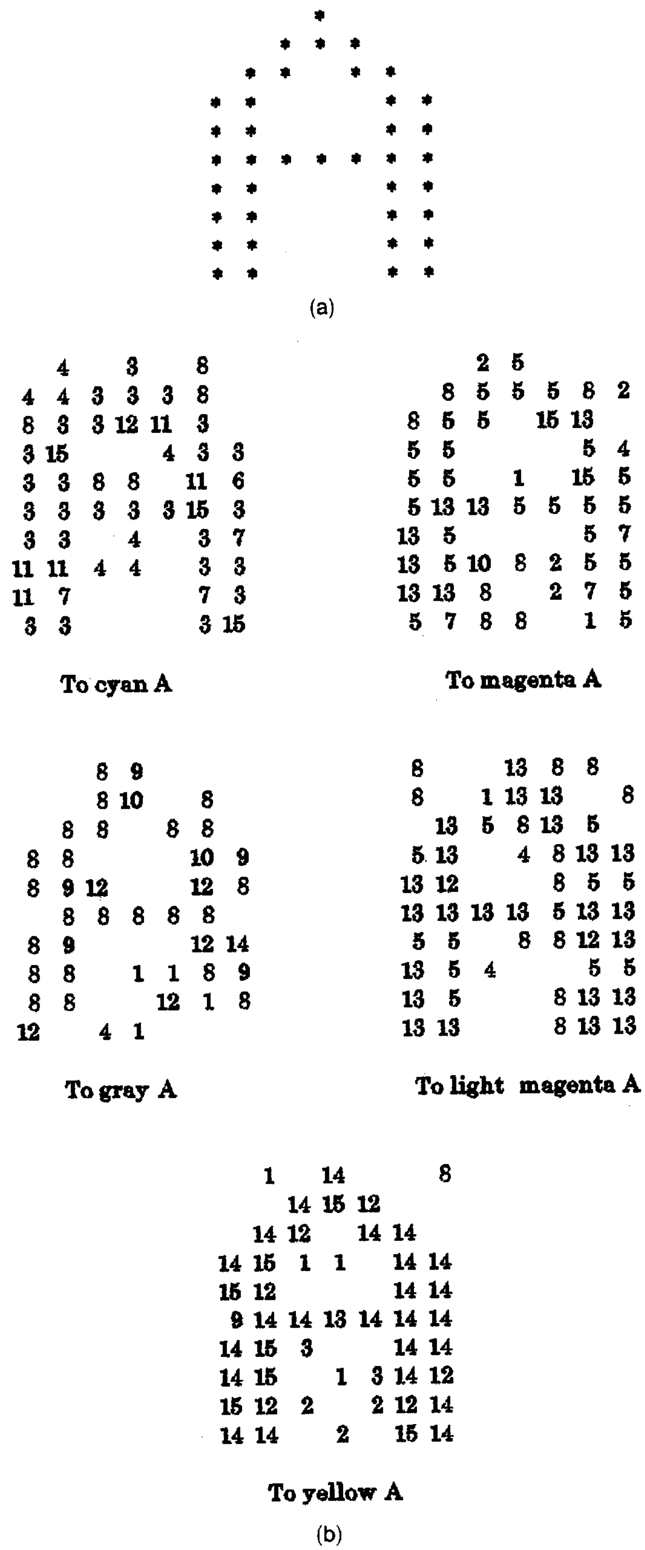

Fig. 2. Computer-simulation results for the recognition of six English letter A's, one composed of only symbols and five composed of color levels but with one color that predominates: cyan level 3, magenta level 5, gray level 8, light magenta level 13, and yellow level 14. In (a) the letter is composed of a 7 dot × 10 dot matrix. In (b) the input for the color letters included 10% noise. All these letters with noise can be recognized correctly by the model.

two models are the same. The storage capacity of the Hamilton network is slightly smaller than that of the Hopfield network.

This conclusion can be qualitatively comprehended as follows: In the Hopfield model,[1] the one-dimensional input information is cut into two parts—the positive and the negative—and mapped into two points of the output space; in the Q-state model,[7] the

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
4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4
4	4	4	4	4	4	4	4	4	4	4	4
9	9	9	9	9	9	9	9	9	9	9	9
9	9	9	9	9	9	9	9	9	9	9	9
9	9	9	9	9	9	9	9	9	9	9	9
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11
11	11	11	11	11	11	11	11	11	11	11	11

Stored Pattern 1

2	3	2	9	0	1	1	15	11	3	0	3
0	0	1	9	0	1	1	2	11	3	1	1
1	1	1	0	1	3	1	0	0	9	1	1
7	6	12	4	6	15	13	4	8	7	12	12
7	13	5	14	4	4	0	5	0	4	12	5
5	4	4	4	12	5	6	12	4	4	4	4
11	11	9	1	8	9	3	9	1	9	10	1
1	10	15	9	8	9	11	9	8	8	9	1
11	9	8	9	9	11	9	9	1	9	9	9
11	1	1	3	10	2	2	11	9	11	9	11
11	10	9	2	11	11	11	15	3	11	11	11
11	10	11	9	3	11	11	10	11	11	10	9

Recognized as Pattern 1

14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10
14	14	14	2	2	2	5	5	5	10	10	10

Stored Pattern 2

14	12	14	11	3	10	4	13	4	10	14	10
6	7	15	6	2	10	4	5	5	10	7	2
14	14	6	11	2	2	0	5	4	11	11	2
14	6	14	14	3	3	5	4	5	14	2	11
14	6	6	10	2	3	5	3	12	10	10	0
6	6	14	2	2	10	4	13	5	10	10	10
14	14	14	10	10	11	13	12	13	2	10	6
15	3	15	2	10	10	5	13	5	10	2	11
15	14	6	2	3	10	1	4	5	11	2	10
14	14	14	10	10	2	13	5	12	2	10	10
7	7	14	3	10	10	4	13	13	10	2	10
14	15	7	2	2	11	5	5	13	11	6	6

Recognized as Pattern 2

8	8	8	8	8	8	13	13	13	13	13	13
5	8	8	8	8	8	13	13	13	13	13	2
5	5	8	8	8	8	13	13	13	13	2	2
5	5	5	8	8	8	13	13	13	2	2	2
5	5	5	5	8	8	13	13	2	2	2	2
5	5	5	5	5	8	13	2	2	2	2	2
5	5	5	5	5	8	13	2	2	2	2	2
5	5	5	5	8	8	13	13	2	2	2	2
5	5	5	8	8	8	13	13	13	2	2	2
5	5	8	8	8	8	13	13	13	13	2	2
5	8	8	8	8	8	13	13	13	13	13	2
8	8	8	8	8	8	13	13	13	13	13	13

Stored Pattern 3

12	12	0	8	0	14	1	13	12	5	13	15
5	8	8	8	10	8	9	14	9	1	13	10
5	1	8	10	9	10	1	15	13	13	12	2
4	5	13	0	0	8	7	13	9	4	2	2
13	13	5	5	12	9	7	13	2	12	6	3
5	5	5	4	3	8	5	2	0	2	2	0
7	5	5	7	5	12	15	2	6	12	2	2
5	9	11	9	10	8	13	5	6	2	2	6
5	5	4	8	8	12	13	13	13	2	6	2
7	5	15	12	8	8	15	11	9	3	6	6
5	10	8	0	8	10	13	13	3	13	12	6
8	8	8	4	4	10	13	15	13	15	15	3

Recognized as Pattern 3

14	14	14	14	14	14	14	14	14	14	14	14
14	5	5	5	5	5	5	5	5	5	5	14
14	5	5	5	5	5	5	5	5	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	11	11	11	11	11	11	5	5	14
14	5	5	5	5	5	5	5	5	5	5	14
14	5	5	5	5	5	5	5	5	5	5	14
14	14	14	14	14	14	14	14	14	14	14	14

Stored Pattern 4

(a)

14	14	2	6	6	14	6	6	6	14	6	14
6	5	5	13	5	5	5	5	13	5	5	6
6	1	9	13	13	13	5	1	5	13	5	6
6	13	5	3	11	11	3	7	11	9	13	7
6	5	1	11	11	11	3	3	11	5	13	10
14	5	13	11	3	11	11	7	11	13	13	14
6	13	9	11	11	5	3	11	11	13	9	2
6	5	5	3	7	3	15	11	15	1	13	14
6	5	5	3	3	3	7	10	3	13	9	10
10	9	13	1	5	13	5	5	13	4	9	6
6	13	13	13	13	5	5	5	13	13	13	6
10	6	6	14	10	6	2	6	6	10	14	6

Recognized as Pattern 4

(b)

Fig. 3. Computer-simulation results for the recognition of the four color patterns: (a) The color patterns that are stored in the network, and (b) the input color patterns with approximately 20% noise. All the input patterns with noise can be recognized correctly by the model.

one-dimensional input information is cut into Q intervals, so the mean-square error of pseudo-orthogonality of the pattern S^ν is increased with Q, and thus it raises the noise of the system. To correctly map each interval into one of the Q points of the output space, the sensitivity of the dynamics with the noise is increased with Q^2, whereas for the Hamilton neural-network model the input information is placed in four-dimensional space and cut into 16 quadrants. So the difficulty encountered with the Q-state model is avoided. Thus the sensitivity of the present model with the noise is the same as that of the Hopfield model, i.e., the signal-to-noise ratio of the model equals that of the Hopfield model. The storage-capacity ratios of the networks are mainly assessed by σ. Therefore the storage capacity of the Hamilton model equals that of the Hopfield model.

4. Application to Color-Pattern Recognition

The Hamilton neural-network model is especially appropriate for recognizing 16-level color patterns consisting of three basic colors (red, blue, and green): The three imaginary parts of the Hamilton number can be treated as the three basic colors, and the real part indicates the color-saturation degree. So the 16-state Hamilton neural network can store 16-level color patterns. The corresponding relation between the computer code and the Hamilton neuron code of the 16-level colors is listed in Table 1.

Using the Hamilton neural network, we process numerical simulations to recognize the color of English letters that are composed of a 7 dot $\times$ 10 dot matrix, such as is seen in Fig. 2(a), in which the background color is black. For example, five letters A with each of the colors cyan, magenta, gray, light magenta, and yellow, are stored in the model. Numerical-simulation results show that these letters A, with different colors, are all the stable patterns stored in the Hamilton neural network. Now, if a letter that is slightly different from one of the stored A's is fed into the neural network, the system can recognize it and recall to the proper A. It shows that, for the input patterns added with approximately 10% random noise (i.e., 30 error bits), the correct recognition ratio is more than 90%. Figure 2(b) shows other letters A with 10% noise in the input patterns. The model can recognize them correctly. In the figure, each integer number indicates a color as shown in Table 1.

Some numerical simulations to recognize the color patterns are also processed by means of the presented model. Four color patterns composed of a 12 dot $\times$ 12 dot matrix, such as are shown in Fig. 3(a), are stably stored in the presented network. Figure 3(a) shows that, for input patterns with approximately 10% or 20% random noise (i.e., 60 or 110 error bits, respectively), the correct recognition ratio is nearly 95% or 75%, respectively. The set of patterns that are shown in Fig. 3(b) as input test objects but that are contaminated with approximately 20% noise can be recognized correctly by the model.

5. Conclusion

In this paper, a 16-state discrete Hamilton neural network is proposed. The stability and the storage capacity are analyzed by means of the signal-to-noise ratio theory and numerical simulation. The storage-capacity ratios of the two models are the same, and the storage capacity of the Hamilton network is just a little smaller than that of the Hopfield network. Because of the feature of one real part and three imaginary parts for the Hamilton number, the 16-state discrete Hamilton neural network can be applied to recognize a 16-level color pattern.

Naturally, not only can the Hamilton number be introduced into the neural network, but also the other 2^n-element numbers, such as the complex number and Cayley number. For the latter two types of numbers, four-level (i.e., ± 1, $\pm \hat{i}$) and 256-level (i.e., ± 1, $\pm \hat{i}_1$, $\pm \hat{i}_2$, $\pm \hat{i}_3$, $\pm \hat{e}$, $\pm \hat{i}_1 \hat{e}$, $\pm \hat{i}_2 \hat{e}$, $\pm \hat{i}_3 \hat{e}$) neural networks can be set up. The detailed work will be discussed in future papers.

This work is supported by the National Natural Scientific Research Foundation under grant 19334032.

References

1. J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," Proc. Natl. Acad. Sci. USA **79,** 2554–2558 (1982).
2. A. D. Bruce, E. J. Gardner, and D. J. Wallace, "Dynamics and statistical mechanics of the Hopfield model," J. Phys. A **20,** 2909–2934 (1987).
3. R. J. Mceliece, E. D. Posner, E. R. Rodemich, and S. S. Venkatesh, "The capacity of the Hopfield associative memory," IEEE Trans. Inf. Theory **33,** 461–482 (1987).
4. H. J. White and W. A. Wright, "Holographic implementation of the Hopfield model with discrete weightings," Appl. Opt. **27,** 331–338 (1987).
5. W. Zhang, K. Itoh, J. Tanida, and Y. Ichioka, "Hopfield model with multistate neurons and its optoelectronic implementation," Appl. Opt. **30,** 195–200 (1991).
6. F. T. S. Yu, C. M. Uang, and S. Yin, "Gray-level discrete associative memory," Appl. Opt. **32,** 1322–1329 (1993).
7. H. Rieger, "Storing an extensive number of grey-toned patterns in a neural network using multistate neurons," J. Phys. A **23,** 1273–1279 (1990).
8. A. J. Noest, "Discrete-state phasor neural networks," Phys. Rev. A **38,** 2196–2199 (1988).
9. G. Y. Sirat, A. D. Maruani, and Y. Ichioka, "Gray-level neural networks," Appl. Opt. **28,** 414–415 (1989).
10. E. U. Conden, H. Odishaw, *Handbook of Physics,* 2nd ed. (McGraw-Hill, New York, 1967), Vol. 1, p. 22.