

Holographic Implementation of a Fully Connected Neural Network

KEN-YUH HSU, HSIN-YU LI, AND DEMETRI PSALTIS, MEMBER, IEEE

Invited Paper

This paper describes an optical implementation of a fully connected neural network similar to the Hopfield network. Experimental results which demonstrate its ability to recognize stored images are given, followed by a discussion of its performance and analysis based on a proposed model for the system.

I. INTRODUCTION

In this paper we present a holographic implementation of a fully connected neural network [1], [2]. This model has a simple structure and is relatively easy to implement while its operating principles and characteristics can be extended to other types of networks, since any architecture can be considered as a fully connected network with some of its connections missing. In the following sections, the basic principles of the fully connected network are reviewed. The optical implementation of the network is presented in Section III and its experimental results are presented in Section IV. Special attention is focused on the dynamics of the feedback loop and the trade-off between distortion tolerance and image-recognition capability of the associative memory. Mathematical modeling and analysis of the system are presented in Section V.

II. THE HOPFIELD MEMORY

The basic structure of the network is shown in Fig. 1. It is a single-layer network with feedback. There are two main components: the neurons and the interconnections. The neurons are distributed in the neural plane. The neurons receive inputs, perform nonlinear thresholding on the received input, provide gain, and re-emit the output patterns. The output of each unit is connected to the input of all other neurons to form a feedback network.

Manuscript received September 25, 1989; revised April 12, 1990. This work is supported by DARPA and in part by the Air Force Office of Scientific Research.

K.-Y. Hsu was with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125, USA. He is now with the Institute of Electro-Optical Engineering, National Chiao Tung University, Hsin-Chu 30050, Taiwan, ROC.

H.-Y. Li and D. Psaltis are with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125, USA.

IEEE Log Number 9039181.

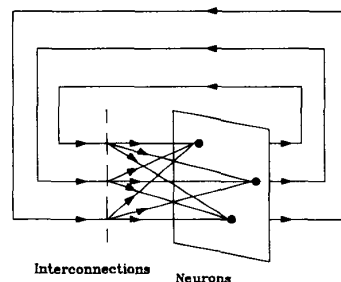


Fig. 1. Two dimensional fully connected network.

There are two phases in the operation of the network: learning and recall. In the learning phase, the information to be stored is recorded according to the outer product scheme [1], [2]. This storage specifies the interconnection strengths between the neurons. In the recall phase, an external input is presented to the system. The state of the system then evolves according to the correlation between the input and the stored data. M N -bit binary words are stored in a matrix $\omega_{i,j}$ according to

$$\omega_{i,j} = \begin{cases} \sum_{m=1}^M v_i^m v_j^m, & \text{if } i \neq j; \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where $v_i^m = \pm 1$, $i = 1, \dots, N$, is the i th bit of the m th memory. Suppose, for example, that \mathbf{v}^{m0} , the m th stored vector, is presented to the system in the recall phase. This vector is multiplied by the matrix $w_{i,j}$, giving the output of the first iteration:

$$\hat{v}_i^{m0} = \text{sgn} \left\{ \sum_{j=1}^N w_{i,j} v_j^{m0} \right\}, \quad (2)$$

where $\text{sgn}\{\cdot\}$ is the thresholding function,

$$\text{sgn}\{x\} = \begin{cases} 1, & \text{if } x \geq 0; \\ -1, & \text{if } x < 0. \end{cases} \quad (3)$$

The thresholded result of the first iteration is then fed back to the system as input for the next iteration, and so on.

There are three operations performed by the system: vector-matrix multiplication, thresholding, and feedback. A network of this type using optoelectronics was first implemented by Psaltis, Farhat, and their colleagues [3], [4]. They used a computer-generated transparency to provide the interconnection matrix. A 1-D array of 32 photodiode pairs followed by electronic thresholding plus a 1-D array of 32 LEDs was used to simulate 32 neurons. In this paper the optical implementation of such a system for 2-D images uses holograms. The design and implementation of this system are presented in the following section.

III. OPTICAL IMPLEMENTATION

The interconnection pattern for 2-D images is described by the following equation:

$$\omega(x, y; \xi, \eta) = \sum_{m=1}^M f_m(x, y) f_m(\xi, \eta), \quad (4)$$

where $f_m(x, y)$ is the m th image, and M is the total number of images to be stored. Note that $\omega(x, y; \xi, \eta)$ is a four dimensional kernel and cannot be implemented directly using a single transparency since a 2-D optical system has only two spatial coordinates. The system described in this paper is based on a method for implementing this 4-D kernel that uses a 2-D array of spatial frequency multiplexed holograms [3], [5], [6]. Jang *et al.* used a 2-D array of $N \times N$ diffused holograms to obtain the 4-D interconnection [7], [8]. Other approaches to this problem include the use of spatial multiplexing [9] and volume holograms [10]–[13].

In the recall phase, the output of the system is described by the equation

$$\hat{f}(x, y, t) = g \left\{ \iint \omega(x, y; \xi, \eta) f(\xi, \eta, t) d\xi d\eta \right\}, \quad (5)$$

where $g\{\cdot\}$ represents the nonlinear thresholding of the neurons, $f(x, y, t)$ is the input to the system at time t , and $\hat{f}(x, y, t)$ is the output of the system. Substituting the expression for $\omega(x, y; \xi, \eta)$ into this equation, and rearranging the order of integration and summation, we obtain

$$\hat{f}(x, y, t) = g \left\{ \sum_{m=1}^M f_m(x, y) \left[\iint f_m(\xi, \eta) \cdot f(\xi - x, \eta - y, t) d\xi d\eta \right]_{x=0, y=0} \right\}. \quad (6)$$

From (6) we see that the implementation of the 2-D associative memory can be achieved in three steps [6]. First the 2-D correlation between the input image f and each of the memories f_m is calculated, and then the correlation function is evaluated at the origin to obtain the inner product values. Second, each inner product is multiplied by the associated stored memory. Third, these products are summed over all memories and thresholded by the neurons.

The implementation of the system will be explained with the aid of Fig. 2. Here four images are spatially separated and stored as the reference images in each of two correlators. When one of the stored patterns A is presented at plane P_1 of the system, the first correlator produces the auto-correlation along with three cross-correlations of plane P_2 . The pinhole array at P_2 samples these correlation functions at the center of each pattern where the inner products

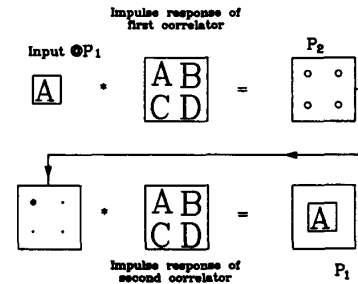


Fig. 2. Block diagram of the operations performed by the optical loop.

between the input and each of the stored images form. Each of the four signals that pass through the pinholes act as delta functions, reconstructing from the second correlator the four images that are stored there. These reconstructed images are spatially translated according to the position of each pinhole and superimposed at plane P_1 . At the center of the output plane of the second correlator we obtain the superposition of the four stored images. The stored image that is most similar to the input pattern gives the strongest correlation signal, hence the brightest reconstructed image. In Fig. 2 we show only the image that is reconstructed by the strongest auto-correlation peak. The weak read-out signal that is due to cross-correlations is suppressed by the thresholding operation of the neurons. The output from the plane of neurons becomes the new input image for the next iteration. In this way the stable pattern that is established in the loop is typically the stored image that is most similar to the original input.

The optical implementation makes use of the Vander Lugt correlator [14] shown in Fig. 3. If we place a hologram at the

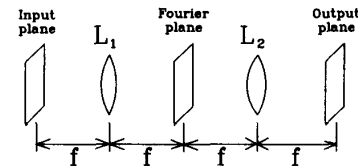


Fig. 3. Vander Lugt correlator.

Fourier plane of the system in Fig. 3 whose transmittance function is the complex conjugate of the Fourier transform of a second, reference image, then it can be shown that the output is the 2-D correlation function between the input and reference images [15]. In our system, the reference image in each of two correlators is a composite of four images that are spatially separated. A transparency is prepared containing the four images and the Fourier transform of this pattern is formed with a lens. A hologram of the Fourier transform of the composite pattern is then formed by recording its interference with a plane wave reference. The holograms that are formed in this manner are placed at the intermediate planes of two Vander Lugt correlators that are cascaded to form the optical memory loop.

The schematic diagram of the overall optical architecture is shown in Fig. 4, and a photograph of the experimental apparatus is shown in Fig. 5. The first correlator consists of the liquid crystal light valve (LCLV) at P_1 , the beam splitter

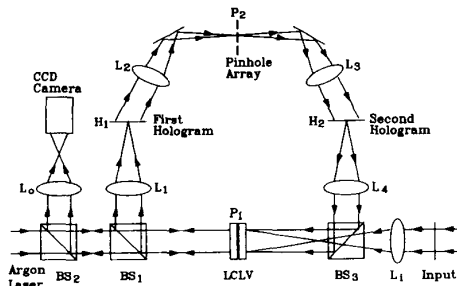


Fig. 4. Schematic diagram of the optical loop.



Fig. 5. Photograph of the optical loop.

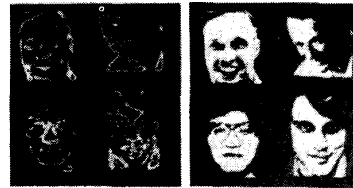
cube BS_1 , the lenses L_1 , L_2 , and the hologram H_1 . The LCLV here functions as a 2-D array of neurons. It consists of a dielectric mirror sandwiched between a light-sensing layer and a light-modulation layer. The light-sensing layer is a CdS photoconductor which acts as a photosensor, whereas the light-modulation layer is a thin layer (a few microns) of nematic liquid crystal. When light strikes the photosensor, its conductance changes, which in turn changes the voltage across the liquid crystal on the other side. When a reading light beam across the liquid crystal layer is reflected off the light-modulating side, its polarization state is modulated in proportion to the voltage across the liquid crystal layer.

The second correlator consists of P_2 , L_3 , H_2 , L_4 , BS_3 , and P_1 shown in Fig. 4. The input pattern is imaged onto the LCLV by lens L_1 and through beam splitter BS_3 . A collimated argon laser beam illuminates the read-out side of the LCLV through beam splitters BS_2 and BS_1 . A portion of the reflected light from the LCLV that propagates straight through BS_1 , is diverted by BS_2 , and it is imaged by lens L_0 onto a CCD television camera through which we monitor the system. The portion of light reflected by BS_1 into the loop is Fourier transformed by lens L_1 and illuminates hologram H_1 . The correlation between the input image and each of the stored images is produced at plane P_2 . The spacing of the pinhole array at P_2 corresponds to the spatial separation between the stored images. The remainder of the optical system from P_2 back to the neural plane P_1 is essentially a replica of the first half, with hologram H_2 storing the same set of images as H_1 .

The holograms in this system are thermoplastic plates, with an area of 1 in^2 and 800 lines/mm resolution. The hologram H_1 is made with a high-pass characteristic for edge enhancement to improve discrimination. H_2 on the other hand is broadband so that the feedback images have high fidelity with respect to the originals. We use a diffuser to



(a)



(b)

(c)

Fig. 6. Stored images. (a) The original images. (b) Images reconstructed from H_1 . (c) Images reconstructed from H_2 .

achieve this when making H_2 . Fig. 6(a) shows the four original images. Fig. 6(b) shows the images reconstructed from the first hologram H_1 , and Fig. 6(c) shows the images reconstructed from the second hologram H_2 .

The pinhole array at P_2 samples the correlation signal between the image coming from the LCLV and the images stored in hologram H_1 . The pinhole diameter used in these experiments ranges from $45 \mu\text{m}$ to $700 \mu\text{m}$. If the pinholes are too small, the light passing through to reconstruct the feedback image is too weak to be detected by the LCLV. On the other hand, large pinholes introduce excessive blurring and cross-talk and make the reconstructed images unrecognizable. The pinhole size also affects the shift invariance of the loop. In order to be recognized, the auto-correlation peak from an external image should stay within the pinhole. Larger pinholes allow more shift in the input image. The system performance under different selections of pinhole diameters is discussed in the next section. As the optical signal goes through the loop, it is attenuated because of the small diffraction efficiency of the Fourier transform holograms and the losses from pinholes, lenses, and beam splitters. To compensate for this loss, we use an image intensifier at the photoconductor side of the LCLV. The microchannel plate of the image intensifier is sensitive to a minimum incident intensity of approximately 1 nW/cm^2 and reproduces the input with an intensity 10^4 times brighter ($10 \mu\text{W/cm}^2$), sufficient to drive the LCLV. If we use a beam with intensity equal to 10 mW/cm^2 to read the LCLV, then the intensity of the output light is approximately 1 mW/cm^2 . Thus, the combination of the image intensifier and the LCLV provides an optical gain up to 10^6 . It turns out that the setting of the gain is the key parameter that mediates the trade-off between distortion invariance and the discrimination capability of the loop. This will also be discussed in the next section.

IV. EXPERIMENTAL RESULTS

The optical associative loop of Fig. 4 can be lumped into the simplified blocks shown in Fig. 7(a). The LCLV is represented as the component **Gain** in Fig. 7(a). The other parts of the loop are all lossy, linear components and are represented by the component **Loss** in Fig. 7(a).

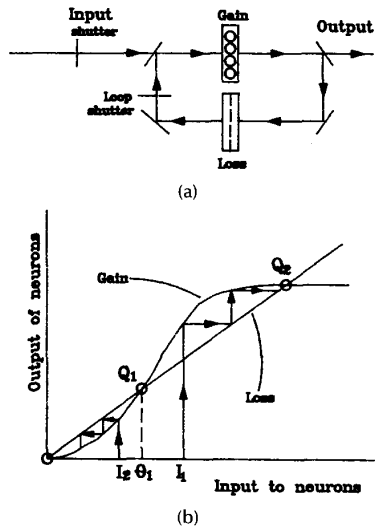


Fig. 7. (a) The gain and loss components of the loop. (b) The stable states of the loop.

The dynamics of the recall process can be understood by using the iteration map shown in Fig. 7(b). In the figure the gain curve represents the input-output response of the neurons, whereas the straight line gives the loop loss due to the holograms, and pinholes, etc. The intersection point Q_1 is the threshold level, and the intersection point Q_2 gives a stable point. If the initial condition is above the threshold level θ_1 , the signal (I_1) grows in successive iterations until it arrives and latches at Q_2 . On the other hand, if the initial condition is below θ_1 , the signal (I_2) decays to zero. The number of iterations (convergence time) depends on the initial condition. This of course is only a simplified picture of what goes on in the loop. A more precise analysis will be given in Section V.

The loop dynamics were measured by controlling the two shutters shown in Fig. 7(a). An example of the temporal response of the loop to an input pattern is shown in Fig. 8. The lower trace represents the intensity of the external input image and the upper trace represents the corresponding light intensity detected at the loop output. Before time t_1 , both shutters are closed and the response is low. At time t_1 the input shutter is opened (with the loop shutter still closed) and the lower trace becomes high. The upper trace shows the corresponding response of the neurons to the external input. At time t_2 the loop shutter is opened and the loop is closed. The feedback signal arrives at the neurons as an additional input and iteration begins. It takes about two seconds in this case for the loop to reach a stable state. At time t_3 the input shutter is closed and the lower trace becomes low. However, the loop remains latched to a stable state, which is one of the stored images. We get similar results with reduced input intensity. It takes longer to reach a stable state when the input is weak, but the final state remains the same.

Since the external input does not affect the shape of the final state, but only selects which state is produced, there is a certain degree of invariance in the system since a distorted version of a stored image can recall the stored image. The effect of distortions such as scale, rotation and shift,

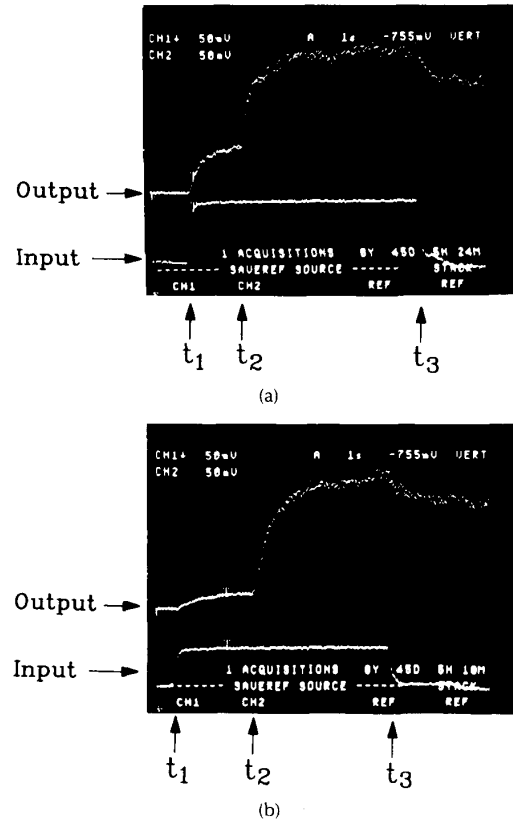


Fig. 8. Temporal response of the loop. (a) Strong input. (b) Weak input. Timing: t_1 = Input ON, t_2 = Feedback ON, t_3 = Input OFF.

is to decrease the initial intensity level of the loop. However, as long as the initial condition is above the threshold (θ in Fig. 7(b)), the loop still converges to a memory state. The strength of the initial condition is determined by the degree of distortion of the input, while the threshold is determined by the neural gain and loop loss.

The images stored in the loop are the four faces shown in Fig. 6(a). Fig. 9(a) shows the response of the system when a partially blocked face is presented to the system with the loop shutter closed. This sets the initial condition. The loop shutter is then opened to close the feedback path, and the state of the system evolves. After some time the loop reaches stable state and a complete face appears. The time for this process ranges from less than one second to several seconds, depending on the initial condition and the system parameters. The complete image remains latched in the loop when the external input is removed. Fig. 9(b) shows the system output at the moment the loop shutter is closed. We see that the feedback image is superimposed on the input. Fig. 9(c) to (e) shows the evolution of the output after the feedback loop is closed. The complete image obtained 2 seconds after the loop is closed is shown in Fig. 9(e). Fig. 9(f) shows that after the external input is removed, the recalled image stays latched. The same situation occurs when other partially blocked faces are used.

In a separate experiment, a rotated version of each of the faces was used as input. Fig. 10(a) shows the output of the

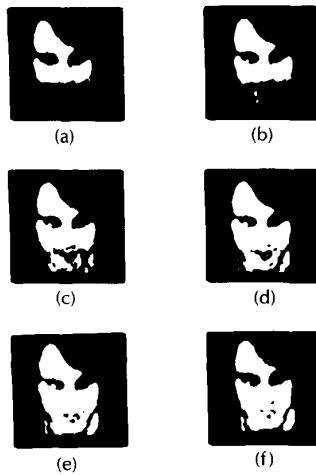


Fig. 9. Retrieval of the complete image from the partial input. (a) The partial input at $t = 0$. (b) $t = 0^+$ (loop closed). (c) $t = 400$ ms. (d) $t = 800$ ms. (e) $t = 2$ sec. (f) Input OFF.

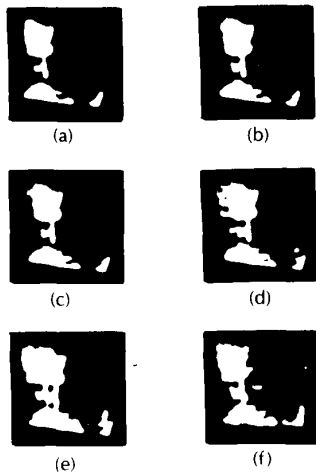
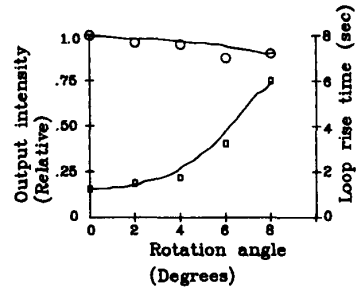


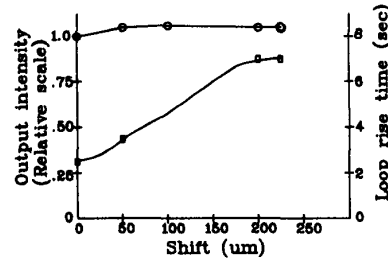
Fig. 10. Retrieval of the complete image from the rotated input. (a) The input at $t = 0$. (b) $t = 0^+$ (loop closed). (c) $t = 1.8$ sec. (d) $t = 3.6$ sec. (e) $t = 4.8$ sec. (f) Input OFF.

system when a rotated version (by 6°) of one of the faces is presented to the system with the loop shutter closed. Fig. 10(b) shows the memory output immediately after the feedback loop is closed. The evolution of the system towards the original unrotated image is shown in Fig. 10(c) to Fig. 10(e).

In Fig. 11(a) the upper curve is the intensity of the final image and the lower curve is the convergence time, both plotted as functions of rotation angle. The larger the rotation angle, the longer it takes to converge. However, once the loop reaches stable state, the output intensity is always the same regardless of the initial rotation. The output intensity drops to zero when the angle is more than 8° . This means that the initial condition is below threshold and the rotated image does not elicit a response. One way to increase tolerance to rotation is by increasing the neural gain so that it can detect weaker feedback signals. It is found that with the gain set 10 times higher, the tolerance increases to 16° .



(a)



(b)

Fig. 11. (a) Loop response to rotated inputs. (b) Loop response to shifted inputs. (Optical gain = 10^4 ; \circ : Output intensity. \square : Loop rise time.)

However, although we can obtain more tolerance by increasing the gain, this enhances crosstalk and may cause the loop to converge to the wrong image. Similar experiments were carried out to measure the ability of the system to tolerate scale changes. The results were similar, with tolerance in the range of 7% to 9%, depending on the gain.

This system has very small tolerance to position errors at the input, i.e., it is not shift invariant. When the input image is translated, the entire correlation pattern in the intermediate plane P_2 shifts also. The autocorrelation peak that is normally aligned with the pinhole is blocked. A small degree of shift invariance exists due to the finite width of the pinhole and the correlation peak. In the experimental system, the pinhole diameter was $90 \mu\text{m}$. Fig. 11(b) shows the strength of the final state and the rise time versus the amount of shift in the input from its nominal position. A larger pinhole yields more shift invariance. But as the pinhole diameter increases, the reconstruction from the second hologram is blurred because the output becomes the convolution of the stored image with its autocorrelation pattern. This results in a loss in correlation strength in subsequent iterations, and can result in insufficient gain for maintaining a stable state.

The experimental results shown above demonstrate the distortion-invariance capability of the associative loop. By raising the neural gain sufficiently high, the loop can always be made to produce an image as a stable state no matter how much we distort the input image. But the ability to reliably produce correct associations between initial and final states degrades as the gain increases. If there is too much gain, then just shining a flashlight at the input of the system can cause it to converge to one of its stable states. If the gain is set too low, the slightest distortion of the stored images renders it unrecognizable. If the gain is set even lower, no input can cause the loop to latch on a stable state. Fig. 12

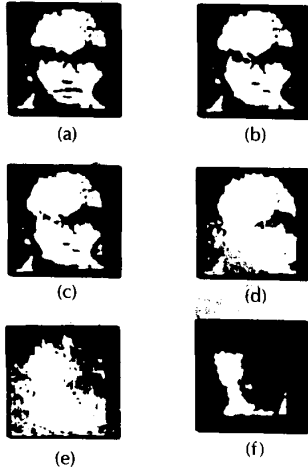


Fig. 12. Loop dynamics with high optical gain. (a) The input at $t = 0$. (b) $t = 0^+$ (loop closed). (c) $t = 1.2$ sec. (d) $t = 1.8$ sec. (e) The input is OFF. (f) Stable state.

shows an example of the behavior obtained when the gain is set too high. An unfamiliar input initially produces an unrecognizable state. When the external input is removed, the system latches erroneously to one of the stored images.

V. A NETWORK MODEL FOR THE OPTICAL LOOP

The optical associative memory presented in this paper is very similar to a Hopfield network, but it is not quite the same. The neurons are simulated by the LCLV, which responds to light intensity, which is the magnitude squared of the light amplitude, the quantity that is modulated by the output stage of the LCLV and multiplied by the weights. Consequently, the input signal to the neurons is first squared before being thresholded and as a result the neural gain is unipolar instead of bipolar (as in the Hopfield model). The interconnection weights are bipolar quantities (actually they can be complex since they are holographic gratings). As mentioned before, the first correlator contains a high pass version of the stored memories. In our analysis we will assume that the high pass filtering operation subtracts the mean value of each image, thus transforming the unipolar initial images into their bipolar versions. Since in the optical system the outer product needed to specify the interconnection matrix is formed as a cascade of two correlators, the resulting weight is the outer product between bipolar and unipolar versions of the stored images. These differences give us characteristics that are distinct from the Hopfield model, such as a ground state (which the Hopfield model does not have) and a dependence of the stable states on the gain the system (in the optical system an increase in gain transforms the shape of the stable states). Thus although the Hopfield model has been analyzed extensively, all the results can not be applied directly to our system, and further analysis is necessary.

In the following, we present a model for the optical associative-memory loop described above. For simplicity we will revert in the subsequent analysis to 1-D, discrete notation. Let x_i^m denote the i -th bit of the m th unipolar (0 or 1) memory. The interconnection matrix that is implemented by the

optics is

$$w_{ij} = \sum_{m=1}^M (x_j^m - a_m)x_i^m, \quad (7)$$

where $a_m = 1/N \sum_{j=1}^N x_j^m$ is the mean value of the m th memory. A neuron is optically simulated by one pixel of the LCLV, which is modeled as shown in Fig. 13. Let x_i denote the out-

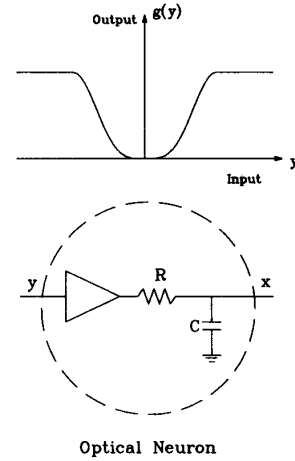


Fig. 13. Model for the LCLV and the gain function.

put of the i th neuron (or equivalently the reflectivity of a pixel on the LCLV). Then

$$\frac{dx_i}{dt} = -x_i + g\left(\sum_{j=1}^N w_{ij}x_j\right), \quad i = 1 \cdots N \quad (8)$$

where g is a nonlinear function describing the neuron response. Note that g is an even function instead of an odd (sigmoid-like) function, as is usually assumed.

In the above equation the response of each pixel is coupled to all other pixels and depends on all the memories stored in w_{ij} . To gain some understanding about what the stable states are and how they are related to the patterns we attempt to store in the system, we introduce a change of variables.

Assume that the stored images x^1, x^2, \dots, x^M are linearly independent. We decompose the vector space R^N into two subspaces, V_1 and V_2 , where V_1 is the vector space spanned by the stored images and V_2 is normal to V_1 . We define a basis $\beta_1 = \{y^1, y^2, \dots, y^M\}$ for V_1 , such that

$$y^i \cdot x^j = \delta_{ij} \quad i, j = 1, \dots, M \quad (9)$$

and we select an orthonormal basis $\beta_2 = \{y^{M+1}, \dots, y^N\}$ for V_2 . We then have $\beta = \beta_1 \cup \beta_2 = \{y^1, \dots, y^N\}$ which forms a basis for R^N .

Consider the vector u whose j th component is $(x_j - 1/N \sum_{k=1}^N x_k)$. (We will call u the bipolar version of x .) Let c_l be the l th component of u expanded in terms of the basis β . Using (9), a set of differential equations can be found for the c_l 's:

$$\frac{dc_l}{dt} = -c_l + \sum_{i=1}^N (x_i^l - a_l)g\left(\sum_{m=1}^M c_m x_i^m\right), \quad l = 1, \dots, M, \quad (10)$$

$$\frac{dc_l}{dt} = -c_l + \sum_{i=1}^N (y_i^l - b_l)g\left(\sum_{m=1}^M c_m x_i^m\right),$$

$$l = M + 1, \dots, N \quad (11)$$

where

$$b_l = \frac{1}{N} \sum_{j=1}^N y_j^l. \quad (12)$$

We also have

$$\sum_{j=1}^N w_{ij} x_j = \sum_{m=1}^M c_m x_i^m. \quad (13)$$

In general, when the state of the system approaches the l th stored pattern, the variable c_l becomes large. In particular, if the stored memories x^m were orthogonal (no overlapping) then c^l would simply be the inner product between the bipolar version u of the state of the system and the l th stored memory. The l th equation of (10) has a driving force which is the inner product between the bipolar version of the l th memory and the output of the neurons. If the state of the system starts to approach one of the stored memories then the corresponding c^l will tend to grow thus providing the system with a tendency to be attracted to that state. Setting $d/dt = 0$ in (9) and using (10), we get the following expression for the equilibrium states of the system:

$$x_i = g\left(\sum_{m=1}^M c_m x_i^m\right). \quad (14)$$

If one of the c_m 's becomes dominant, then the stable state of the system resembles the corresponding stored pattern. Note, however, that the neurons will also pick up some cross-correlation components. This is a property observed in our experiments where increase of the gain led to the distortion of the stable states and eventually to the creation of unrecognizable mixture states.

Note that c_1, \dots, c_M are coupled together in (10), but the driving terms of the equations for c_{M+1}, \dots, c_N (11) depend only on c_1, \dots, c_M . Thus once the steady states of c_1, \dots, c_M are known, so will those of c_{M+1}, \dots, c_N . Therefore we need only consider the l coupled equations in (10).

To gain some insight of how the system behaves, we introduce a geometrical method to illustrate how the system evolves to a stable state, and how it is influenced by the parameters such as gain and initial conditions. In order to illustrate the concept, we will consider the case where only two images, x^1 and x^2 , are stored in the memory. As we shall see, the two-image case contains all the salient features of the dynamics. Similar arguments can be made to extend the results to the general case of multiple stored images.

Equation (10) is reduced to two equations:

$$\frac{dc_1}{dt} = -c_1 + \sum_{i=1}^N (x_i^1 - a_1)g(c_1 x_i^1 + c_2 x_i^2) \quad (15)$$

$$\frac{dc_2}{dt} = -c_2 + \sum_{i=1}^N (x_i^2 - a_2)g(c_1 x_i^1 + c_2 x_i^2). \quad (16)$$

Recall that a_1 and a_2 are the average levels of the input images x^1 and x^2 . Let $h_1(c_1, c_2)$ represent the driving term in (15), and $h_2(c_1, c_2)$ the summation term in (16). For simplicity, assume that x^1 and x^2 have no overlapping nonzero components. Thus x_i^1 can be nonzero only when $x_i^2 = 0$, and vice versa.

In this case, the driving forces can be written as

$$h_1(c_1, c_2) = \sum_{x_i^1 \neq 0} (x_i^1 - a_1)g(c_1 x_i^1) - a_1 \sum_{x_i^2 \neq 0} g(c_2 x_i^2) \quad (17)$$

$$h_2(c_1, c_2) = \sum_{x_i^2 \neq 0} (x_i^2 - a_2)g(c_2 x_i^2) - a_2 \sum_{x_i^1 \neq 0} g(c_1 x_i^1) \quad (18)$$

There are two parts in each of the driving forces. Consider $h_1(c_1, c_2)$. The first term comes from the correlation between the neuron state $g(c_1 x^1)$ and the bipolar version of stored image x^1 , and the second term results from the coupling between c_1 and c_2 through the dc level a_1 . Since a_1 and the gain function $g(x)$ are always positive, the second term gives a negative contribution to the driving force. This means that the coupling pulls the system away from x^1 . The same description also applies to c_2 . We plot $h_1(c_1, c_2)$ against c_1 for $c_2 = 0$ and $c_2 \neq 0$ in Fig. 14(a).

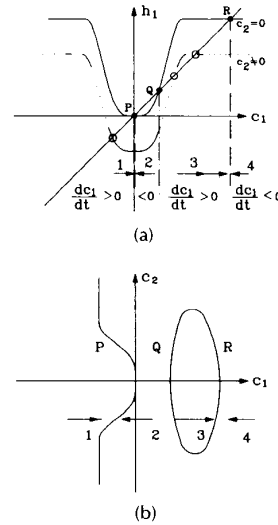


Fig. 14. The driving force and the dynamics of the loop. (a) The driving force for the first stored image. (b) The boundary lines of the equilibrium states of the first image.

In the figure, the solid curve represents the case $c_2 = 0$, and the dashed curve represents the case for $c_2 \neq 0$. We also plot the line $h(c_1) = c_1$ in the same figure. It is seen that there are three intersections, P , Q , and R , between the straight line and the solid curve $h_1(c_1, c_2)$ (for $c_2 = 0$). As we increase the value of c_2 from 0 to a positive or negative value, the curve $h_1(c_1, c_2)$ changes due to the second summation term in (18), as shown in Fig. 14(a). The intersection points P , Q , and R then also change. As c_2 increases, the points Q , and R will typically merge together and then vanish. If we plot out the values of c_1 corresponding to P , Q , and R for different c_2 values in the (c_1, c_2) plane, they will trace out three curves. An example is shown in Fig. 14(b) (here the curves by Q , and R merge to become a closed loop). We will call these the boundary lines (of c_1).

Consider the three intersection points for a particular c_2 as shown in Fig. 14(a). The c_1 axis is divided into four regions, designated as 1 to 4. In regions 1 and 3, c_1 is smaller than $h_1(c_1, c_2)$ and $dc_1/dt > 0$. Thus, in these regions the system state evolves in the direction of increasing c_1 . This is rep-

represented by the arrows pointing to the right in the figure. On the other hand, in regions 2 and 4, $dc_1/dt < 0$; thus, the system evolves toward decreasing c_1 . The corresponding situation is shown in the (c_1, c_2) plane in Fig. 14(b). Here the regions 2 and 4 merge. The arrows again denote the direction that c_1 changes.

By going through the same procedure, we can obtain similar boundary lines for c_2 . We then plot the two groups of boundary lines in the same (c_1, c_2) plane to obtain the phase diagram for (15) and (16). An example is shown in Fig. 15.

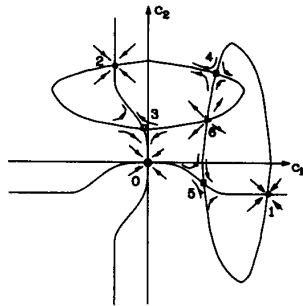


Fig. 15. Phase flow of the two-image auto-associative memory. States 0, 1, 2 are stable. States 3, 4, 5 are unstable (saddle points). State 6: Source state. (unstable)

We see that there are 7 equilibrium points; one source, three sinks, and three saddles. The three sinks represent the null state (no image) and the two stored images. Point 1 represents the stable state corresponding to stored image x^1 , since at that position c_1 is large and c_2 is small. On the other hand, at point 2 c_1 is small and c_2 is large, and this represents the stable state corresponding to stored image x^2 . It can be seen from the figure that if we start from an initial state that is close to one of the stored states, the system will converge to that state. Otherwise, it will decay to zero.

From the phase diagram we see that the stable state is always a mixture state of the stored memories. The extent of mixture can be reduced by reducing the neural gain. However, if the gain is too small, then the system will not be able to sustain the stored memories. To see why this is so, consider the three intersection points in Fig. 14(a). Reducing the gain shrinks the $h_1(c_1, c_2)$ curve, so that the points Q , and R merge at a lower value of c_2 in Fig. 14(b). If we lower the gain further, the intersection points Q , and R will disappear altogether, and there will be only one boundary line in Fig. 14(b). Thus in Fig. 15, as the gain decreases, the intersection points 4 and 6 will first disappear as the two closed loops (boundary lines) shrink. As the gain decreases further, the loops disappear, and with it the intersection points 1, 2, 3, and 5. In this case, there will be only one equilibrium state, viz., the null state at the origin. No matter where the initial state is, the system always decays to zero.

On the other hand, suppose the neural gain is set very high. In this case the loops in Fig. 15 will become larger, and two more equilibria points can appear, as shown in Fig. 16. The state m is a strongly mixed state of x^1 and x^2 . We also see that m has a large region of attraction. Thus it is important that the gain is not set too high.

Next consider the case where the stored memories have

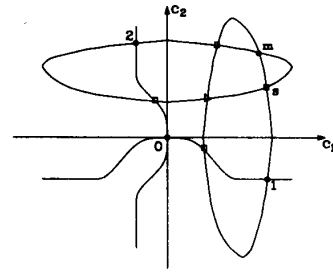


Fig. 16. The dynamics of the loop at high gain. Two new equilibrium states are generated: M is a mixture state and s is a saddle point.

some slight overlap. In principle we can still plot out corresponding boundary lines for this case. The shape and position of these lines will be altered somewhat from the nonoverlapping case. However, since the neural gain function is continuous the general features of the system will be the same if the overlap is small. As the overlap between the stored states increases, the boundary lines in the phase diagram become distorted. In computer simulations, the stable points that ought to resemble the vectors we attempt to store in the memory, become a mixture of all the stored states and the system performance degrades. We do not yet have a prediction for the amount of overlap that is tolerated in this system.

It is interesting to use the method described above to investigate the effect of using an all-pass hologram instead of a high-pass hologram in the first correlator of our system. Note that in this case, the interconnection matrix w_{ij} will be symmetric. We expand here x instead of u , and consider the components c_i of x expanded in basis β . Equations (15) and (16) then become

$$\frac{dc_1}{dt} = -c_1 + \sum_{i=1}^N x_i^1 g(c_1 x_i^1 + c_2 x_i^2) \quad (19)$$

$$\frac{dc_2}{dt} = -c_2 + \sum_{i=1}^N x_i^2 g(c_1 x_i^1 + c_2 x_i^2) \quad (20)$$

By going through similar arguments, we can draw the boundary lines and the phase diagram for this system. Fig. 17 shows one example. It is seen that there are four stable states: two memory states, 1 and 2, one null state 0, and one mixture state m . If we decrease the neural gain, then the points 1, 2, s_1 , and s_2 , may disappear. However, the mixture state m always exists. This shows why a high-pass hologram

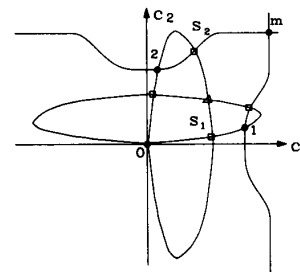
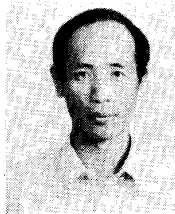


Fig. 17. The dynamics of the loop without the high pass hologram. There are four stable states: 1 and 2 are the stored states, 0 is the null state. m is a mixture state. The other states are unstable.

is crucial for good performance of the memory loop, a fact confirmed by the experimental system.

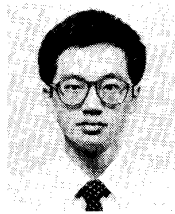
REFERENCES

- [1] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci. USA*, vol. 79, pp. 2554-2558, Apr. 1982.
- [2] J. A. Anderson, J. W. Silverstein, S. A. Ritz, and R. S. Jones, "Distinctive features, categorical perception, and probability learning: some applications of a neural model," *Psychological Review*, vol. 84, pp. 413-451, 1977.
- [3] D. Psaltis and N. Farhat, "Optical information processing based on an associative-memory model of neural nets with thresholding and feedback," *Opt. Lett.*, vol. 10, pp. 98-100, 1985.
- [4] N. H. Farhat, D. Psaltis, A. Prata, and E. Paek, "Optical implementation of the Hopfield Model," *Appl. Opt.*, vol. 24, pp. 1469-1475, 1985.
- [5] Y. S. Abu-Mostafa and D. Psaltis, "Optical neural computers," *Scientific American*, vol. 256, no. 3, pp. 88-95, 1987.
- [6] E. G. Paek and D. Psaltis, "Optical associative memory using Fourier transform holograms," *Opt. Eng.*, vol. 26, no. 5, pp. 428-433, May 1987.
- [7] J. S. Jang, S. W. Jung, S. Y. Lee, and S. Y. Shin, "Optical implementation of the Hopfield model for two-dimensional associative memory," *Opt. Lett.*, vol. 13, pp. 248-250, 1988.
- [8] J. S. Jang, S. Y. Shin, and S. Y. Lee, "Optical implementation of quadratic associative memory with outer-product storage," *Opt. Lett.*, vol. 13, pp. 693-695, 1988.
- [9] N. H. Farhat and D. Psaltis, "Optical implementation of associative memory based on models of neural networks," in *Optical Signal Processing* Chap. 2.3, J. L. Horner, Ed. San Diego, CA: Academic Press, 1987.
- [10] L. S. Lee, H. M. Stoll, and M. C. Tackitt, "Continuous-time optical neural associative memory," *Opt. Lett.*, vol. 14, p. 162, 1989.
- [11] B. H. Soffer, G. J. Dunning, Y. Owechko, and E. Maron, "Associative holographic memory with feedback using phase-conjugate mirrors," *Opt. Lett.*, vol. 11, pp. 118-120, 1986.
- [12] D. Psaltis, D. Brady, and K. Wagner, "Adaptive optical networks using photorefractive crystals," *Appl. Opt.*, vol. 27, pp. 1752-1759, 1988.
- [13] D. Psaltis, D. Brady, X.-G. Gu, and S. Lin, "Holography in artificial neural networks," *Nature*, vol. 343, no. 25, pp. 325-330, 1990.
- [14] A. B. Vander Lugt, "Signal Detection by Complex Spatial Filtering," *IEEE Trans. Inform. Theory*, vol. IT-10, no. 2, pp. 139-145, 1964.
- [15] J. W. Goodman, *Introduction to Fourier Optics*, Chap. 2, New York: McGraw Hill, 1968.



Ken-Yuh Hsu received the B.S. degree in Electrophysics in 1972 and the M.S. degree in Electrical Engineering in 1975, all from the National Chiao-Tung University, Taiwan, R.O.C. He received his Ph.D. degree in Electrical Engineering in 1989 from the California Institute of Technology, Pasadena, California.

After completion of the M.S., he remained at Chiao-Tung University as Lecturer at the Department of Electrophysics from 1975 to 1984. He came to the California Institute of Technology to study as a Ph.D. student in 1984. After obtaining his degree in 1989, he returned to Taiwan, and is now a faculty member of the Graduate Institute of Electro-optical Engineering at Chiao-Tung University.



Hsin-Yu Li received the B.S. degree in Communications in 1984, and the M.S. degree in Electro-optics in 1986, all from the National Chiao-Tung University, Taiwan, R.O.C. He is now a Ph.D. student at the California Institute of Technology.



Demetri Psaltis (Member, IEEE) received the B.Sc. in electrical engineering and economics in 1974 and the M.Sc. and Ph.D. degrees in electrical engineering in 1975 and 1977, respectively, all from Carnegie-Mellon University, Pittsburgh, PA.

After the completion of the Ph.D., he remained at Carnegie-Mellon, as a Research Associate and later as a Visiting Assistant Professor, for a period of three years. In 1980, he joined the faculty at the California Institute of Technology, Pasadena, where he is now Professor of Electrical Engineering and consultant to industry. His research interests are in the areas of optical information processing, holography, radar imaging, pattern recognition, neural networks, optical memories, and optical devices. He has authored or co-authored over 170 publications in these areas.

Dr. Psaltis is a fellow of the Optical Society of America and received the International Commission of Optics Prize in 1989.