Implementation of Optical Neural Networks Based on Parallel Rank-One Interconnections and Time Integration

Ho-In Jeon, Jin-San Choi, Chang-Min Shin, Yo-Sik Kang, and Nak-Hee Jung

Department of Electronic Engineering
Kyung Won University
San 65, Bok-Jung-Dong, Soo-Jung-Gu Sung-Nam-Shi, Kyung-Gi-Do

Abstract

An optical implementation of higher order neural networks based on principal component analysis and time integration has been described. The principal component analysis combined with time integration allows larger input size than fully spatial neural networks at the cost of certain amount of time consumption. This time-integration usage actually breaks down the barrier of the maximum space-bandwidth product that optical systems can offer.

1. Introduction

Higher order neural networks are a useful way of expanding a data set and thereby utilizing that data better-especially in a single layer network [1]. The higher order neural networks have demonstrated dramatic improvement of its storage capacity and the noise immunity [1]. The problem is that even a second order neural network requires as large as $O(N^3)$ interconnection networks to store one-dimensional vectors of N elements. Consequently, this limits the size of the input pattern. One way of solving this problem is to make the size of the memory matrix the same as that of input patterns and utilize the time domain to implement the recall process. Principal component analysis combined with time integration [2,3] can be used to implement this idea. The purpose of this paper is to show that combining optics with principal component analysis and time integration allows us to operate an optical two-dimensional any higher order neural networks, with the physical size of the memory space the same as that of input vectors. This permits increased dimensionality of the input vectors. The system we propose to use is based on the principal component analysis which provides practical implementation of parallel rank-one N^4 interconnection system discussed earlier [2,3].

In a Hopfield-like second-order neural network, we store a set of M binary patterns V_{ij}^s ($s = 1, 2, \cdots, M$ and $i, j = 1, 2, \cdots, N$) using the sixth order matrix (tensor)

$$W_{ijklmn} = \sum_{s=1}^{M}(2V_{ij}^s - 1)(2V_{kl}^s - 1)(2V_{mn}^s - 1), \tag{1}$$

where the input is assumed to be the unipolar binary [1, 0] and W_{ijklmn} is a sixth-rank tensor which plays the role of recalling the stored information from distorted input pattern. The output pattern y_{ij} for this memory can be retrieved by performing

$$y_{ij} = \sum_{k,l}^{N}\sum_{m,n}^{N} W_{ijklmn}\, x_{mn}\, x_{kl}, \tag{2}$$

and thresholding the output with predetermined threshold levels, and feeding it back into the input until convergence is achieved. It is assumed that we are using the same 2D $\mathbf{x}$ vector for both $\mathbf{x}$ terms.

To implement the second order neural networks based on principal component analysis, the optical system shown in Fig. 1 can be directly used, where, the time integration for k and i implements the second order neural network. The advantage of this time integration usage is that, instead of the space domain, we can utilize the time domain to overcome the size limitation of the

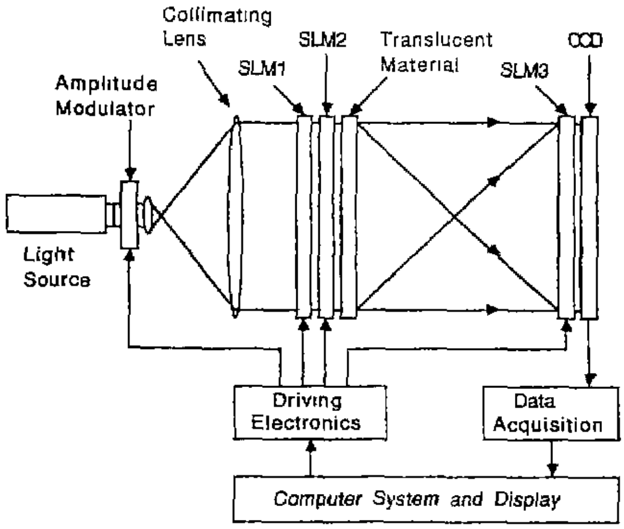

Figure 1: Schematic diagram of massively parallel interconnections for optical neural networks. The amplitude modulator, SLM1, SLM2, and SLM3 implement the singular values s_i's, input vectors, eigenvectors $\mathbf{v}^T$, and $\mathbf{u}$, respectively.

optical system. Moreover, according to the principal component analysis, we may use less number of principal components than $k \times r$ at the expense of very little degradation of the system performance. This gives extra time saving, and thus provides better system performance than conventional higher order neural networks.

For computer simulations, we first encoded $M = 6$ characters "C", "A", "O", "T", "F", and "S" with 10×10 binary pixels. Then these 6 binary patterns are stored in the memory matrix W based on outer-product learning rule. Very interestingly, all the rearranged submatrices W_i's are of rank $r = M$. So, it is easy to see that the rank r approximation of the submatrices W_i produces the exactly same results as the original Hopfield neural network does. Moreover, the set of singular values and the corresponding eigenvectors of W_i's showed a highly structured appearance, which can reduce the reconfiguration time of the overall system. In our simulation, only 20 different sets out of 100 were repeated. Therefore, about 80 % of the overall processing time could be saved. This phenomenon is readily understandable in terms of object redundancies.

If we utilize time as an additional domain, we can implement any higher order neural networks with less space but more time expenditure. The second-order neural network with principal component analysis allows the size of the input pattern up to $N = 1,000$ at the cost of $r \times N^2$ reconfiguration time for the time integration. The increase of the size of the input pattern is obtained because the size of the memory space is same as that of the input. Moreover, the repeated appearance of the same set of the singular values and the corresponding eigenvectors allows us to save about 80 % of the overall processing time. The system is, of course, adaptive because we use real-time SLMs for the implementations of the $\mathbf{u}$'s and $\mathbf{v}$'s.

References

[1] C. Lee Giles et al., "High Order Correlation Model for Associative Memory," *AIP Proc. Neural Networks for Computing*, Edited by J. S. Denker, Vol. 151, pp. 86-99, Snow Bird, Utah, 1986.

[2] H. J. Caulfield, "Variable and Fixed Rank 1 N^4 Interconnections," *Proc. SPIE, Optical Computing*, Vol. 963, pp. 564-569, 1988.

[3] H. J. Caulfield, Ho-In Jeon, J. Brown, and Paul Werbos, "Variable and Fixed Rank-One Interconnections," *Applied Optics*, Vol. 29, pp. 2019-2023, January, 1990.