

Optical finite impulse response neural networks

Paulo E. X. Silveira and Kelvin H. Wagner

Optoelectronic Computing Systems Center
Department of Electrical and Computer Engineering
University of Colorado, Boulder, CO 80309-0525

ABSTRACT

We present a time and space integrating optical architecture for multi-layer finite impulse response neural networks (FIRNN). The proposed architecture is capable of forward propagation and on-line learning in the form of backward propagation. FIRNNs are first presented and analyzed. From the analysis it is observed that the implementation of FIRNNs requires the calculation of temporal convolutions, which inspire the use of time-integrating and space-integrating optical architectures. A novel device is proposed for the space-integrating architecture, based on the use of a rotating volume hologram. Initially, two single-layer architectures based on the space integrating and time integrating architectures are presented, leading to the multi-layer architecture, which uses a combination of both architectures, folding them together in such a way that all the operations of the order $O(N^3)$ are performed optically and only the less computationally intensive operations are performed electronically.

Keywords: finite impulse response neural networks, optical processing, time-integrating, space-integrating.

1. INTRODUCTION

Finite impulse response neural networks (FIRNN), with their capacity for temporal processing, add new capabilities to adaptive neural networks. This added capacity requires higher computational complexity. It is possible, however, to use three-dimensional optical processing in volume holograms in order to perform the computation in the temporal dimension simultaneously with the spatial dimensions and thereby achieve high throughput real-time operation.

The purpose of this paper is to explore optical implementations of FIRNNs making use of a combination of space integrating (SI) and time integrating (TI) architectures. It is organized as follows: Section 1 provides a description of finite impulse response (FIR) filters and neural networks, as an introduction to FIRNNs. Section 2 analyzes FIRNNs and derives the equations describing their operation in forward and backward propagation. Section 3 describes the SI and TI approaches to the spatio-temporal weighted sums and explains how they can be applied to the implementation of FIRNNs. Two single-layer network architectures are first presented, followed by an architecture for multi-layer FIRNNs capable of back-propagation learning. Section 4 concludes the paper.

Let us start with a brief introduction to finite impulse response (FIR) filters and neural-networks. FIR adaptive filters are commonly used in signal enhancement, channel equalization, system identification, signal prediction and channel modeling problems.¹ They consist of one single input node which is time-delayed over a tapped-delay line (see Fig. 1). The filter output is produced from the weighted sum of delayed versions of the input signal, where the summation weights are the adaptive parameters. FIRs can be combined in order to form an adaptive array, where several input nodes are summed in order to produce the output, as commonly used in adaptive phased array processing.² Traditionally, the output is subtracted from a desired signal producing an error signal and the weights are adapted in order to reduce the mean squared error (MSE). The *gradient descent* algorithm is commonly used for this goal, according to which each individual weight follows the gradient of the MSE with respect to itself. The least mean squares (LMS) algorithm^{2,3} is a simplification of the gradient descent algorithm, which requires less computation, achieving a complexity of the order $O(NT)$, where N is the number of input nodes and T is the number of time delays. Adaptive FIRs have been previously optically implemented using time integrating and space integrating architectures.⁴⁻⁶

Traditional neural networks are composed of several interconnected layers. In each layer the input nodes are connected to the output nodes through a weighted sum, where the weights are the adaptive parameters of the

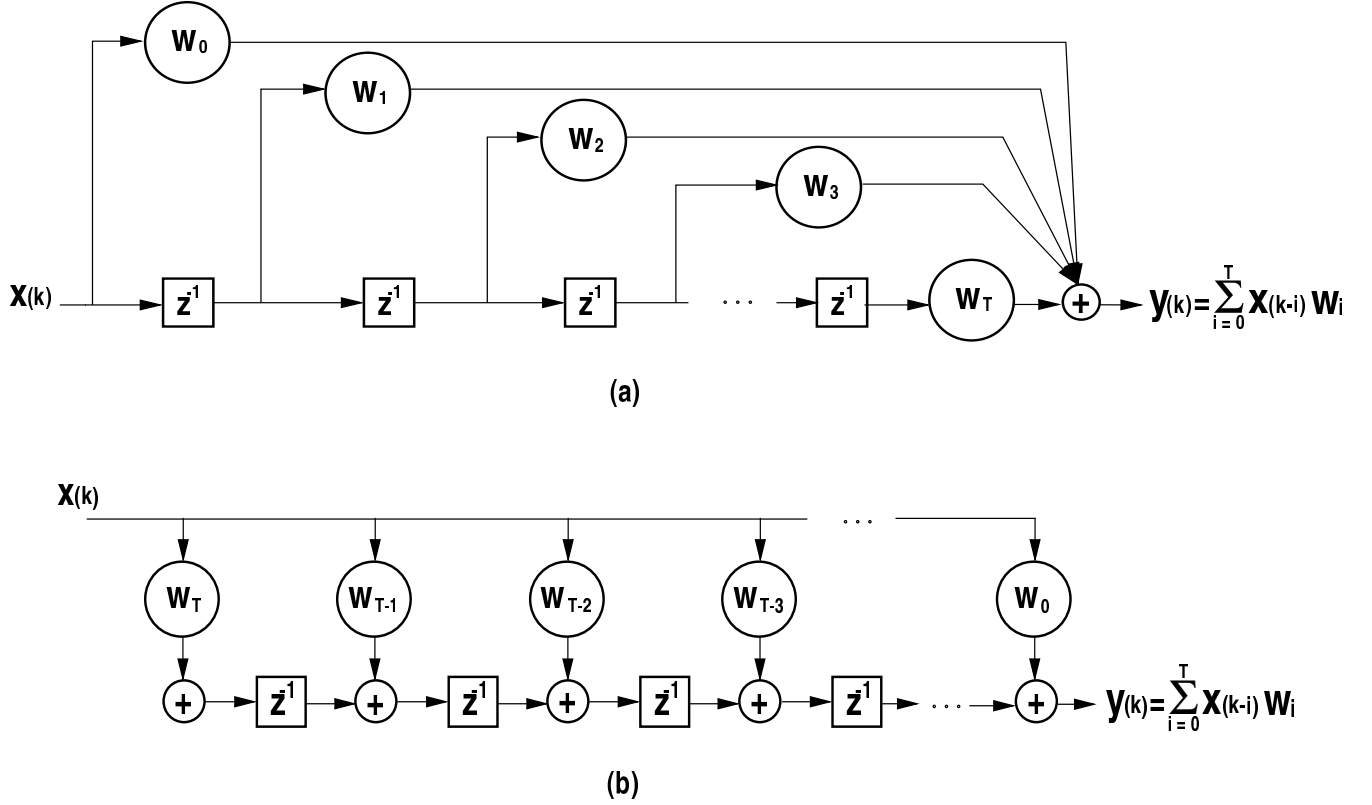


Figure 1. Two representations of a FIR filter. (a) - Space-integrating version, in which the weights are delayed and then summed. (b) Time-integrating version, in which the weights are summed as they are delayed. Both versions perform functionally equivalent operations.

network, and a non-linear operation is performed at each output node, which usually is a sigmoid for classification problems⁷ or a gaussian for functional approximation.⁸ The calculation of the final outputs as the inputs are propagated through layers is called *forward propagation*, and it requires $\sum_l N^l N^{l+1}$ operations, where N^l is the number of nodes present at layer l . The operation of modifying the interconnection weights in order to cause the network to perform a desired function is called *learning*. In *supervised learning* sets of previously known input-output pairs are used to modify the weights. In *on-line learning* weight updates take place simultaneously with forward propagation. Different paradigms exist for weight adaptation, but a common one is the same gradient descent used in FIRs. An efficient form of adapting weights in individual layers with respect to the output in the final layer is called *back-propagation learning*,⁹ which also requires $\sum_l N^l N^{l+1}$ operations in traditional neural networks. Several optical implementations of neural networks have been proposed in the past.¹⁰⁻¹³ The reasoning behind the advantages of implementing traditional neural networks optically resides in its capacity of performing high throughput global interconnects and of processing information with a high degree of parallelism, being capable of simultaneously performing all operations required for forward propagation and backward propagation.

A finite impulse response neural network (FIRNN) is a neural network whose interconnects are composed of finite impulse response (FIR) adaptive filters¹⁴ (see Fig. 2). Alternatively, it can be thought of as a multi-layer adaptive array in which the outputs of each individual layer undergoes a non-linear operation. Besides all the applications normally attributed to FIRs and neural networks, FIRNNs are also used for multi-dimensional time-series prediction,¹⁵ real-time speech recognition¹⁶ and any applications that require the multi-dimensional cross-correlation of signals. Because of the added temporal dimension, their order of complexity increases to $O(N^2 T)$ (see section 2), making their optical implementation even more attractive than that of traditional neural-networks. Architectures capable of performing on-line learning using the efficient back-propagation algorithm are of special interest.

A previous FIRNN optical architecture used a rotating photorefractive crystal (PRC) delay line in order to im-

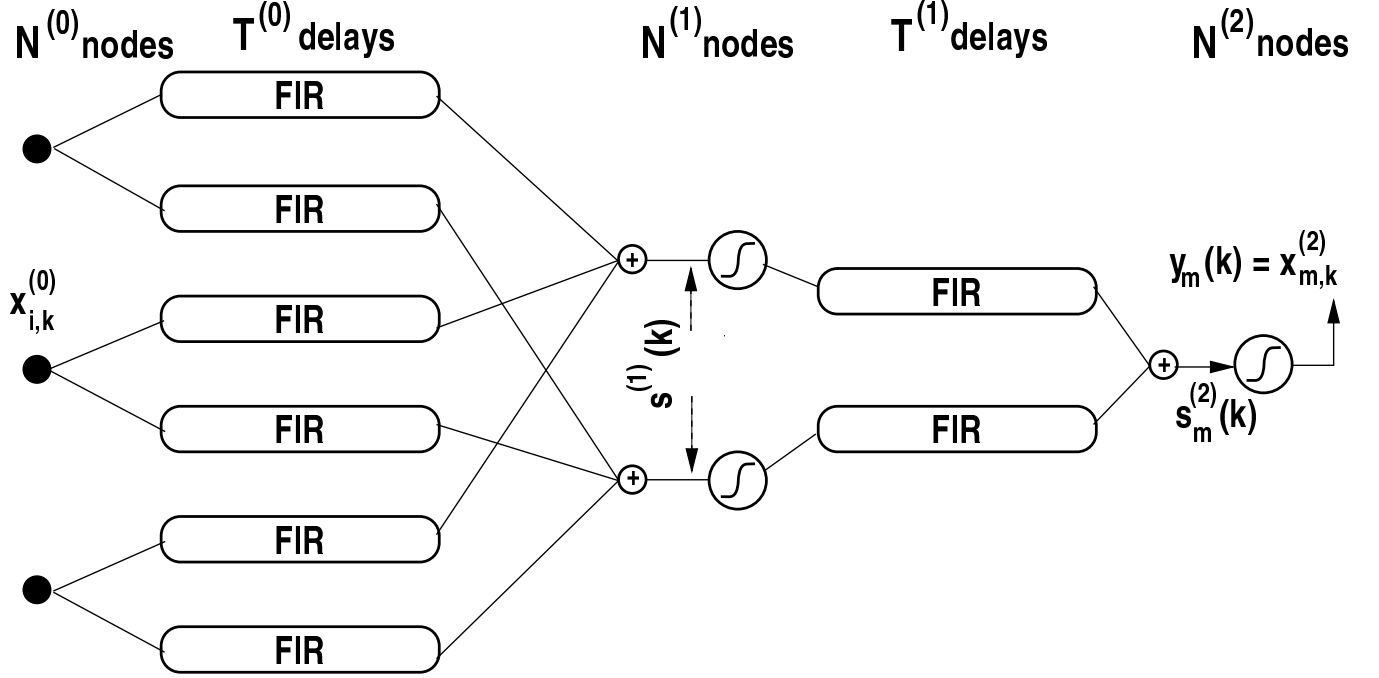


Figure 2. A FIRNN depicted as a neural network in which its interconnects are composed by FIR filters. A sigmoidal non-linear function is represented in this case.

plement a single-layer FIRNN with an associative learning algorithm.¹⁷ Another used the TI optical architecture with acoustooptic devices (AOD) and charge-coupled devices (CCD) in order to implement a multi-layer non-linear FIRNN, but which was incapable of on-line learning.¹⁸

2. FIRNN ANALYSIS

We will start our analysis by describing the signals in a FIRNN in forward propagation. Then we will define a *desired signal* which will be used during supervised learning of the network and an error performance measure which will allow us to implement weight adaptation in the form of a back-propagation algorithm modified in order to account for the causal time delays in the network.¹⁴

Let $x_{i,\tau}^l$ be the input to node i in layer l at the time index k (a comma is used to separate spatial and temporal indices), and $w_{ji,\tau}^l(k)$ represent the weight interconnecting node i to node j at layer l , time index τ at time step k . The sum of inputs to node j before the non-linear transformation is given by the total output of each one of the FIR filters arriving at node j :

$$s_j^{l+1}(k) = \sum_{i=1}^{N^l} \sum_{\tau=0}^{T^l} w_{ji,\tau}^l(k) x_{i,k-\tau}^l, \quad (1)$$

where T^l is the total number of delays at layer l and N^l is the total number of nodes at layer l , where $0 \leq l \leq L$, where L represents the total number of layers. After the non-linear operation $f(\cdot)$ we have the output of layer l given by

$$\begin{aligned} x_{j,k}^{l+1} &= f \left(\sum_{i=1}^{N^l} \sum_{\tau=0}^{T^l} w_{ji,\tau}^l(k) x_{i,k-\tau}^l \right) \\ \underline{x}^{l+1}(k) &= f(\underline{\underline{w}}^l : \underline{\underline{x}}^l(k)) \end{aligned} \quad (2)$$

where an under bar indicates a spatial dimension, an over bar indicates a temporal dimension and the dyad is assumed to be over the temporal dimension and the spatial dimension which $\underline{\underline{w}}$ and $\underline{\underline{x}}$ have in common (represented by indices

i and τ). Notice that equation 2 describes a spatio-temporal weighted sum followed by a non-linear function, where delayed versions of $\underline{x}^l(k)$ are used, such that a temporal convolution between $\underline{w}$ and $\underline{x}$ is performed. This operation consists of the multiplication and summation of terms of a $N \times T$ matrix with a $N \times 1$ vector, such that a total of $\sum_l N^l N^{l+1} T^l$ operations are performed at each time step.

Since there are L layers in the network, $\underline{x}^L(k)$ represents a vector of network outputs. Then let $\underline{e}(k) = \underline{d}(k) - \underline{x}^L(k)$ be the error vector at time step k , where $\underline{d}(k)$ represents the desired output at time step k . We then define a cost function C as

$$C = \sum_{k=1}^K \|\underline{e}(k)\|^2, \quad (3)$$

According to the gradient descent learning rule, weights are modified as

$$w_{ji,\tau}^l(k+1) = w_{ji,\tau}^l(k) - \eta \frac{\partial C}{\partial w_{ji,\tau}^l}, \quad (4)$$

where η is the learning rate. Following the traditional back-propagation algorithm, a delta term is defined as

$$\delta_j^l(k) = \frac{\partial C}{\partial s_j^l(k)}. \quad (5)$$

Applying the chain rule of derivatives, substituting terms and indices we arrive at the following weight update equation,

$$w_{ji,\tau}^{L-1-n}(k+1) = w_{ji,\tau}^{L-1-n}(k) - \eta \delta_j^{L-n}(k - \Upsilon^n) x_{i,k-\Upsilon^n-\tau}^{L-1-n}, \quad (6)$$

which consists of a spatial-temporal outer-product between a delayed version of the input $x_{i,k-\Upsilon^n-\tau}^{L-1-n}$ and the back-propagated error term $\delta_j^{L-n}(k - \Upsilon^n)$. The outer-product consist of the multiplication of terms of a $N \times T$ matrix with a $N \times 1$ vector. Therefore, it is also an operation of the order $O(N^2 T)$. Two variables were introduced in order to account for causality: n and Υ . n is an auxiliary index varying from 0 to $L-1$ and, Υ is the sum of time delays from the present layer to the network output, defined by the recursive formula:

$$\Upsilon^n = \begin{cases} 0 & n = 0 \\ \Upsilon^{n-1} + T^{L-n} & 0 < n \leq L-1 \end{cases}. \quad (7)$$

The back-prop term δ is given by

$$\delta_j^{L-n}(k - \Upsilon^n) = \begin{cases} -2e_j(k) f'(s_j^L(k)) & n = 0 \\ f'(s_j^{L-n}(k - \Upsilon^n)) \sum_{m=1}^{N^{L-n+1}} \sum_{c=0}^{T^{L-n}} w_{mj,c}^{L-n} \delta_m^{L-n+1}(k - \Upsilon^n + c) & 1 \leq n \leq L-1 \end{cases} \quad (8)$$

Notice that the last term of equation 8 is also a two-dimensional weighted sum, but this time a temporal correlation between $\underline{w}$ and $\underline{\delta}$ is performed. Once again $\sum_l N^l N^{l+1} T^l$ operations per time step are required. $f'(\cdot)$ represents the derivative of the discriminant function with respect to its input (*i.e.* $f'(s_j^{L-n}) = \partial f(s_j^{L-n}) / \partial s_j^{L-n}$).

Fig. 3 depicts the signal flow in the last layer of a FIRNN during forward and backward propagation. In forward propagation the input signals from the previous layers are multiplied by the weights, summed and delayed, evaluating equation 2. Simultaneously, a derivative of the non-linear function with respect to the partial sums is generated. The output error is calculated at the output of the network. This term is multiplied by the learning rate and the non-linear derivative, producing the delta terms used to update the weights in the output layer. Part (b) depicts backward propagation, where the delta terms are multiplied by exactly the same weights used in forward propagation, but delayed in reverse order to implement the temporal correlation. The output of each FIR is then multiplied by a delayed version of the non-linear derivative, producing the delta terms used to update the previous layer. Notice that back-propagation is a purely linear operation, if one considers the calculation of the non-linear derivative as part of forward propagation. Also notice that the delta terms are automatically delayed by the correct amount of delay during back-propagation, making it unnecessary to store those terms.

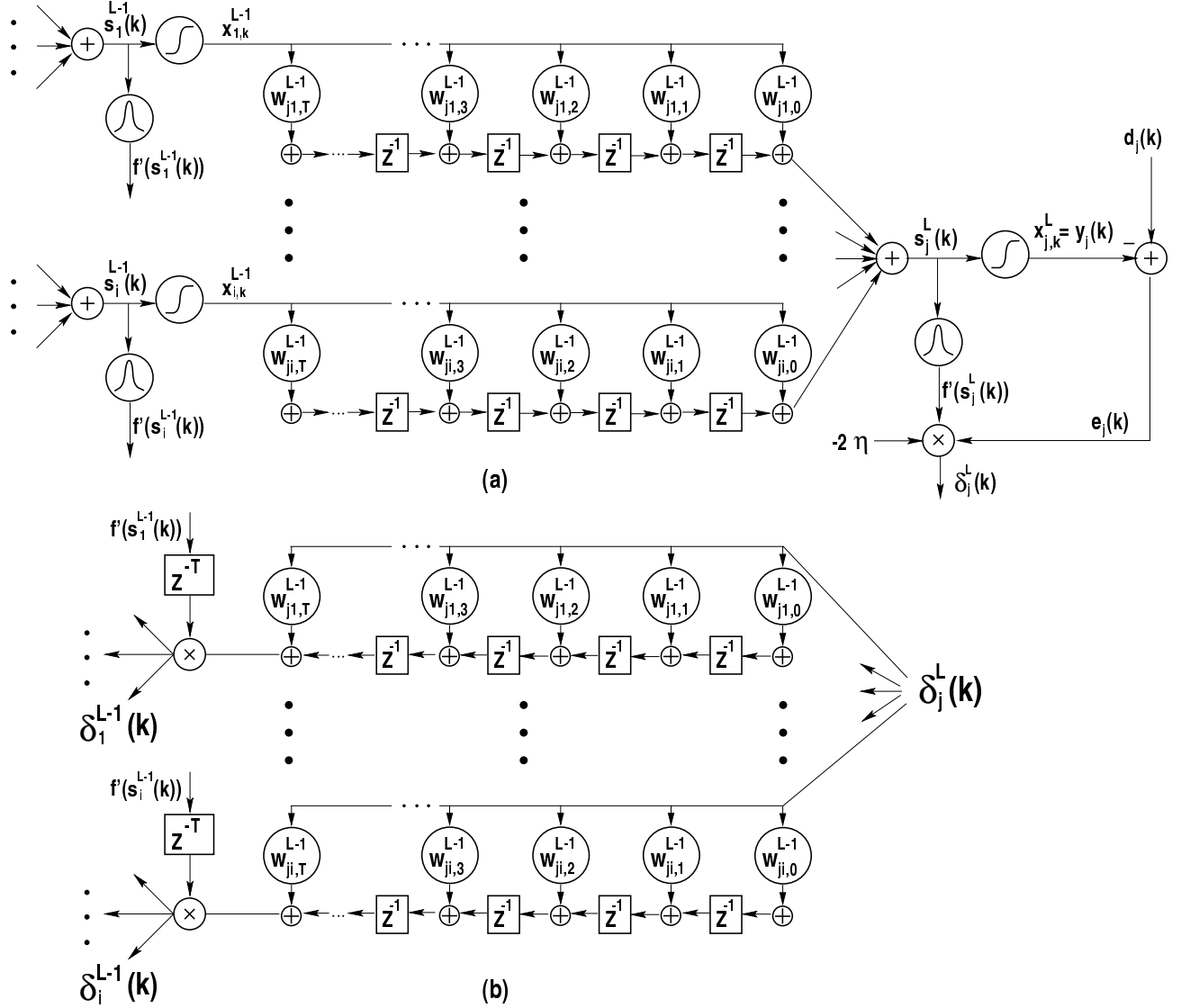


Figure 3. Signal flow in the last layer of a FIRNN during (a) forward propagation and (b) backward propagation. The weights shown in (a) and (b) are exactly the same, but signals flow in opposite direction in each case. Squares represent time delays.

3. OPTICAL ARCHITECTURE

Let us start by describing the implementation of single layer FIRNNs, complete with on-line gradient descent learning. The first one, shown in figure 4 makes use of the SI architecture to forward propagate a spatio-temporal signal $\underline{x}$ producing a vectorial output $\underline{y}$. It operates as follows: the input signal $\underline{x}$ feeds the optical input, which consists of an array of modulators. The input array then interferes with a reference beam in a rotating dynamic volume hologram producing dynamic gratings which physically rotate about the symmetry axis of the volume hologram. The diffraction of the reference by the rotating gratings produces a plane of delayed versions of the input array. This plane is then diffracted by a hologram representing the interconnection weights, recorded in a photorefractive crystal (PRC). The diffracted output is detected in a detector array that is in an image plane of the error modulator array, producing signal $\underline{s}(k)$ via spatial integration. The non-linear operations are performed electronically, as depicted in the dotted balloon, producing the final output $\underline{y}(k)$ and the delta error term $\underline{\delta}(k)$. This term is fed back and is used to modulate an array of electro-optic (EO) modulators, producing the beam used to write a grating with the input plane $\underline{x}(k)$, performing the outer-product necessary for weight adaptation.

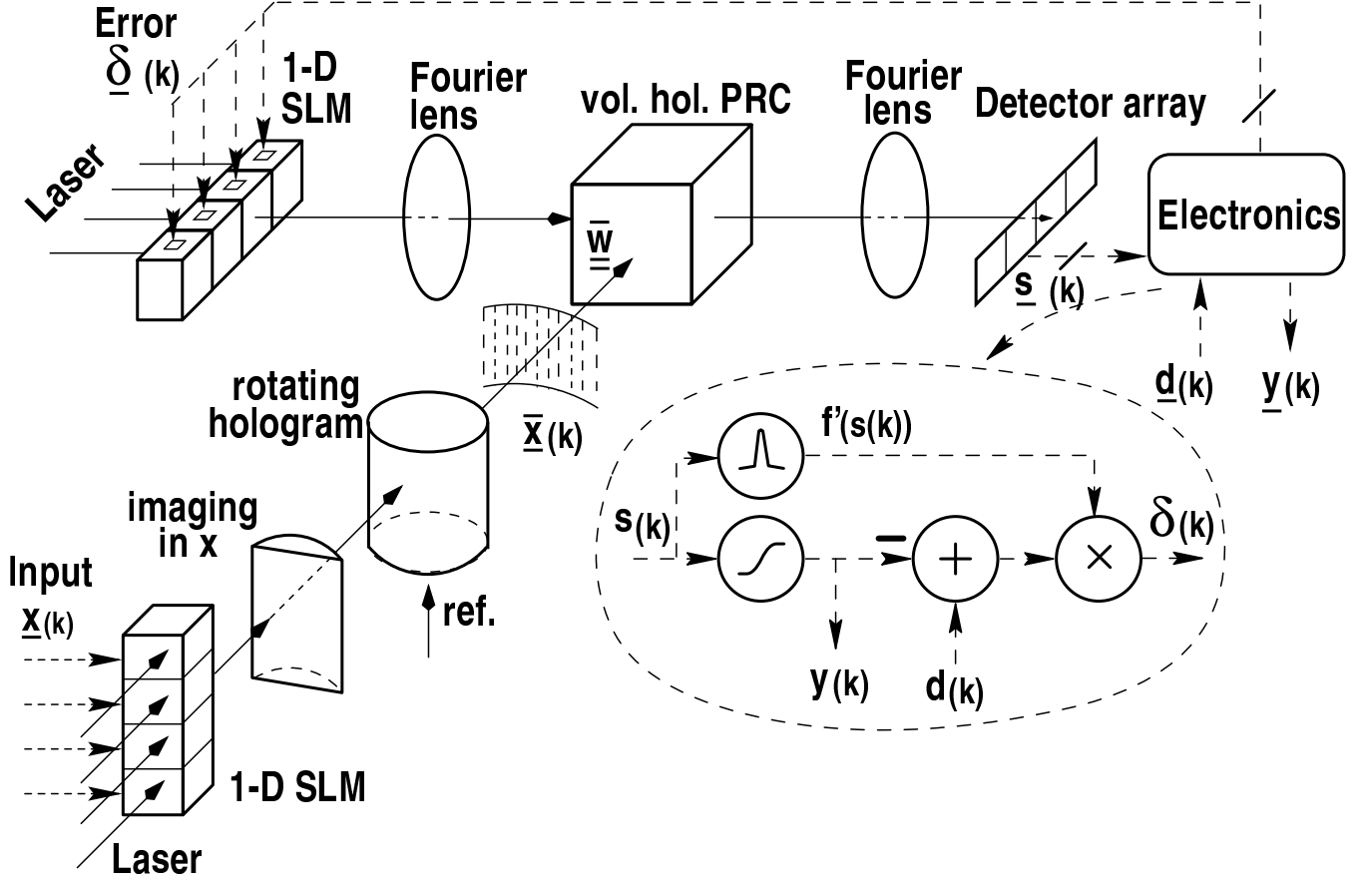


Figure 4. One layer of the FIRNN architecture using a delayed input plane (where the scrolling SLM is implemented with a rotating volume hologram) and space integration for all the inner products. Solid lines represent optic signals and dashed lines represent electronic signals. The diagram in the dotted balloon describes the operations performed electronically.

Notice that $N^{(1)}$ operations are performed electronically, while $2N^{(0)}N^{(1)}T^{(0)}$ operations are performed optically (cubic order computational complexity). Moreover, in the usual case when the network has less output nodes than input nodes, the number of electronic components and the number of EO modulators is reduced.

Fig. 5 describes in more detail the rotating volume hologram used in the SI architecture. Its function is to provide us with an efficient scrolling SLM, capable of delaying a spatial vector $\underline{x}$, producing a spatio-temporal plane $\underline{\bar{x}}$. The input beam writes a grating with the reference beam. As the volume hologram rotates, the recorded gratings are always Bragg matched with the reference beam, simultaneously displaying delayed versions of the input beam. When the signal and reference beam are perpendicular as in Fig. 5(a) the output plane describes a cylinder with its symmetry center in the rotation axis of the volume hologram. This geometry produces a rectilinear output compatible with the shape of a CCD or a smart-pixel array, in contrast to previous implementations of rotating holograms as scrolling SLMs, which produced delays along segments of diffracted conic arcs.¹⁷ Fig. 5(b) uses a liquid gate to easily show how the output cylinder can be mapped to a flat rectangular surface. As the input signal is delayed, its delayed versions are angularly displaced by the rotating hologram, such that their plane of equal optical path difference (OPD) lays on a cylinder. A cylindrical diverging lens compensates for the OPD variation with angle, moving the focal plane to a rectangular surface. Notice that the use of a gate is optional, since it can be replaced with plano-concave diverging lenses which compensate for the focal power of the curvature of the holographic material.

Fig. 6 explains the operation of a single layer FIRNN using time integration to perform the temporal convolutions. Once again, the input is represented by an array of modulators. The input vector is diffracted by a volume hologram representing the interconnect weights and the diffracted beam is detected by a shifting detector array, such as a

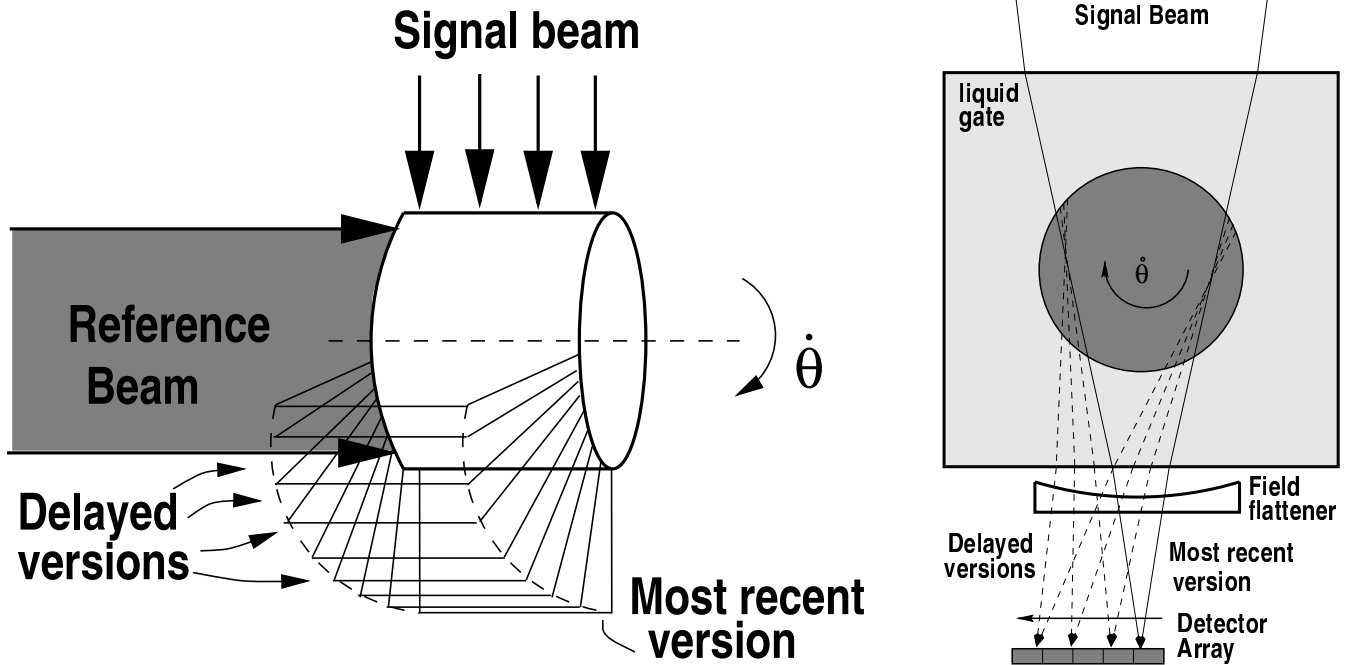


Figure 5. A rotating volume hologram implementing a scrolling SLM. (a) When the reference and signal beams are perpendicular to each other, the output plane describes a cylinder. (b) Top view. A liquid gate helps explain how the output plane can be imaged into a flat plane using a diverging lens to compensate for the variation of OPD with angle.

TDI-CCD or a traveling fringes detector. The temporal integration takes place as the previously detected charges are shifted in space and accumulated with the charges generated from the currently detected diffracted beams. The output signal out of the TDI-CCD undergoes exactly the same electronic processing as in the SI architecture, producing output signal $y(k)$ and the error term $\underline{\delta}(k)$. However, now it is necessary to produce delayed versions of $\underline{\delta}(k)$ in order to perform weight adaptation. A scrolling SLM similar to the one described in Fig. 5 could be used for this purpose. A simpler alternative would be to use a multi-channel acousto-optic deflector (AOD). The number of output nodes would then be limited to 32 due to acoustic diffraction in current AO technologies, but the plane of delayed signals would be provided by one single device. In the AOD the propagation delay of the acoustic waves provides us with the necessary delay to form the error plane $\underline{\delta}(k)$. This plane interferes with a delayed version of the input beam, performing the outer-product responsible for weight adaptation. A simplified version of this architecture, in which a single output node is present and a single AOD is required, is known as the BEAMTAP architecture,¹⁹ used in adaptive phased array processing.

Combination of the input plane and output plane architectures allows for the simultaneous implementation of forward propagation, backward error propagation, and weight updating in a multi-layer FIRNN. Figure 7 shows one possible optoelectronic architecture. For the two-layer case depicted here we have that $\Upsilon^n = 0$ for the last layer ($n = 0$), and $\Upsilon^n = T$ for the first layer ($n = 1$).

A SI architecture similar to the one explained in Fig. 4 is used in forward propagation. Starting from the input, at SLM 1, the one-dimensional array of modulated input signals $\underline{x}^{(0)}(k)$ is recorded in the first rotating hologram, so that the diffraction of the reference beam produces the two-dimensional array $\underline{\bar{x}}^{(0)}(k)$, which is diffracted by the adaptive weight gratings recorded in the volume hologram PRC 1. The array of diffracted signals, $\underline{s}^{(1)}(k)$, is detected by the smart-pixel array (SPA) 1, which performs the non-linear functions $f(\cdot)$ and $f'(\cdot)$, producing the electronic signals $\underline{x}^{(1)}(k)$ and $f'(\underline{s}^{(1)}(k - T))$, respectively. $\underline{x}^{(1)}(k)$ modulates a laser reference beam and is recorded in the second rotating hologram, whose diffraction from its reference beam produces signal $\underline{\bar{x}}^{(1)}(k)$, which is then diffracted by the weight gratings in the volume hologram PRC 2, producing the array of signals $\underline{s}^{(2)}(k)$, which is detected by SPA 2. This SPA generates the final network output $\underline{x}^{(2)}(k)$ and uses the electronic input $\underline{d}(k)$ (the desired signal) to generate the weight update term $\underline{\delta}^{(2)}(k)$, which is fed to SLM 3, which is imaged through PRC 2 onto SPA 2. Notice

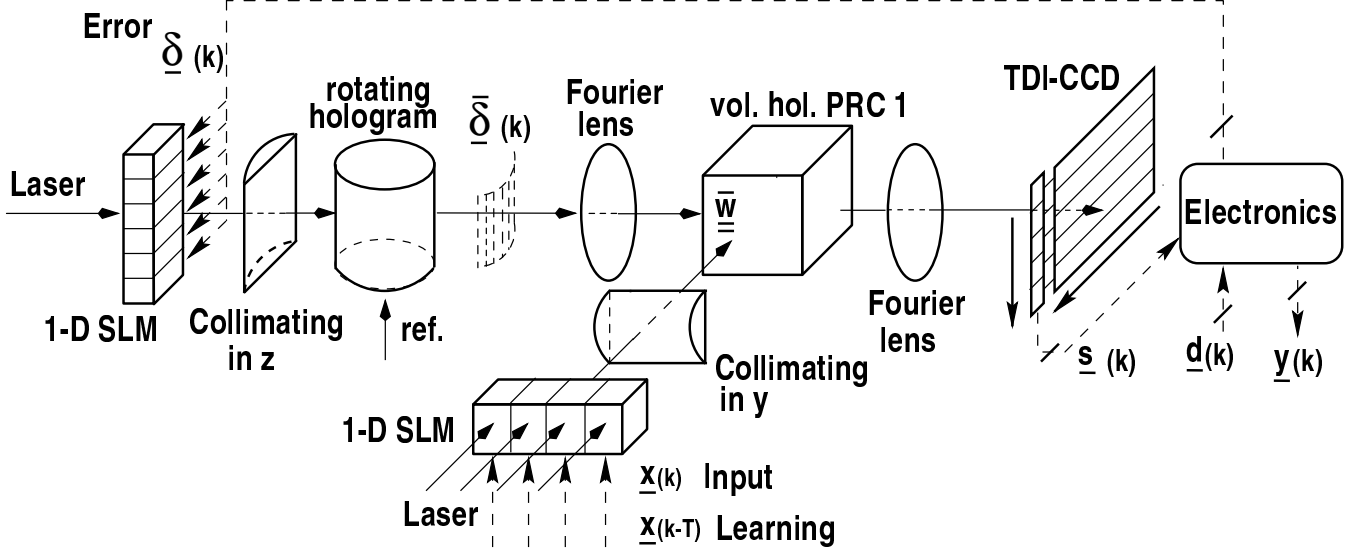


Figure 6. FIRNN optical architecture using a time-delay-and-integrate output plane to calculate the temporal convolutions while using space integrating diffraction from the hologram to calculate the spatial inner products. Solid lines represent optic signals and dashed lines represent electronic signals. The input vector is diffracted by the volume hologram and is detected by the TDI-CCD, which performs the temporal convolution as it shifts and accumulates the detected signals. The error signal is propagated through the velocity matched multi-channel AOD, producing the plane $\bar{\delta}$, which interferes in the PRC with the input signal, modifying the interconnection grating.

that the second rotating hologram displays $\bar{x}^{(1)}(k)$. The interference of this array with the error gradient signal in SLM 3 produces the outer-product used for updating the weights in the volume hologram PRC 2.

Simultaneously, the signal $\delta^{(2)}(k)$ in SLM 3 is back-propagated through the volume hologram PRC 2, being diffracted to the TDI-CCD, which time integrates the back-propagated error in a conjugate plane of the rotating hologram. The TDI-CCD outputs are electronically multiplied by the delayed derivative of the discriminant function, $f'(\underline{s}^{(1)}(k-T))$, stored in SPA 1, producing the error gradient $\delta^{(1)}(k-T)$ applied to SLM 2 which is imaged and pixel aligned through PRC 1 onto SPA 1. The first layer requires the use of a delayed version of its input when updating the weights, since equation 6 in this 2-layer case becomes $\Delta w_{ji,\tau}^{(1)} = -\eta \delta_j^{(1)}(k-T) x_{i,k-T-\tau}^{1-n}$. Setting the angular velocity $\dot{\theta}$ of the rotating hologram 1 to less than π/T , we have that this scrolling SLM will be displaying simultaneously $\bar{x}^{(0)}(k)$ and $\bar{x}^{(0)}(k-T)$, depending on which part of the output plane is used. Using signal $\bar{x}^{(0)}(k-T)$ to perform an outer-product with signal $\delta^{(1)}(k-T)$ from SLM 2, the weights in the volume hologram 1 are correctly updated, finishing the learning cycle.

The spatio-temporal signal plane $\bar{x}^{(0)}(k)$ and the delayed plane $\bar{x}^{(0)}(k-T)$ must be perfectly aligned, since the signal plane $\bar{x}^{(0)}(k)$ reads the grating written by the signal plane $\bar{x}^{(0)}(k-T)$ in combination with the error gradient $\delta^{(1)}(k-T)$. This alignment is performed by the combiner shown in more detail in Fig. 8. A waveplate is used to flip the polarization in the delayed region $\bar{x}(k-T)$ to extraordinary. Walkoff in the off axis cut birefringent crystal serves to align the reading and writing halves of the delayed input plane. An additional compensating crystal may be required to equalize the wavefront curvature. A system consisting of mirrors and beamsplitters could also be used.

4. CONCLUSION

FIRNNs have been presented and analyzed. Two single-layer optical architectures have been presented for their implementation, one using a scrolling SLM implementation and the second using a scrolling time-delay-and-integrate detector array such as a TDI-CCD. A multi-layer FIRNN architecture simultaneously capable of forward propagation, backward error propagation, and weight adaptation has been explained. The all-optical architectures presented here make use of the parallelism of volume holographic optical processing to perform all of the cubic order linear computations (the N 2-dimensional weighted sums and outer-products), and electronics for performing the low

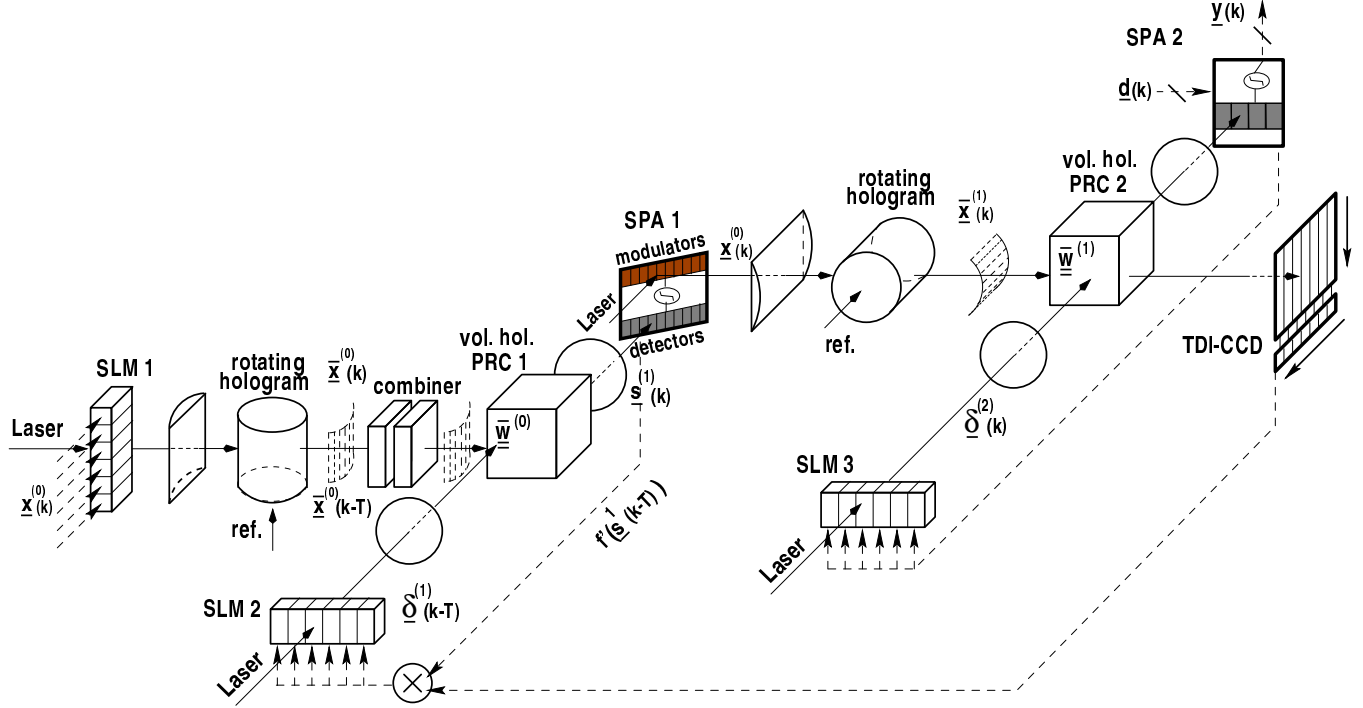


Figure 7. Optical architecture of a complete two-layer FIRNN, including forward propagation, backward error propagation and weight updating. Optic signals are represented by solid lines and electronic signals are represented by dashed lines. The SI architecture is used for forward propagation and the SI architecture is used for backward propagation of the output δ terms.

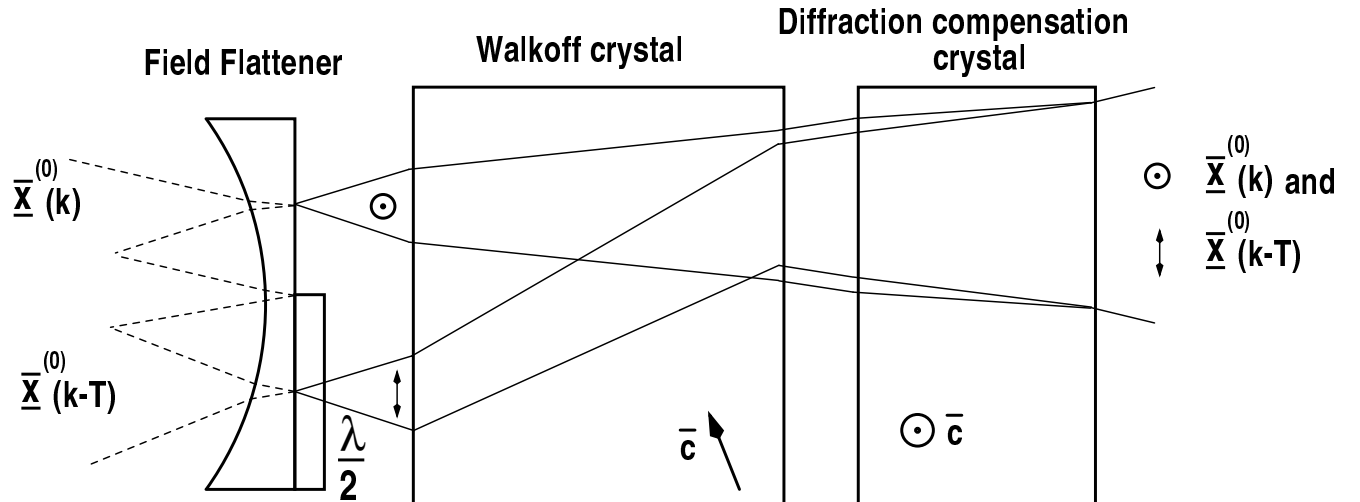


Figure 8. Optical combiner used to align plane $\underline{\bar{x}}^{(0)}(k)$ with delayed plane $\underline{\bar{x}}^{(0)}(k-T)$. The first crystal is cut at an angle α with respect to the crystal axis $\bar{c}$, causing the extraordinarily polarized plane to walk-off towards the ordinarily polarized plane. The second crystal is used to produce equal radii of curvature.

complexity non-linear operations (output non-linearities and its derivatives), making efficient use of the capabilities and limitations of each technology.

ACKNOWLEDGMENTS

We wish to acknowledge the support by Dr. William Miceli of ONR and the Office of the Secretary of Defense DDR&E through the MURI program grant no N00014-97-1-1006.

REFERENCES

1. P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Kluwer Academic Publishers, 1997.
2. L. J. G. B. Widrow, P. E. Mantey and B. B. Goode, "Adaptive antenna systems," *Proceedings of the IEEE* **55**(12), pp. 2143–2161, 1967.
3. B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice Hall, 1985.
4. A. V. Vanderlugt, "Adaptive optical processor," *Applied Optics* **21**, p. 4005, 1982.
5. J. Rhodes, "Adaptive filter with a time-domain implementation using correlation cancellation loops," *Applied Optics* **22**, p. 282, 1983.
6. J. H. Hong and I. McMichael, "Novel optical technique for phased-array processing," *Optical Engineering* **30**(12), p. 1976, 1991.
7. C. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.
8. D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks," *Complex Systems* **2**, pp. 321–355, 1988.
9. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representation by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, D. E. Rumelhart and J. L. McClelland, eds., ch. 8, MIT Press, Cambridge, MA, 1986.
10. D. Psaltis and Y. Qiao, "Adaptive multilayer optical networks," *Progress In Optics* **31**, pp. 227–261, 1993.
11. M. A. Neifeld and D. Psaltis, "Optical implementations of radial basis classifiers," *Applied Optics* **32**, pp. 1370–1379, 1993.
12. K. Wagner and D. Psaltis, "Multilayer optical learning networks," *Applied Optics* **26**, pp. 5061–5076, 1987.
13. K. Wagner and T. M. Slagle, "Optical competitive learning with VLSI liquid-crystal winner-take-all modulators," *Applied Optics* **32**, pp. 1408–1435, 1993.
14. E. A. Wan, "Temporal backpropagation for FIR neural networks," *Proc. Int. Joint Conf. Neural Networks*, pp. I 575–580, June 1990.
15. E. A. Wan, "Time series prediction by using a connectionist network with internal delay lines," in *Time Series Prediction: Forecasting the Future and Understanding the Past*, A. S. Weigend and N. A. Gershenfeld, eds., Addison-Wesley, 1993.
16. A. H. Waibel, T. Hanazawa, G. E. H. K. Shikano, and K. J. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech and Signal Processing* **37**(3), pp. 328–339, 1989.
17. Z. Gan and D. Z. Anderson, "Acoustic signal recognition with a photorefractive time-delay neural network," *Optics-Lett* **19**(9), pp. 655–657, 1994.
18. P. E. X. Silveira and K. H. Wagner, "Time delay neural networks," *Int. Conf. on Optics in Computing* **34**(90), pp. 266–269, 1998.
19. K. H. Wagner, S. Kraut, L. Griffiths, S. Weaver, R. Weverka, and A. Sarto, "Efficient true-time-delay adaptive array processing," in *Proc. SPIE 2845*, IEEE, 1996.