

December 2003

Hermite neural network correlation and application

M. R. Mackenzie
University of Wollongong

K. Tieu
University of Wollongong, ktieu@uow.edu.au

Follow this and additional works at: <https://ro.uow.edu.au/engpapers>



Part of the [Engineering Commons](#)

<https://ro.uow.edu.au/engpapers/38>

Recommended Citation

Mackenzie, M. R. and Tieu, K.: Hermite neural network correlation and application 2003.
<https://ro.uow.edu.au/engpapers/38>

Hermite Neural Network Correlation and Application

Mark R. Mackenzie and A. Kiet Tieu

Abstract—A method for obtaining the correlation of two Hermite neural network is developed. The method is based on the fact that a Hermite function is unchanged by the Fourier transform, which allows an expression for the correlation to be obtained directly from the network weights without the need for the Fourier transform. Comparative results with other neural network correlation methods are presented on simulated radar signals.

Index Terms—Correlation, Hermite functions, neural network, radar.

I. NOMENCLATURE

α, β, γ	Scalar constants.
μ	Feedback constant of the gradient descent algorithm.
$a(t), b(t)$	Time functions.
A_n, B_n	Neural network weights for $a(t)$ and $b(t)$, respectively.
$R(t)$	Correlation of $a(t)$ and $b(t)$.
$a_h(t)$	Hermite neural network interpolation of $a(t)$.
$h_n(t)$	Orthonormal Hermite function of order $\{n\}$.
$H_n(t)$	Hermite polynomial of order $\{n\}$.
$F\{a(t)\}$	Fourier transform of $a(t)$.
$l_m^n(t)$	Orthonormal associated Laguerre function of order $\{m, n\}$.
$L_m^n(t)$	Associated Laguerre polynomial of order $\{m, n\}$.
$hr_n(t)$	Hermite–Rodriguez function of order $\{n\}$.
HNN	Hermite neural network.
HRNN	Hermite–Rodriguez neural network.
FNN	Fourier neural network.
SNN	Sigmoidal neural network.

II. INTRODUCTION

DUE to its Gaussian weight function, the Hermite series expansion has proven to be useful for signal processing applications. Applications of the Hermite series have been reported in visual information processing [1]–[3] as well as biomedical engineering [4], [5]; however, in biomedical engineering, motivation for using the Hermite series is because the lower order Hermite functions bear some resemblance to the impulse response signals received from electrically stimulated organs. This allows for a compact series interpolation.

Manuscript received September 24, 2002; revised March 27, 2003. The associate editor coordinating the review of this paper and approving it for publication was Prof. Derong Liu.

The authors are with the School of Mechanical, Materials, and Mechatronic Engineering, University of Wollongong, Wollongong, N.S.W. Australia 2522 (e-mail: kiet_tieu@uow.edu.au).

Digital Object Identifier 10.1109/TSP.2003.819005

Recently, the adaptation of algorithms such as back propagation, which was originally developed for the sigmoidal neural network, to other types of functions besides the sigmoid has led to the general classification of orthogonal series expansions as a neural network [6], [7]. Consequently, the Hermite series expansion of a function may be regarded as a Hermite neural network (HNN). The useful feature of neural networks is that they may be implemented as parallel analogue hard-wired circuits [8], [9], which greatly improves computation speed.

An important property of a Hermite function is that its Fourier transform is also a Hermite function apart from a multiplication factor that is dependent on the order of the Hermite function. Therefore, the Fourier transform of a function, approximated by a Hermite series neural network, can be obtained readily. This property is exploited in this paper to develop an algorithm for calculating the cross correlation using HNNs. This algorithm bypasses the usual Fourier transform approach to calculate the correlation. Instead, the correlation is calculated directly from the Hermite series coefficients. The advantage is that for some applications (which are defined in Section III-B), the direct approach is computationally more efficient. These applications are investigated further in Section IV, where a comparison with other neural networks is given.

The layout of the paper is as follows. In Section III, after the Hermite neural network is defined, the correlation of two HNNs is derived. Section IV discusses the method of application of the HNN correlation, particularly training the network. Section V applies the HNN to Doppler radar and also the CHIRP radar.

III. THEORY

A. Hermite Orthonormal Neural Networks

The HNN approximation for a function $a(t)$ is defined to be

$$a_h(t) = \sum_{n=0}^N A_n h_n(t) \quad (1)$$

where the coefficients $\{A_n\}$ are a series of constants, referred to as weights, which minimize the mean square error defined as

$$\text{mse} = \int_{-\infty}^{+\infty} (a(t) - a_h(t))^2 dt. \quad (2)$$

These weights are also sometimes called the Hermite transform of $a(t)$. The Hermite orthonormal functions $h_n(t)$ are related to the Hermite polynomials $H_n(t)$ by

$$h_n(t) = \frac{H_n(t)e^{-t^2/2}}{\sqrt{2^n n! \sqrt{\pi}}} \quad (3)$$

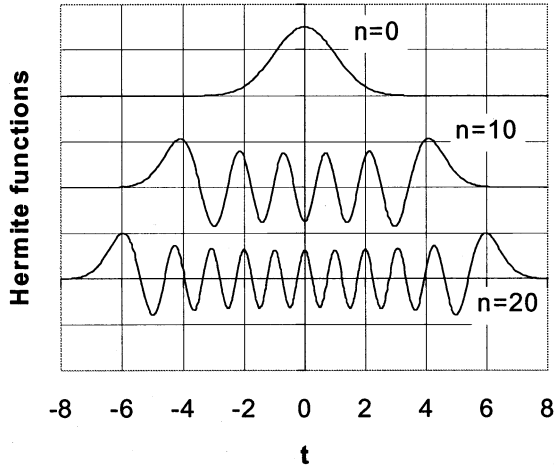


Fig. 1. Orthonormal Hermite functions of order $n = 0, 10$, and 20 .

where one of the many possible explicit forms of the Hermite polynomials is the derivative of the Gaussian function

$$H_n(t) = (-1)^n e^{t^2} \frac{d^n}{dt^n} (e^{-t^2}). \quad (4)$$

Some of these Hermite functions are shown in Fig. 1. The higher order functions resemble a cosine/sine wave of slowly decreasing frequency and increasing amplitude.

B. Correlation Using Hermite Neural Networks

Correlation algorithms using Hermite series has been successfully applied by Nayakkankupam *et al.* [10] and Konstantopoulos [11] to the deconvolution of blurred Gaussian images. Nayakkankupam matched the scale of associated Hermite series in the frequency domain to achieve a linear set of equations in the form of a matrix equation; the limitation is that it requires the solution of a matrix equation. Konstantopoulos algorithms were based on the convolution properties of the Hermite–Rodriguez functions [4]; these functions are Gaussian windowed, Hermite functions. In this section, we develop an alternative correlation algorithm.

The cross correlation¹ of two function $a(t)$ and $b(t)$ is defined in the usual way as

$$R(t) = a(t) * b(t) = \int_{-\infty}^{+\infty} a(\tau) b(\tau + t) d\tau. \quad (5)$$

Using HNNs for $a(t)$ and $b(t)$, the cross correlation of these functions is

$$\begin{aligned} R_h(t) &= \int_{-\infty}^{+\infty} a_h(\tau) b_h(\tau + t) d\tau \\ &= \sum_{n,m}^N A_n B_m \{h_n(t) * h_m(t)\} \end{aligned} \quad (6)$$

¹The closely related convolution integral is

$$a(t) \otimes b(t) = \int_{-\infty}^{+\infty} a(\tau) b(t - \tau) d\tau.$$

The results presented in this paper can be applied to the convolution integral by direct multiplication in the frequency domain rather than using the complex conjugate.

where $h_n(t) * h_m(t)$ is the correlation of two Hermite functions of different order, which is given by

$$h_n(t) * h_m(t) = \int_{-\infty}^{+\infty} h_n(\tau) h_m(\tau + t) d\tau. \quad (7)$$

Therefore, the correlation of two HNNs reduces to that of a summation of cross correlated Hermite functions.

Although it is possible to evaluate the cross correlation of the Hermite functions $h_n(t) * h_m(t)$ in the time domain, a mathematically simpler method is to work in the frequency domain and then take the inverse Fourier transform to obtain the correlation function. In the frequency domain, the integration associated with the correlation becomes a complex conjugate multiplication, which is easier to manipulate. Using this method, the correlation of the Hermite functions from a frequency viewpoint is

$$h_n(t) * h_m(t) = (-1)^m j^{n+m} \int_{-\infty}^{+\infty} h_n(\omega) h_m(\omega) e^{j\omega t} d\omega \quad (8)$$

where we have used the Fourier transform $F\{h_n(t)\}$ of the Hermite function given by

$$F\{h_n(t)\} = (-j)^n \sqrt{2\pi} \cdot h_n(\omega). \quad (9)$$

For the special case of $n = m$, the correlation is a Laguerre function [15]. The general case of $n \neq m$ is not usually given in the general mathematical literature in the form of (7), although (8) can be found listed in integration tables [12] as a cosine and sine integration. It is beneficial to show how it can be evaluated directly because a similar procedure may possibly be applied to other types of orthonormal functions, which are not listed in tables. It can be evaluated directly as follows. The product formula [12] for Hermite polynomials is

$$\begin{aligned} H_n(\omega) H_m(\omega) &= \sum_{k=0}^m k! 2^k \binom{m}{k} \binom{n}{k} H_{m+n-2k}(\omega) \quad (m \leq n). \end{aligned} \quad (10)$$

Substituting (10) and (3) into (8) gives the integration equation

$$\begin{aligned} h_n(t) * h_m(t) &= \frac{(-1)^m j^{n+m}}{\sqrt{2^{n+m} \pi n! m!}} \sum_{k=0}^m k! 2^k \binom{m}{k} \binom{n}{k} \\ &\times \int_{-\infty}^{+\infty} e^{-\omega^2} H_{m+n-2k}(\omega) e^{j\omega t} d\omega \quad (m \leq n). \end{aligned} \quad (11)$$

Equation (11) can be rewritten in a more suitable form for integration with the Gaussian derivative definition of the Hermite polynomials [see (4)]. Substituting (4) into (11) gives

$$\begin{aligned} h_n(t) * h_m(t) &= \frac{(-1)^m j^{n+m}}{\sqrt{2^{n+m} \pi n! m!}} \sum_{k=0}^m k! 2^k \binom{m}{k} \binom{n}{k} (-1)^{n+m-2k} \\ &\times \int_{-\infty}^{+\infty} \frac{d^{n+m-2k}}{d\omega^{n+m-2k}} (e^{-\omega^2}) e^{j\omega t} d\omega \quad (m \leq n). \end{aligned} \quad (12)$$

The integration in (12) is the inverse Fourier transform (IFT) of the derivative of a Gaussian function. This can be derived

from the Fourier transform of the Gaussian function [13] by the differentiation.

$$\int_{-\infty}^{+\infty} \frac{d^n}{d\omega^n} (e^{-\omega^2}) e^{j\omega t} d\omega = (-j)^n t^n e^{-t^2/4} \sqrt{\pi}. \quad (13)$$

Using this result, we have

$$\begin{aligned} h_n(t) * h_m(t) &= \frac{(-1)^n}{\sqrt{2^{n+m} n! m!}} \\ &\times \sum_{k=0}^m k! 2^k \binom{m}{k} \binom{n}{k} (-1)^k t^{n+m-2k} e^{-t^2/4} \\ &(m \leq n). \end{aligned} \quad (14)$$

By changing the order of the summation, this may be written as an orthonormal associated Laguerre function

$$\begin{aligned} h_n(t) * h_m(t) &= \begin{cases} (-1)^{n+m} l_m^{n-m}(\frac{t^2}{2}), & \text{for } t \geq 0 \\ l_m^{n-m}(\frac{t^2}{2}), & \text{for } t < 0 \end{cases} \quad (m \leq n) \end{aligned} \quad (15)$$

where $l_m^n(t)$ is the orthonormal associated Laguerre function related to the associated Laguerre polynomials $L_m^n(t)$ by

$$l_m^n(t) = \frac{t^{n/2} e^{-t/2} L_m^n(t)}{\sqrt{\frac{(n+m)!}{m!}}}. \quad (16)$$

With this result, we can now evaluate the correlation of the two HNNs. Substituting (15) into the network correlation (6), we obtain the final result

$$R_h(t) = \sum_{m=0}^N \sum_{n=m}^N (A_n B_m (-1)^{n+m} \delta_{n,m} + A_m B_n) \cdot l_m^{n-m} \left(\frac{t^2}{2} \right) \quad \text{for } t \geq 0 \quad (17)$$

where

$$\delta_{n,m} = \begin{cases} 1, & \text{for } n \neq m \\ 0, & \text{for } n = m \end{cases} \quad (18)$$

with a similar summation for $t < 0$.

This equation is a summation of N^2 terms. Stone and Williams [14] have shown that only the Fourier family of linear transforms can replace the N^2 summation terms of a convolution integral in the time domain by N summations in the transform domain. Generally, the computation of the correlation will be faster with a Fourier transform network, except for the special case that occurs when one of the functions is a Hermite function $a(t) = h_k(t)$. In this case, the correlation reduces to a summation of N terms given by

$$\begin{aligned} R_h(t) &= \sum_{m=0}^k B_m (-1)^{m+k} l_m^{k-m} \left(\frac{t^2}{2} \right) \\ &+ \sum_{m=k+1}^N B_m l_m^{m-k} \left(\frac{t^2}{2} \right), \quad \text{for } t \geq 0 \end{aligned} \quad (19)$$

with a similar summation for $t < 0$. Functions that can be represented by a single Hermite function are the Gaussian function $h_0(t)$ and the rotationally invariant CHIRP radar signal [15]. The Gaussian function correlation is important in the filtering of

noisy signals [16] as well as in the previously mentioned image and signal deconvolution [10], [11].

IV. METHOD

In order to apply the HNN correlation to practical problems, the various properties of the Hermite and Laguerre functions must be considered, and the generation of these functions and the training of the network weights must be specified.

A. Signal Duration and Bandwidth of Hermite Functions

The HNN behaves as a window in the time domain. Outside this window, the Hermite functions decay exponentially, limiting the effective range over which a function may be approximated to within the window. The width of the HNN window is equal to the duration of the largest order Hermite function $h_N(t)$ occurring in the network. The useful range of application of the HNN is then

$$|t| \leq \sqrt{2N+1} \quad (20)$$

where the right-hand side, which is equal to the duration of the Hermite function of order $\{N\}$, may be determined via the Quantum mechanic solution of the Harmonic oscillator as the location where the oscillator energy becomes negative [17].

In view of the isomorphic Fourier transform of the Hermite function, a similar windowing effect occurs in the complex frequency domain. The useful bandwidth is

$$|\omega| \leq \sqrt{2N+1}. \quad (21)$$

Together, the bandwidth and window width of (20) and (21) define the size N of the neural network required to approximate a function.

B. Recurrence Relations for Hermite and Laguerre Functions

Hermite and associated Laguerre functions are computed numerically by the following recurrence relations, respectively [18]

$$\begin{aligned} h_{k+1}(t) &= t \cdot h_k(t) \sqrt{\frac{2}{k+1}} \\ &- h_{k-1}(t) \sqrt{\frac{k}{k+1}} \end{aligned} \quad (22)$$

$$\begin{aligned} l_{k+1}^n(t) \sqrt{\frac{k+1}{n+k+1}} &= \frac{2k+n+1-t}{k+1} \cdot l_k^n(t) \\ &- \frac{k+n}{k+1} \sqrt{\frac{n+k}{k}} \cdot l_{k-1}^n(t) \end{aligned} \quad (23)$$

with the initial values

$$\begin{aligned} h_0(t) &= \frac{1}{\sqrt{\pi}} e^{-t^2/2} \\ h_1(t) &= \frac{2t}{\sqrt{2\pi}} e^{-t^2/2} \end{aligned} \quad (24)$$

$$\begin{aligned} l_0^n(t) &= \frac{t^{n/2} e^{-t/2}}{\sqrt{n!}} \\ l_1^n(t) &= \frac{(n+1-t) \cdot t^{n/2} e^{-t/2}}{\sqrt{(n+1)!}} \end{aligned} \quad (25)$$

Due to the double indices of the associated Laguerre function, more than one recurrence relation exists. Equation (23) computes the associated Laguerre function from a recurrence rela-

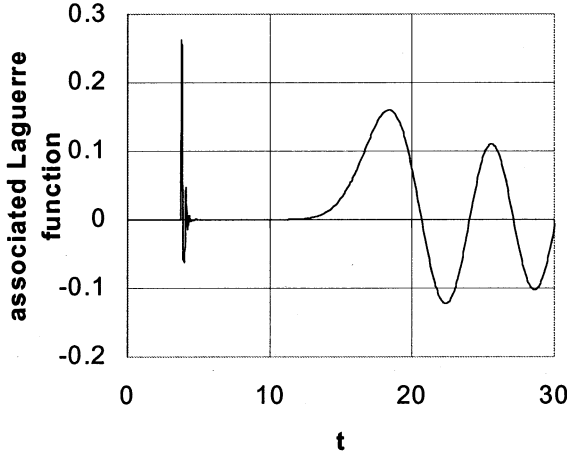


Fig. 2. Associated Laguerre function $l_{50}^{75}(t)$ showing instability of the recurrence relation at small t .

tion using the subscript. Unfortunately, for large n , it becomes difficult to use because of the power term in the initial conditions of (25). In this case, the following recurrence equation, which sidesteps the power term by using the superscript, was employed [19]

$$l_m^{k+1}(t) = \frac{k+t}{\sqrt{t \cdot (m+k+1)}} l_m^k(t) - l_m^{k-1}(t) \sqrt{\frac{m+k}{m+k+1}} \quad (26)$$

where the initial values for $k = 0$ and $k = 1$ were computed from (23). For small values of t , this equation is unstable (see Fig. 2). Further investigation revealed there is a cut-off point: Above this point, the recurrence relation is stable; below it is unstable, although in this region, the magnitude of the Laguerre function is negligible. The cut-off point is a function of the order (n, m) of the associated Laguerre function. Although its location can be estimated by analytical methods [20], this approach was found to be overly cautious. Instead, a two-dimensional (2-D) lookup table, which maps the cut-off point as a surface function of (n, m) , was found to be more suitable. To avoid excessive storage, not all (n, m) were tabulated; linear interpolation was used to locate the cut-off point for those values of (n, m) not given in the table.

One of the important characteristics of neural networks is that they can be implemented using analog electronics [8], [9]. A method of electronically generating Hermite functions was recently proposed by Chesnokov [21], which suggests the possibility of an analogue HNN, rather than the computer generated functions described above.

C. Training of the Neural Network

For a continuous function defined on $\{-\infty, \infty\}$, the weights of the HNN are optimum [22] with respect to the mean square error [see (2)] when

$$A_n = \int_{-\infty}^{+\infty} a(t) h_n(t) dt. \quad (27)$$

This integration may be approximated numerically with Gauss–Jacobian integration [23]. Sometimes, the scale and/or bias of

the network are also optimized with back propagation or a similar algorithm [24].

The difficulty with Gauss–Jacobian numerical integration is that the zeroes of the Hermite functions are required. For an order $\{n\}$ Hermite function, there are n zeroes. Consequently, for a neural network consisting of a total of N Hermite functions, a large number of zeroes must be found. In biomedical engineering applications, where only a few Hermite functions are required, this is acceptable, but for large networks, it reduces computational speed or storage space. For this reason, we use a simple summation, similar to Euler integration, which is given by

$$A_n = \sum_{i=0}^{i=I} a(i\Delta t) h_n(i\Delta t) \quad (28)$$

where Δt is the integration step size, and I is the total number of samples. A feature of this type of integration is that it is also suitable for randomly sampled data. For random data, this type of numerical integration generalizes to Monte Carlo integration. Although instability may occur in Euler integration, this has not been encountered in any of the applications so far investigated with the HNN.

Numerical integration is only an approximation to the analytical continuous integration. In addition, the data most often encountered in practice is discrete and often corrupted with noise. To cope with these situations, the least mean square (LMS) algorithm was applied after the weights had been estimated with the integration. LMS reduces the mse between the neural network approximation and the discrete data by successive iterations of the following algorithm:

$$\begin{aligned} A_k((i+1)\Delta t) &= A_k(i\Delta t) \\ &+ \mu \left(a(i\Delta t) - \sum_{n=0}^N A_n(i\Delta t) h_n(i\Delta t) \right) \\ &\times h_k(i\Delta t) \end{aligned} \quad (29)$$

where μ is the learning rate constant chosen in the range 0.0 to 1.0. In more elaborate training schemes, μ may be adaptively adjusted or adapted using directional minimization.

V. APPLICATION OF HNN CORRELATION IN SIGNAL PROCESSING

The objective of this section is to compare the HNN correlation with other methods on practical signal processing problems. In Section IV-A, we compare its performance with an FNN, an HRNN, and a conventional SNN on Doppler radar signals. In Section IV-B, we compare the HNN performance with an FNN on CHIRP radar signals. Fourier and Hermite–Rodriguez neural networks are defined in the same way as the HNN of (1), except that a Fourier or Hermite–Rodriguez function replaces the Hermite function in the network.

A. Demodulation of Doppler Radar Signals

Fig. 3(a) shows the signal received from the detector of a Doppler radar system. To obtain the target range, a filter matched to the signal pulse removes the carrier frequency,

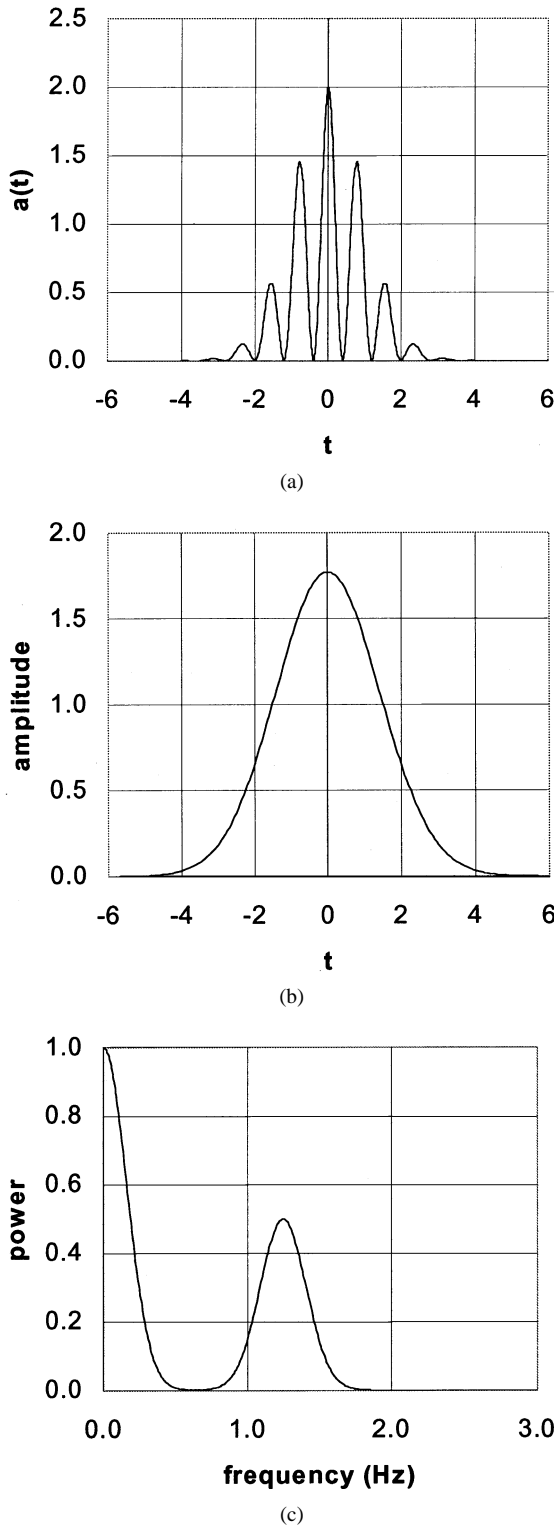


Fig. 3. (a) HNN approximation of Doppler radar signal. (b) HNN demodulation of Doppler radar signal. (c) HNN Fourier transform of Doppler radar signal.

which is proportional to the velocity of the target. The HNN is particularly well suited to processing these types of signals because the target range and velocity are obtained directly from the weights of the HNN, which are fitted to the Doppler signal. The new neural network formed by replacing the Hermite functions in the HNN approximation of the Doppler signal by their Fourier transform gives the frequency spectrum of

the Doppler signal from which the velocity may be determined. From (9), the frequency spectrum is

$$F\{a_n(t)\} = \sqrt{2\pi} \sum_{n=0}^N (-j)^n A_n h_n(t). \quad (30)$$

Removal of the carrier frequency to obtain the range of the target is achieved by taking the correlation of the HNN with the Gaussian function, which functions as a matched filter. Since the fundamental Hermite function is Gaussian, this may be taken to be the filter, and the correlation is then the Laguerre network of (5), with $h_k(t) = h_0(t)$.

The following mathematical model of a Doppler signal was investigated:

$$a(t) = (1 + \cos(2\pi ft)) e^{-t^2/2\sigma^2} \quad (31)$$

where the frequency $f = 1.25$ Hz and the Gaussian width $\sigma = 1.0$. Simulated random noise, with a uniform probability density, of varying strength was added. SNRs were evaluated across the interval $\{-10 \leq t \leq 10\}$. An HNN of 50 elements has a width of 10s; however, an HNN with 70 elements was used to ensure an accurate approximation of the Doppler signal. Fig. 4(a) shows the root mean square error (rmse) of the HNN versus the number of Hermite functions in the network. Fig. 3(a)–(c) shows the signal, matched filter output, and frequency spectrum, respectively, approximated by the HNN of 70 elements.

Comparison of the signal-to-noise ratio (SNR) with an FNN (period = 20 s) on the same signal is shown in Fig. 6(a), where the number of Fourier functions to achieve a satisfactory rmse was 80 [see Fig. 4(b)]. Since both networks achieve the same SNR, the choice between each network is then dependent on the application. The FNN is periodic and suitable for signals of compact support, whereas the HNN is preferable for signals defined on the entire real line. The other difference is that the FNN requires zero-padding to avoid a circular error in the correlation, whereas the HNN does not. In Section IV-B, we demonstrate an application that favors the HNN.

Hermite–Rodriguez functions [4], [11] may also be applied successfully to process the Doppler signal. These functions are Gaussian windowed Hermite functions, which are defined as

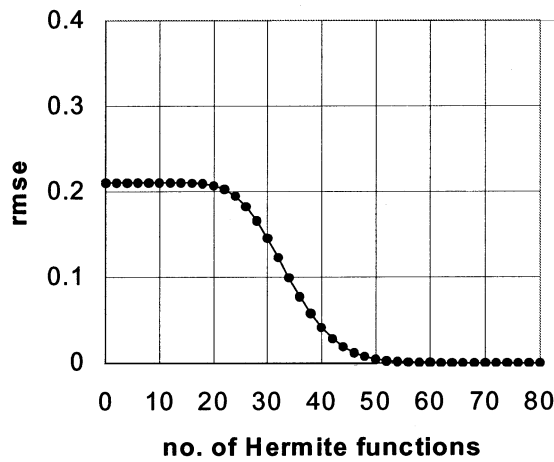
$$hr_n(t) = \frac{h_n(t) e^{-t^2/2}}{\pi^{1/4}} \quad (32)$$

where $h_n(t)$ is an orthonormal Hermite function. Like the Hermite series, simple expressions also occur for the correlation and Fourier transform of the Hermite–Rodriguez functions. The relevant mathematical equations given below are taken from Lo Conte *et al.* [4] The correlation of two Hermite–Rodriguez functions is given by

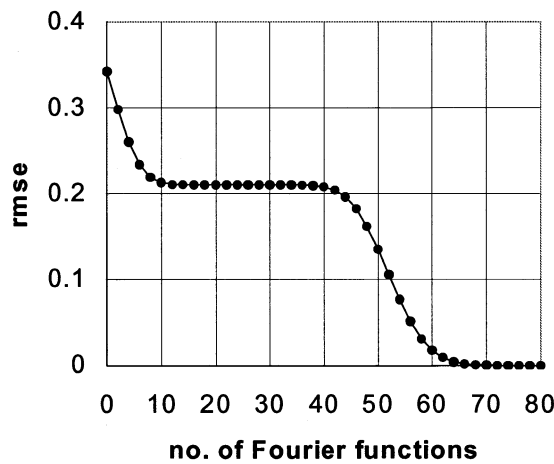
$$hr_n(t) * hr_m(t) = \sqrt{\frac{(n+m)!}{2^{n+m} n! m!}} hr_{n+m} \left(\frac{t}{\sqrt{2}} \right) \quad (33)$$

while the Fourier transform of the Hermite–Rodriguez function is an associated Laguerre function

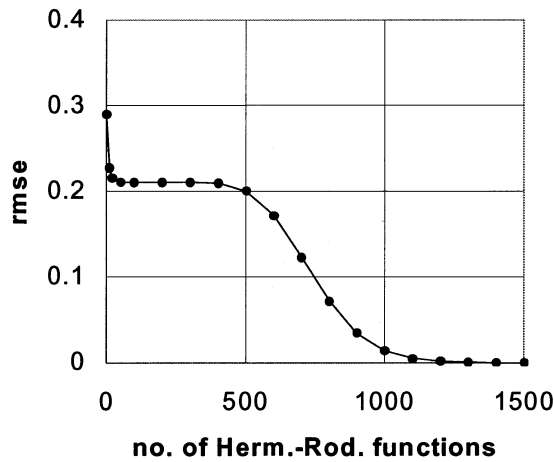
$$F\{hr_n(t)\} = (-j)^n l_n^0 \left(\frac{\omega^2}{2} \right). \quad (34)$$



(a)



(b)

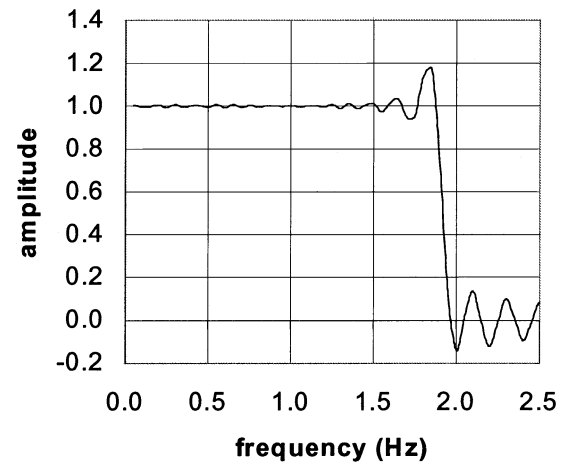


(c)

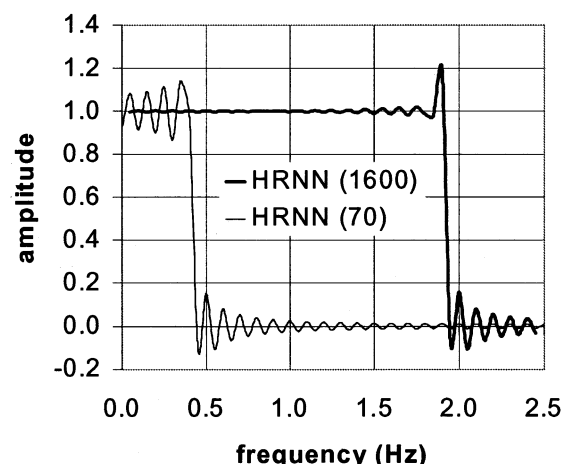
Fig. 4. (a) Root mean square error (rmse) versus number of Hermite functions. (b) Root mean square error versus number of Fourier functions. (c) Root mean square error versus number of Hermite–Rodriguez functions.

Note that the scale is reduced and that the order of the Hermite–Rodriguez function increased by the correlation operation.

Application requires the duration and bandwidth of the HRNN to be matched to the Doppler signal. Unlike the HNN, which increases in duration with the order $\{N\}$ of the function,



(a)



(b)

Fig. 5. (a) Frequency response of HNN of size $n = 70$. (b) Frequency response of HRNN of size $n = 70$ and $n = 1600$.

the HRNN is independent of N . Instead, it is limited in duration by the Gaussian window to approximately the range

$$|t| \leq 2. \quad (35)$$

However, the bandwidth in the frequency domain increases with the order $\{N\}$ of the function according to

$$|\omega| \leq \sqrt{2N+1} \quad (36)$$

which is the same as for the Hermite function [see (21)]. Scaling of the Hermite–Rodriguez series is required by introducing the variable

$$t \rightarrow \frac{t}{\alpha}. \quad (37)$$

Scaling the HRNN changes the duration and bandwidth of the network to, respectively

$$|t| \leq 2\alpha \quad (38)$$

and

$$|\omega| \leq \frac{\sqrt{2N+1}}{\alpha}. \quad (39)$$

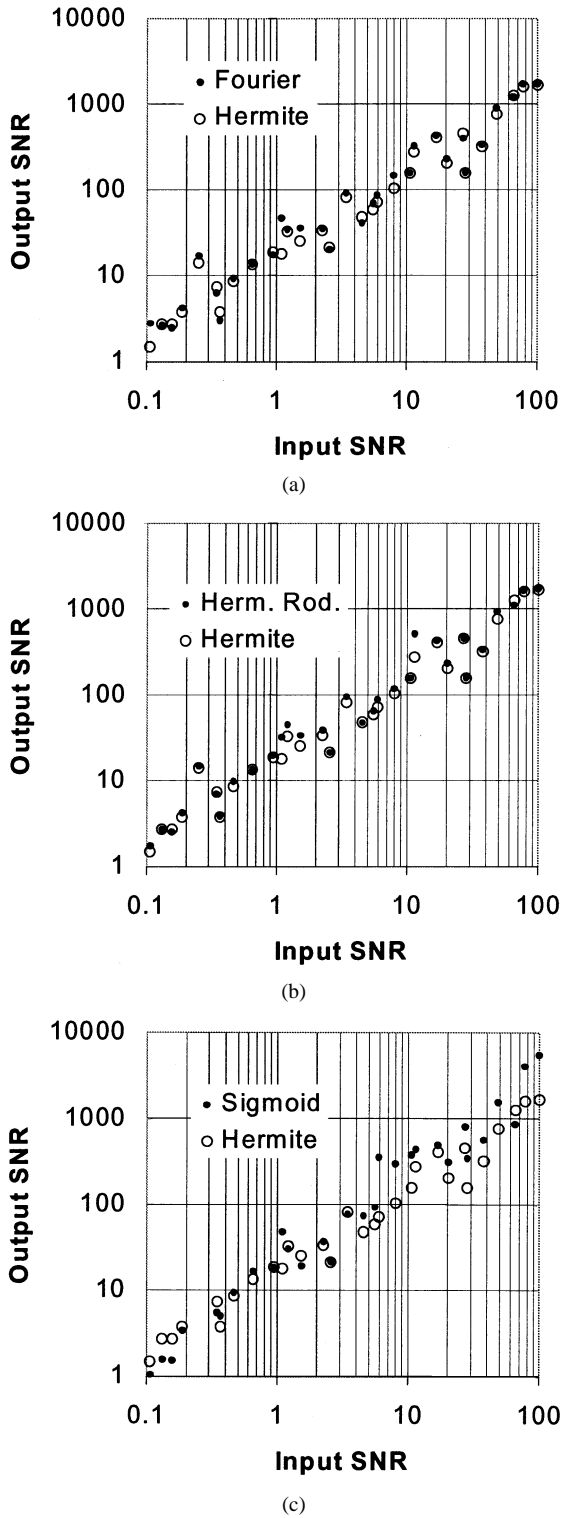


Fig. 6. (a) SNR of demodulated signal by HNN and FNN. (b) SNR of demodulated signal by HNN and HRNN. (c) SNR of demodulated signal by HNN and conventional SNR.

Using scaled Hermite–Rodriguez functions, the correlation with the Gaussian function is

$$hr_n\left(\frac{t}{\alpha}\right) * hr_0\left(\frac{t}{\beta}\right) = \left(\frac{\alpha}{\gamma}\right)^n hr_n\left(\frac{t}{\gamma}\right) \quad (40)$$

where α and β are the scaling factors, and $\gamma^2 = \alpha^2 + \beta^2$.

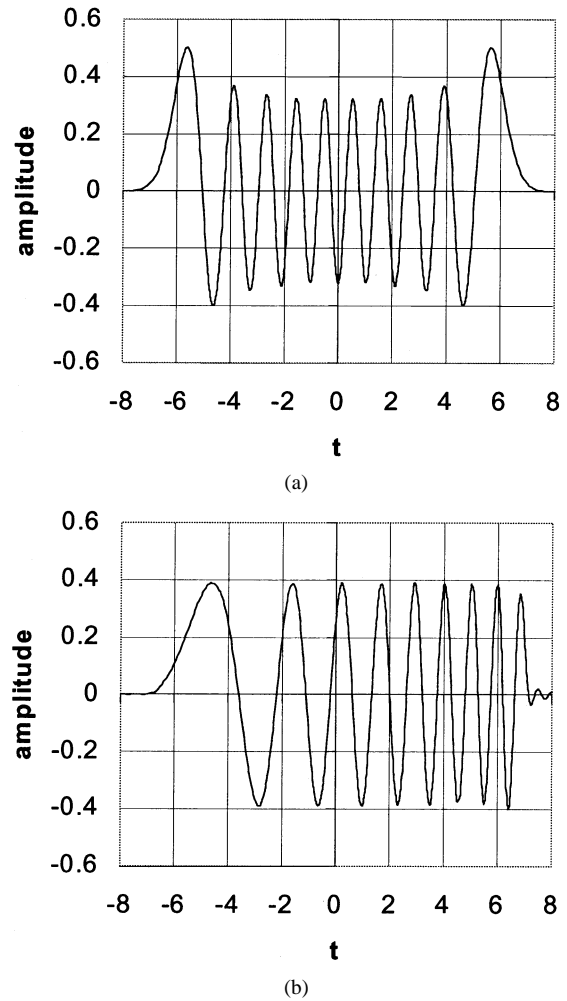


Fig. 7. (a) Hermite signal of order 18. (b) CHIRP signal matched to Hermite function of order 18.

The scaling parameters were chosen as follows. $hr_0(t/\beta)$ was matched to the Gaussian radar pulse and functions as the filter by setting $\beta = \sqrt{2}$. For the purpose of comparison, $hr_n(t/\alpha)$ was chosen to have a width of 10s by setting $\alpha = 5$. Fig. 4(c) shows the root mse of the HRNN interpolation as a function of the number of elements of the series. More than 1500 Hermite–Rodriguez functions are needed before the error drops to a sufficiently small value. The reason for this is that the frequency response of the scaled HRNN is considerably smaller than an HNN of the same duration. Fig. 5(a) shows the frequency response of the HNN on a unit amplitude cosine. Fig. 5(b) shows the response of the HRNN with 70 and 1600 Hermite–Rodriguez functions. The poor frequency response is unavoidable because a sufficiently large α must be chosen to ensure that the series has sufficient duration to interpolate the Doppler signal.

Fig. 6(b) shows the output SNR of the HNN and the HRNN ($N = 1600$) versus the input SNR. Both networks achieve the same SNR, but the HNN is considerably more efficient since it achieves the same result with far fewer terms in the series. In view of their Gaussian window, the HRNN may prove more suitable for correctly analyzing the frequency spectrum of signals subject to glitches.

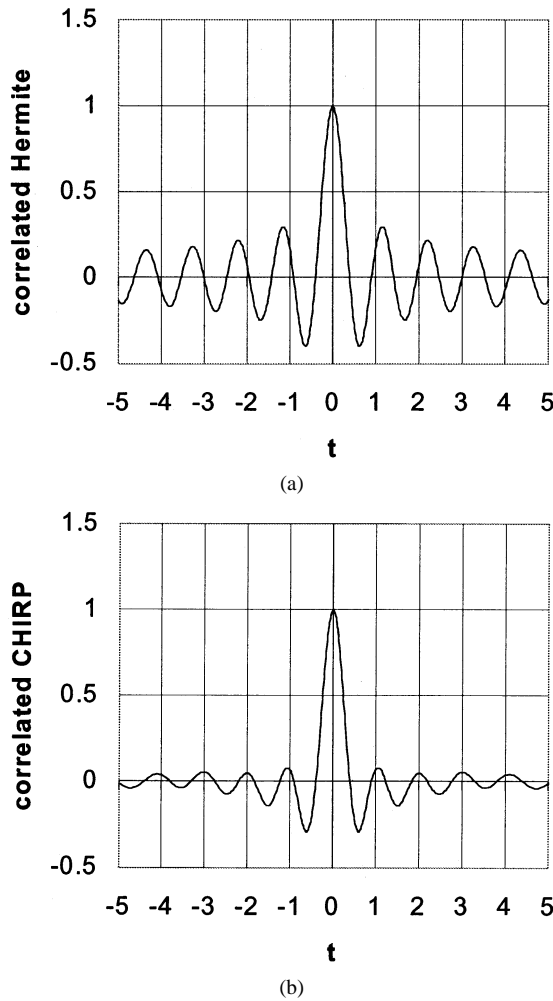


Fig. 8. (a) Correlated Hermite signal of order 18. (b) Correlated CHIRP signal with the width of the first peak matched to the Hermite signal of Fig. 8(a).

The widespread use of the SNN warrants a comparison. Although incapable of performing a Fourier transform, the SNN is capable of demodulating the Doppler signal to obtain the filtered output. This is achieved by forcing the network to respond only to the amplitude modulated part of the signal by deliberately choosing a network with only a few elements. For the Doppler signal, a single layer network of three sigmoid functions performed satisfactorily. Initial weights of the network were randomly chosen, followed by gradient descent to interpolate the Doppler signal. Training was slow with computational time at least ten times greater than the HNN, although the final SNR was slightly greater than the HNN [see Fig. 6(c)]. Faster training algorithms are available for the SNN, but recent research indicates that these may degrade the SNR [25].

B. Correlation of CHIRP Radar Signal

In the second application, we investigate the Hermite modulated radar signal [15]. Conventional frequency-modulated radar uses the CHIRP signal. The CHIRP signal has the desirable characteristic of a long transmitted pulse of high energy that is suitable for long-range measurements, which correlates with itself into a narrow, high-resolution, pulse yielding accurate range measurements. The high-range resolution is

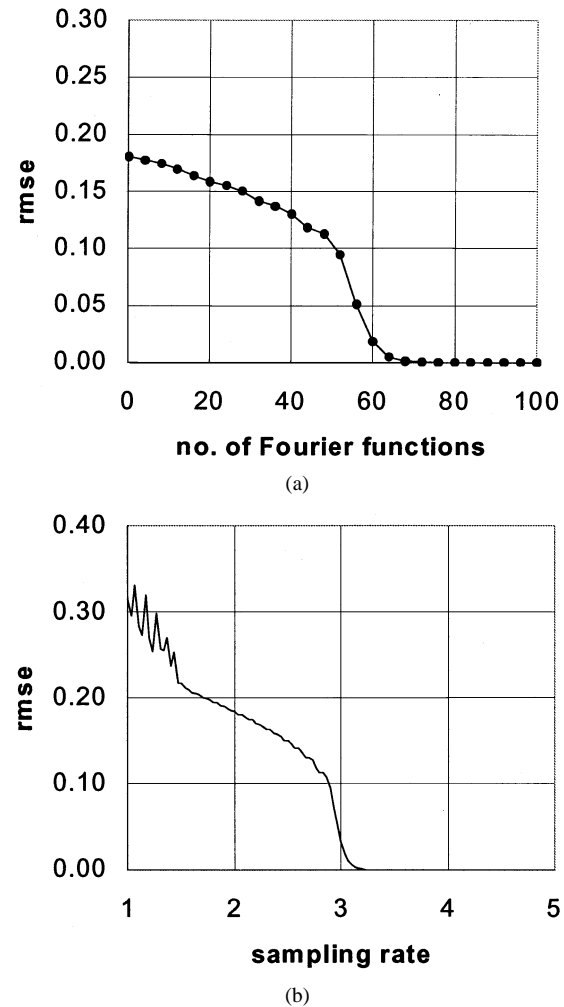


Fig. 9. (a) Root mse of FNN approximation of Hermite signal versus number of Fourier functions. (b) Root mse of FNN approximation of Hermite signal versus sampling rate.

achieved by linearly sweeping the carrier frequency across the width of the transmitted pulse. Although it is an improvement over a nonfrequency modulated signal, the CHIRP signal is still not ideal. If the target is moving, several measurements are required to resolve the velocity and range of the target.

In an effort to develop an improved CHIRP signal, Klauder [15] suggested the Hermite function as a possible modulation signal. Compared with the conventional CHIRP signal, it has a circular ambiguity function enabling the velocity and range of a target to be resolved from a single measurement. The disadvantage is that the Hermite signal has larger side lobes than the CHIRP signal, reducing its performance against multiple targets [see Figs. 7 and 8]. Since the Hermite correlation algorithm is based on the Hermite series, it is ideally suited to processing Hermite signals. To ascertain the performance of the Hermite correlation algorithm, it was compared with a Fourier correlator on simulated, noisy Hermite radar signals.

The Hermite function used for the transmitted/received signal was of order $\{n = 18\}$. This signal has a width $t = 7$ s, and the full width of the first peak of the correlated output is $t = 0.8$ s [see Fig. 7(a)]. Simulated random noise, with a uniform probability density, of varying strength was added to the received Her-

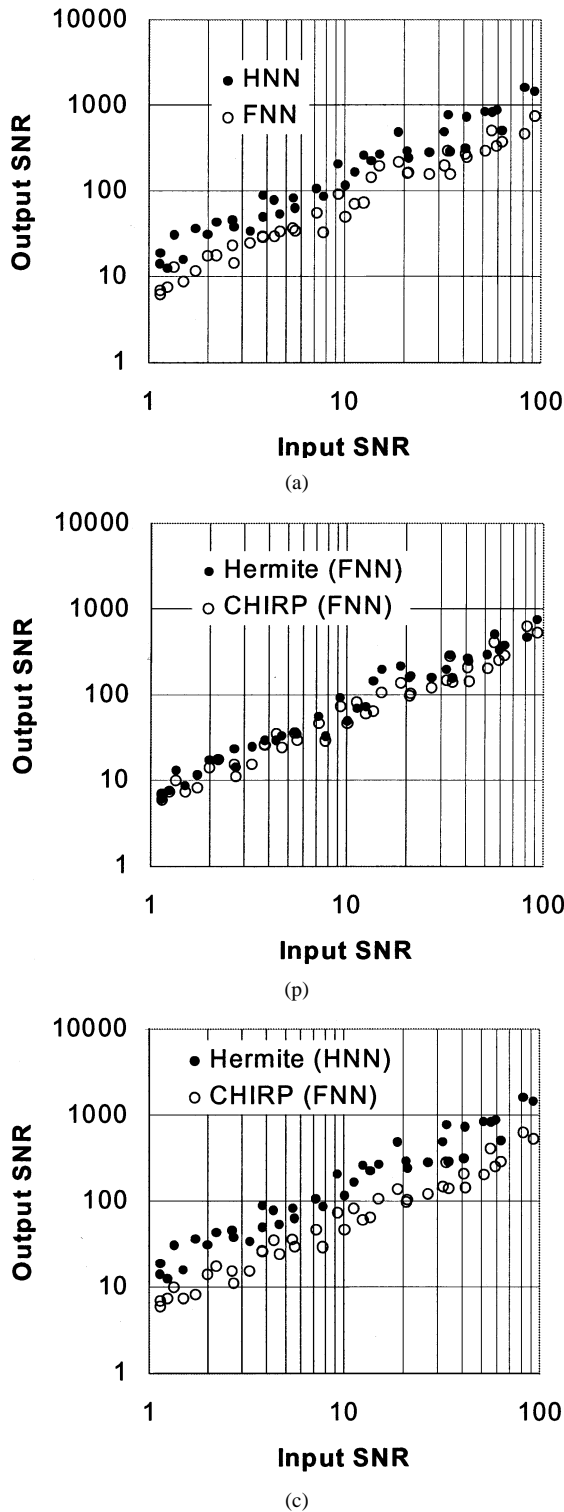


Fig. 10. (a) SNR of Hermite signal with HNN and FNN correlators. (b) SNR of Hermite signal and CHIRP signal with FNN correlators. (c) SNR of Hermite signal with HNN correlator and CHIRP signal with FNN correlator.

mite signal. The Hermite correlator consisted of two networks. The first network contained a single element: the clean, Hermite function just prior to transmission. The second network, consisting of 20 Hermite functions, was used to model the noisy returned signal, shifted in phase in proportion to the target's range. Range information is given by the shift in the peak of the correlated output signal of the two networks.

The Fourier correlator consisted of two Fourier networks. This operated in a similar way to the Hermite correlator, except that the first network, which was fitted to the clean Hermite signal, consisted of more than one element. In order to apply the Fourier network, several parameters were first required; an appropriate sample rate must be chosen, and the number of elements of the network must be large enough to enable a good fit with the Hermite signal. Fig. 9(a) shows the root mse of the Fourier network fitted to the clean Hermite signal. A satisfactory root mse was achieved with 80 Fourier functions. Fig. 9(b) shows the sample rate, which indicates that a rate of at least four samples/s is required; for convenience, a sample rate of 10 samples/s was used.

SNRs were evaluated on the correlated output of both networks. The results are shown in Fig. 10(a). The SNR of the Hermite correlator is superior to the Fourier correlator due to the more efficient modeling of the signal, which allows a lower number of functions to perform the same correlation.

It is also of interest to compare the SNR properties of the Hermite radar signal with a conventional CHIRP signal [see Fig. 7(b)]. For the purpose of comparison, the CHIRP signal power, frequency sweep rate, and duration were carefully chosen to match the Hermite signal as well as the width of the first peak of the correlated Hermite signal as closely as possible [see Fig. 8(a) and (b)]. The CHIRP signal was processed using the Fourier correlator. When the Hermite signal was also processed with the Fourier correlator, the SNRs were similar [see Fig. 10(b)]. However, the SNR of the Hermite signal, processed with the Hermite correlator, was superior [see Fig. 10(c)].

VI. CONCLUSION

An analytical algorithm for computing the correlation of two Hermite neural networks has been developed and compared with other neural networks. Using this algorithm, the correlation of two HNNs, each consisting of N elements, is a summation of N^2 associated Laguerre functions. A summation of N associated Laguerre functions is obtained only for the Gaussian and Hermite CHIRP signals.

Several applications were investigated using simulated random noise. For the demodulation of Doppler radar signals, the HNN correlator achieved the same SNR as an HRNN correlator but was considerably more efficient in terms of the number of network functions required. For this application, the HNN achieved a comparable SNR with that of an FNN of the same size. Compared with a conventional neural network composed of sigmoidal functions, the HNN training time was considerably faster.

The choice between the HNN and FNN correlator is dependent on the signal that is approximated. When applied to Hermite CHIRP radar signals, the HNN correlator provides a higher SNR and a more compact neural network than is possible with an FNN correlator.

REFERENCES

- [1] J.-B. Martens, "The Hermite transform-theory," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1595–1606, Sept. 1990.
- [2] J. Yang and A. Reeves, "Bottom-up visual image processing probed with weighted Hermite polynomials," *Neural Networks*, vol. 8, no. 5, pp. 669–691, 1995.

- [3] Y. V. Venkatesh, K. Ramani, and R. Nandini, "Hermite sieve as a wavelet-like array for 1D and 2D signal decomposition," *Proc. Inst. Elect. Eng. Vis. Image Signal Process.*, vol. 141, no. 5, pp. 348–356, 1994.
- [4] L. R. Lo Conte, R. Merletti, and G. V. Sandri, "Hermite expansions of compact support waveforms: applications to myoelectric signals," *IEEE Trans. Biomed. Eng.*, vol. 41, pp. 1147–1159, Dec. 1995.
- [5] M. Lagerholm, C. Peterson, G. Braccini, L. Edenbrandt, and L. Sornmo, "Clustering ECG complexes using Hermite functions and self-organizing maps," *IEEE Trans. Biomed. Eng.*, vol. 47, pp. 838–848, July 2000.
- [6] M. E. Ulug, "A rapid learning orthonormal neural network for signal processing," in *Proc. Int. Joint Conf. Neural Networks*, vol. 4, June 1992, pp. 265–270.
- [7] S. S. Yang and C. S. Tseng, "An orthonormal neural network for function approximation," *IEEE Trans. Syst., Man, Cybern. B*, vol. 26, pp. 779–784, Oct. 1996.
- [8] S. Osowski, "Neural network for estimation of harmonic components in a power system," *Proc. Inst. Elect. Eng. C*, vol. 139, no. 2, pp. 129–135, 1992.
- [9] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*. New York: Wiley, 1993.
- [10] M. V. Nayakkankuppam and Y. V. Venkatesh, "Deblurring the Gaussian blur using a wavelet transform," *Pattern Recogn.*, vol. 28, no. 7, pp. 965–976, 1995.
- [11] C. Konstantopoulos, L. Mittag, and G. Sandri, "Deconvolution of Gaussian filters and antidiffusion," *J. Appl. Phys.*, vol. 68, no. 4, pp. 1415–1420, 1990.
- [12] A. P. Prudnikov, Yu. A. Brychkov, and O. I. Marichev, *Integrals and Series*. New York: Gordon and Breach, 1986, vol. 2, Special functions, p. 505.
- [13] A. Papoulis, *The Fourier Integral and its Applications*. New York: McGraw-Hill, 1962, pp. 24–25.
- [14] H. S. Stone and L. R. Williams. (1995) On the uniqueness of the convolution theorem for the Fourier transform. NEC Labs. Amer., Princeton, NJ. [Online]. Available: <http://citeseer.nj.nec.com/176038.html>
- [15] J. R. Klauder, "The design of radar signals having both high range resolution and high velocity resolution," *Bell Syst. Tech. J.*, vol. 39, no. 4, pp. 809–820, 1960.
- [16] M. R. Mackenzie and A. K. Tieu, "Gaussian correlation using a Hermite/Laguerre neural network," *IEEE Trans. Neural Networks*, submitted for publication.
- [17] P. A. Lindsay, *Introduction to Quantum Mechanics for Electrical Engineers*. New York: McGraw-Hill, 1967.
- [18] G. G. Walter, *Wavelets and Other Orthogonal Systems with Applications*. Boca Raton, FL: CRC, pp. 86–87.
- [19] W. Magnus, F. Oberhettinger, and R. P. Soni, *Formulas and Theorems for the Special Functions of Mathematical Physics*, 3rd ed. New York: Springer-Verlag, 1966 (obtained by the addition of the second and third recurrence relations on p. 241 and then normalizing the result).
- [20] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. New York: Cambridge Univ. Press, 1992, pp. 178–183.
- [21] V. Chesnokov, "A new adaptive architecture: Analogue synthesiser of orthogonal functions," in *Proc. Inst. Elect. Eng. Conf. Artificial Neural Networks*, 1999, pp. 714–719.
- [22] E. Kreyszig, *Advanced Engineering Mathematics*, 7th ed. New York: Wiley, 1993.
- [23] A. I. Rasiyah, R. Togneri, and Y. Attikouzel, "Modeling 1-D signals using Hermite basis functions," *Proc. Inst. Elect. Eng.—Vis. Image Signal Processing*, vol. 144, no. 6, pp. 345–354, 1997.
- [24] T.-Y. Kwok and D.-Y. Yeung, "Use of bias term in projection pursuit learning improves approximation and convergence properties," *IEEE Trans. Neural Networks*, vol. 7, pp. 1168–1183, Sept. 1996.
- [25] R. Caruana, S. Lawrence, and L. Giles, *Overfitting in Neural Nets: Back-propagation, Conjugate Gradient, and Early Stopping*. Denver, CO: Neural Information Syst., 2000.

Mark R. Mackenzie received the Honors degree in mechanical engineering from the University of Adelaide, Adelaide, Australia, in 1988 and the M.S. degree in mechanical engineering from the University of Wollongong, Wollongong, Australia, in 1995. He is currently pursuing the Ph.D. from the University of Wollongong.

A. Kiet Tieu received the Honors degree in mechanical engineering from the University of Western Australia, Perth, from which he also received the Ph.D. degree.

He was with B.H.P. Steel for eight years and has been with the University of Wollongong, Wollongong, Australia, for 20 years. He is now a Professor and Discipline Leader of Mechanical Engineering at the University of Wollongong.