Deconvolution Neural Networks for Ultrasonic Testing

M.S. Unluturk and J. Saniie

Electrical and Computer Engineering Department
Illinois Institute of Technology, Chicago, IL 60616

Abstract - In this study, three novel design procedures have been developed in implementing deconvolution using neural network algorithms. The first method is called the Deconvolution Neural Network (DNN), the second method is named the Autoassociative Deconvolution Neural Network (ADNN), and the third method is refered to as the Probabilistic Deconvolution Neural Network (PDNN). The DNN trains the network by employing brute force and by exposing the network to a set of target echoes with and without noise. The ADNN processes the data for signal-to-noise ratio enhancement using an Autoassociative neural network, and then applies the deconvolution neural network. The PDNN consists of two processing stages. The first stage estimates parameters using Gram-Charlier approximation to describe the probability density functions corresponding to target echoes and scattering noise. Then, in the second processing block, these parameters are used to classify and detect multiple target echoes. Results obtained in the performance analysis of these algorithms indicate that multiple interfering target echoes can be deconvolved and resolved accurately in the presence of noise.

1. INTRODUCTION

This study presents design techniques for deconvolution neural networks utilizing the backpropagation learning algorithm in order to detect multiple target echoes in noisy environments. A study of this type is useful for detecting ultrasonic flaw echoes in scattering noise. Methods presented here are attractive in the sense that they have the potential to solve deconvolution problems without knowing in advance the solution methodology, and they offer advantages such as real-time processing, adaptability and training capability.

In target detection, the measurement signal is determined in terms of the input (i.e., wavelet) and the impulse response function h(t) (target) through the convolution operation as $y(t) = h(t) * x(t)$. Then, the deconvolution becomes a process of finding a good estimate of h(t) from the knowledge of y(t) and x(t). To find the target impulse response, h(t), the backpropagation learning algorithm is used in designing the deconvolution neural network (DNN). Figure 1 shows a canonic neural network form for deconvolution. Neural networks are, in their most general sense, a collection of various layers of nodes which can be connected in a variety of configurations[1-2]. They have found applications in many areas including ultrasound [3] for detection and characterization. We have designed a three layer fully interconnected neural network, utilizing the backpropagation learning algorithm. Each node consists of the weighted sum of the nodes in the preceding layer passed through the sigmoid function. A set of desired output values is then compared to the actual output of the neural network for every set of input values. The weights are then appropriately updated using the gradient of the output error with respect to the weight value being updated.

An adaptive hidden neuron algorithm is included in the design of the deconvolution neural network (DNN). This adaptive hidden neuron algorithm is promising for determining the optimal number of hidden neurons. The next section provides design techniques concerned with the deconvolution neural network (DNN) in order to achieve a high probability of detection for a reasonably low signal-to-noise ratio. In Section 3, the autoassociative neural network is introduced to improve signal-to-noise ratio, followed by the deconvolution neural network (Autoassociative Deconvolution Neural Network, ADNN). Section 4 presents the Probabilistic Deconvolution Neural Network (PDNN) as an alternative to DNN and ADNN methods.

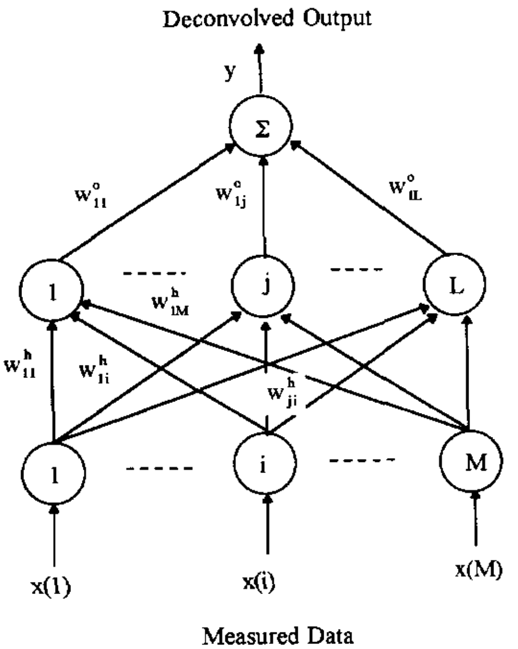

Figure 1. Canonic form for the deconvolution neural network.

0-7803-2940-6/95/$4.00

2. DESIGN OF DECONVOLUTION NEURAL NETWORKS

The deconvolution neural network (DNN) is implemented using the backpropagation algorithm and by exposing the network to a set of target echoes. In detection problems, the received signal, r(n), is given as (see Figure 2)

$$r(n) = \begin{cases} u(n) * \sum_i a_i \delta(n - n_i) + v(n) & \Rightarrow \text{Target+Noise} \\ v(n) & \Rightarrow \text{Noise} \end{cases} \quad (1)$$

where u(n) is the detection wavelet, v(n) is the noise, and a_i is the amplitude (or reflectivity which corresponds to the size of the target) of the target impulse, $\delta(n-n_i)$, detected at n_i (i.e., location of the target).

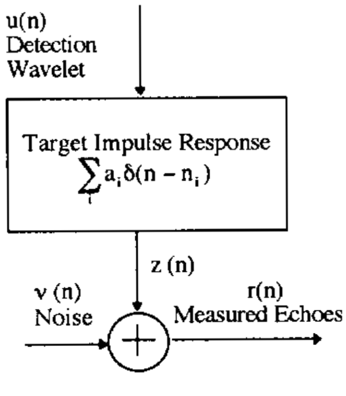

Figure 2. System model of target detection.

Our objective in this study is to detect the location and the size of the target at the output of the deconvolution neural network. To prevent the deconvolution neural network from making a false decision, data normalization is performed to eliminate the effect of signal offsets and measurement scales. The input to the deconvolution neural network can be modeled as a matrix form

$$X = \begin{bmatrix} x(1) & x(2) & \cdots & 0 \\ 0 & x(1) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & x(N) \end{bmatrix} \quad (2)$$

where each column represents one set of normalized input data with an array length of M. Note that a total of M+N-1 set of input data are presented to the deconvolution neural network. Then, the output vector of the deconvolution neural network (see Figure 1), $\bar{y}$, can be estimated

$$\bar{y} = W^o \tanh(W^h X),$$

$$\bar{y} = [0 \cdots 1 \cdots 0], \quad y(i) = \begin{cases} 1, & \text{target + noise} \\ 0, & \text{noise} \end{cases}$$

$$i = 1, \ldots, M + N - 1, \quad (3)$$

$$W^o = [w_{11}^o, \ldots, w_{1j}^o, w_{1L}^o],$$

$$W^h = \begin{bmatrix} w_{11}^h & w_{12}^h & \cdots & w_{1i}^h & w_{1M}^h \\ w_{21}^h & w_{22}^h & \cdots & w_{2i}^h & w_{2M}^h \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ w_{L1}^h & w_{L2}^h & \cdots & w_{Li}^h & w_{1M}^h \end{bmatrix},$$

where W^h and W^o are the weight matrixes for the hidden layer and the output layer respectively.

In deconvolution neural network design, a key issue is determining the number of hidden neurons. Improper selection of hidden neurons results in overfitting and underfitting problems. Overfitting (i.e., too many neurons) performs satisfactory for design data, but fails significantly for test data. Underfitting (i.e., too few hidden neurons) causes unsatisfactory convergence (i.e., DNN is not fully trainable). To avoid problems of overfitting and underfitting an adaptive algorithm for determining the number of hidden neurons is required. The adaptive hidden neuron algorithm starts with three hidden neurons and adaptively increases the number of hidden neurons until convergence is guaranteed. This approach avoids both problems of overfitting and underfitting.

The deconvolution neural network (DNN) is shown in Figure 3. This network has been evaluated for target echo detection. The length of the target echo is 41 samples which is equivalent to the number of input neurons. The number of hidden neurons using adaptive algorithm is found to be 5. Figure 4 shows the testing echo located at 62 without noise. The DNN output is found to be an impulse at 62. Figure 5 shows a testing result for two interfering echoes located at 62 and 64 with signal-to-noise ratio (SNR) of 7.04 dB ($SNR = 20\log_{10}\left(\frac{Max(|z(i)|)}{Max(|v(i)|)}\right)$). As results indicate, DNN locates echoes successfully.

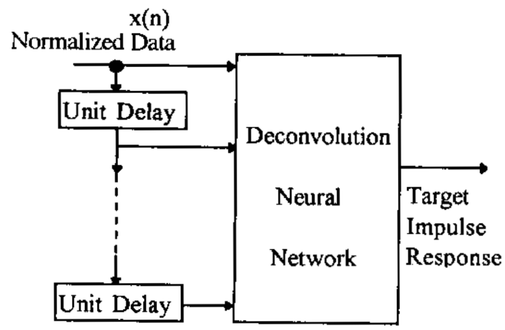

Figure 3. Deconvolution neural network.

The main objective for using DNN is to detect target impulses using ultrasonic measurements. The experimental data is obtained using a 5 MHz broadband transducer a 50 MHz sampling rate for data acquisition. The total number of training vectors are equal to 5975 with an array length of 39 points which is the same as the number of input neurons. The optimal number of hidden neurons is found to be 5 using the adaptive hidden

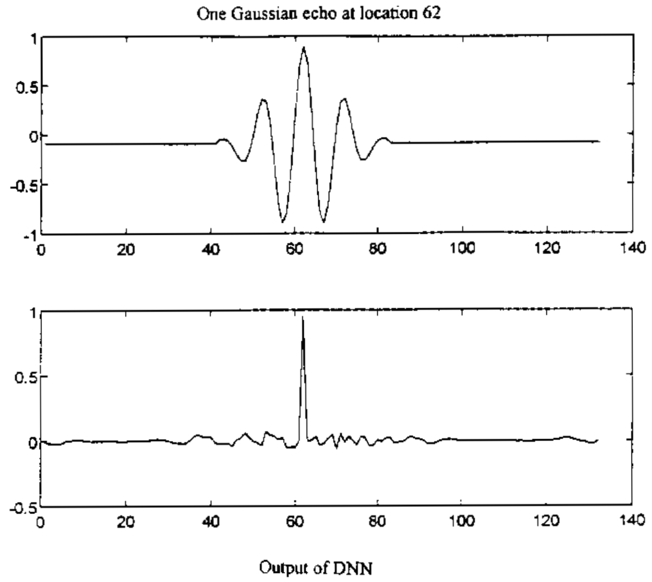

Figure 4. Echo detection using deconvolution neural network.

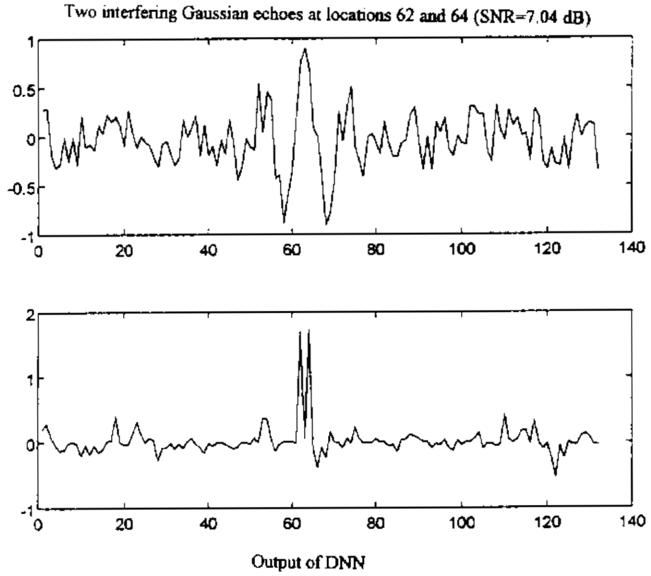

Figure 5. Detection of two interfering echoes using DNN

neuron algorithm. For the echo locations 39, 40 and 42, results are given in Figure 6 (SNR is 16.9). This result demonstrates that DNN achieves a high probability of detection with accurate estimation of echo position when the backscattered signal has SNR of a few decibels.

3. AUTOASSOCIATIVE DECONVOLUTION NEURAL NETWORK

This section introduces the autoassociative neural network in order to improve signal-to-noise ratio (SNR) before applying the deconvolution neural network. The block diagram of the Autoassociative Deconvolution Neural Network (ADNN) is depicted in Figure 7. The number of output neurons in the autoassociative neural network is equal to that of input neurons.

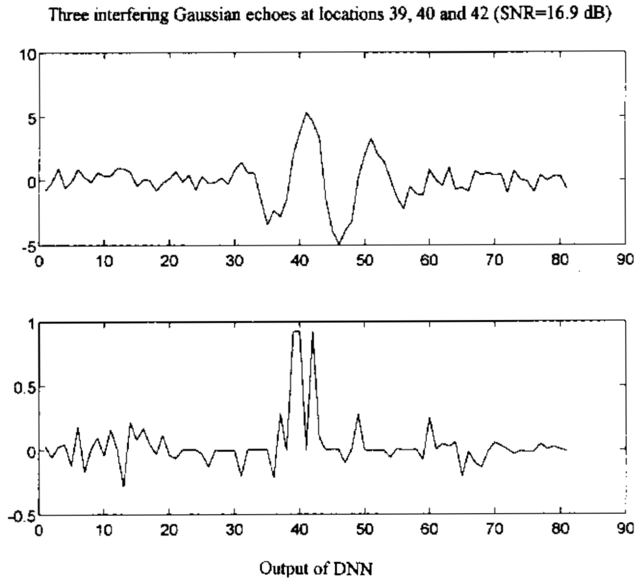

Figure 6. Detecting three echoes using DNN (Experimental data).

After training this neural network, if a testing input resembles one that is used in the training phase, the neural network generates an output which is close to a training output. This means that if a testing input is identical to one of the training patterns and is corrupted by noise, the network filters the noise and outputs the desired echo.

We have evaluated the performance of ADNN using ultrasonic experimental data. Figure 8 displays three target echoes which are located at 78, 80 and 82 (SNR=14.15 dB) and the output of the DNN. Figure 9 shows the output of the same DNN after the SNR enhancement (ADNN). These experimental results demonstrate that SNR enhancement improves the performance of the deconvolution neural network for echo detection. Furthermore, highly interfering echoes can be detected and resolved using ADNN.

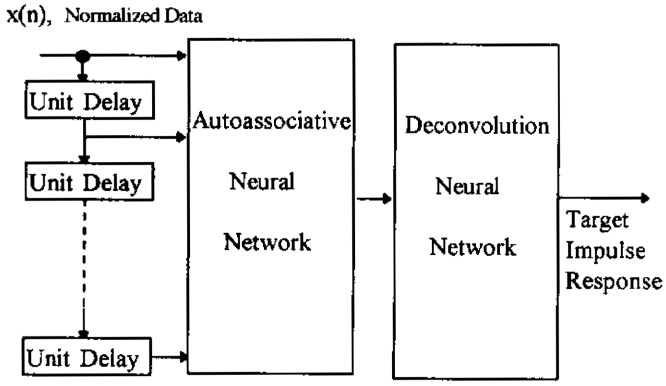

Figure 7. Autoassociative Deconvolution Neural Network.

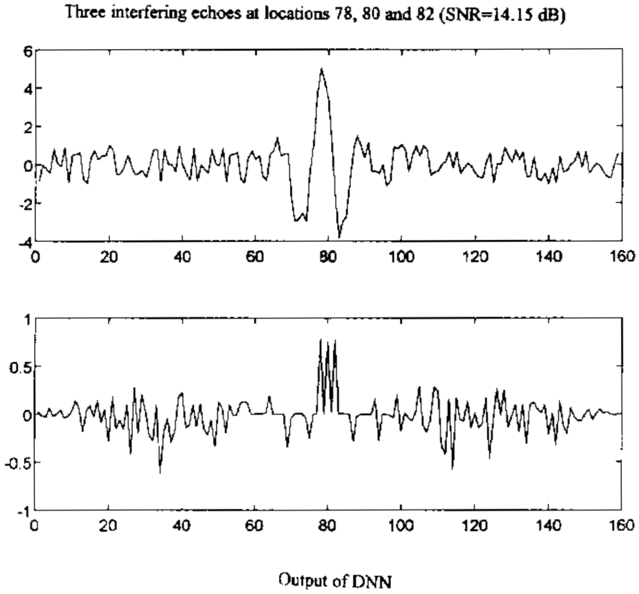

Figure 8. Detecting three echoes using DNN (Experimental data).

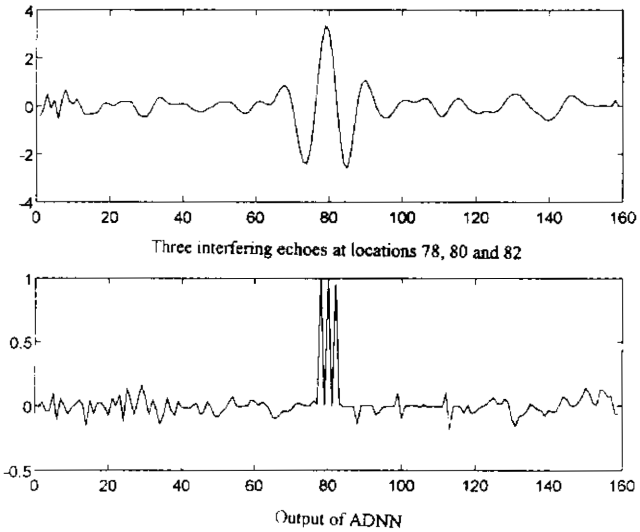

Figure 9. Detecting three echoes using ADNN (Experimental data).

4. PROBABILISTIC DECONVOLUTION NEURAL NETWORK

The Probabilistic Deconvolution Neural Network (PDNN) achieves a high probability of detection for a reasonably low signal-to-noise ratio. The PDNN is realized by utilizing the Gram-Charlier series [4] approximation of the probability density function (pdf). The Gram-Charlier series coefficients are used as the input vector, and the output vector represents impulses corresponding to target echoes.

The Gram-Charlier series expansion of the probability density function of a random variable with mean μ and variance σ^2 can be represented as

$$\rho(x) = \frac{1}{\sigma}\sum_{i=0}^{\infty} c_i \Phi^{(i)}\left[\frac{x-\mu}{\sigma}\right] \qquad (4)$$

where $\Phi(x)$ is a Gaussian probability density function and $\Phi^{(i)}(x)$ represents the i-th derivative of $\Phi(x)$. For normalized data where ($\mu=0$, $\sigma^2=1$), the above equation can be simplified to [4]

$$\rho(x) = \frac{1}{\sigma}\left[\begin{array}{l}\Phi\left(\frac{x-\mu}{\sigma}\right) + c_3\Phi^{(3)}\left(\frac{x-\mu}{\sigma}\right) + c_4\Phi^{(4)}\left(\frac{x-\mu}{\sigma}\right) + \\ c_5\Phi^{(5)}\left(\frac{x-\mu}{\sigma}\right) + c_6\Phi^{(6)}\left(\frac{x-\mu}{\sigma}\right) + \cdots\end{array}\right] \qquad (5)$$

where c_i coefficients are related the central moments of $\rho(x)$. Note that since the data is normalized ($\mu=0$, $\sigma^2=1$) this results in $c_1=0$, and $c_2=0$. Using the above relationship the pdf of target plus noise or noise can be estimated by

$$\begin{aligned}\rho(x) = U(x)\Bigg\{&\left[1 - c_3\left[\left(\frac{x-\mu}{\sigma}\right)^3 - 3\left(\frac{x-\mu}{\sigma}\right)\right]\right] + \\ &c_4\left[\left(\frac{x-\mu}{\sigma}\right)^4 - 6\left(\frac{x-\mu}{\sigma}\right)^2 + 3\right] \\ &-c_5\left[\left(\frac{x-\mu}{\sigma}\right)^5 - 10\left(\frac{x-\mu}{\sigma}\right)^3 + 15\left(\frac{x-\mu}{\sigma}\right)\right] + \\ &c_6\left[\left(\frac{x-\mu}{\sigma}\right)^6 - 15\left(\frac{x-\mu}{\sigma}\right)^4 + 45\left(\frac{x-\mu}{\sigma}\right)^2 - 15\right]\Bigg\},\end{aligned} \qquad (6)$$

$$U(x) = \frac{1}{\left(2\pi\sigma^2\right)^{\frac{1}{2}}\sigma} e^{\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]}.$$

The block diagram for PDNN is shown in Figure 10. Note that the length of the target echo, n, is 39 points. Training of PDNN is achieved using a sliding window from two classes of signals. One class represents the noise only and the other represents target plus noise. The coefficients c_i for $i > 6$ are small compared to other c_i for $i < 6$ [4]. Therefore, the coefficients c_3, c_4, c_5 and c_6 are used as features for recognition using neural networks trained by a backpropagation algorithm. The output of the neural network is 1 representing target plus noise or -1 representing noise only.

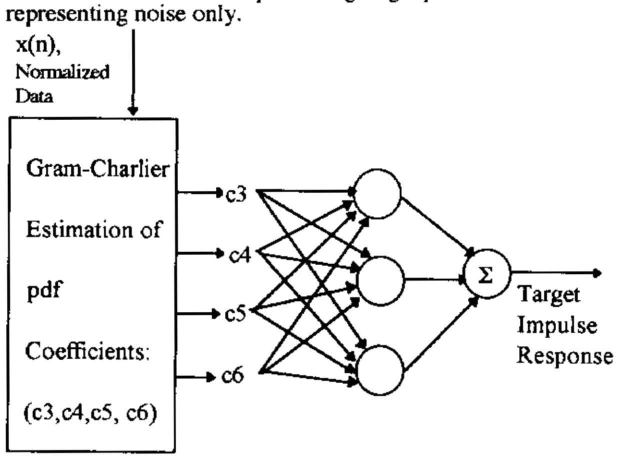

Figure 10. Probabilistic deconvolution neural network.

The performance of PDNN is evaluated using computer simulations. Figure 11 shows a testing result for the three unresolvable echoes whose position are 52, 54 and 56 and SNR is 4.43 dB. This result demonstrates that the PDNN achieves a high probability of detection in the presence of noise. The Gram-Charlier PDNN has been evaluated using experimental ultrasonic data. The data is normalized before being applied to the PDNN. Figure 12 shows the testing result for three interfering echoes (SNR is 13.97 dB) whose positions are 43, 45 and 47. The detection of these echoes is successful leading to the conclusion that the PDNN is an effective method for deconvolving and resolving interfering multiple echoes.

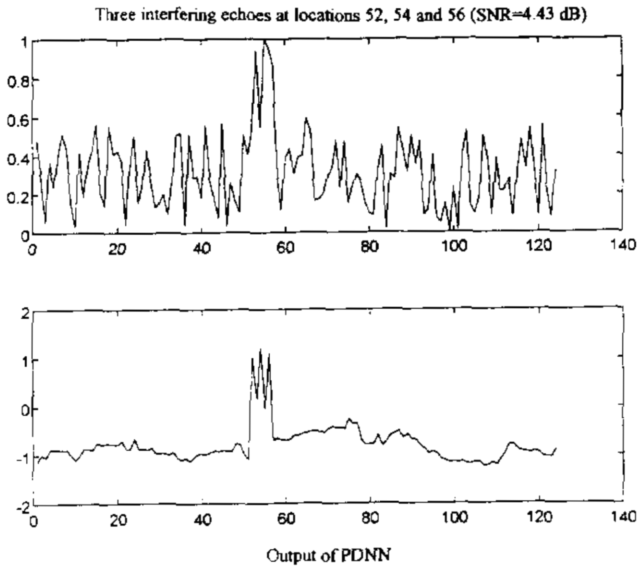

Figure 11. Detecting three echoes using PDNN (Simulated data).

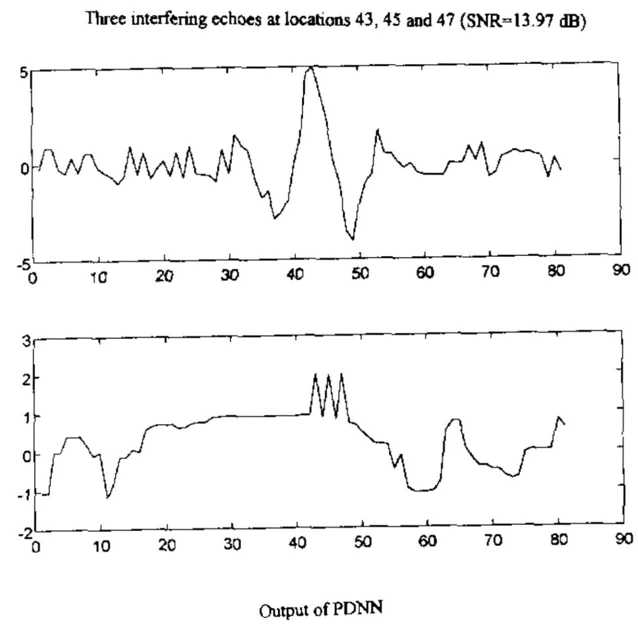

Figure 12. Detecting three echoes using PDNN (Experimental data).

5. CONCLUSION

This study presents three methods of deconvolution neural networks (i.e., DNN, ADNN, and PDNN) in order to detect and resolve multiple interfering target echoes in noisy environments. These deconvolution techniques are desirable in the sense that we do not need to know the solution methodology in advance, and they offer training capability. Results show that the deconvolution neural network recognizes multiple target echoes and locks into the signature of these echoes. Overall, the deconvolution neural network offered a high probability of detection for a reasonably low signal-to-noise ratio.

6. REFERENCES

[1] A. Cichocki and R. Unbehauen, Neural Networks for Optimization and Signal Processing, John Wiley & Sons Publishing Company, 1993.

[2] T. Masters, Practical Neural Network Recipes in C++, Academic Press Publishing Company, 1993.

[3] J. Saniie, M. Unluturk and T. Chu, "Frequency Discrimination Using Neural Networks with Applications in Ultrasonics Microstructure Characterization," IEEE Ultrasonics Symposium Proceedings, pp. 1195-1199, 1992.

[4] M. G. Kendall and A. Stuart, The Advanced Theory of Statistics, Charles Griffin & Company Limited, 1963.