A Novel Thresholding Technique for Adaptive Noise Reduction using Neural Networks

G.Sambasiva Rao* C. NagaRaju Dr.L.S.S. Reddy *** Dr.E.V. Prasad******

* Professor in CSE, St. Ann's College of Engineering & Technology, Chirala-523001
** Professor in CSE, K.L. College of Engineering, Vijayawada-522502
***Principal of K.L. College of Engineering, Vijayawada-522502
**** Principal of J.N.T. University, Kakinad-533003

Abstract
The speckle corrupted image is a traditional problem in both biomedical and in synthetic aperture processing applications, including synthetic aperture radar (SAR). In a SAR image, speckle manifests itself in the form of a random pixel-to-pixel variation with statistical properties similar to those of thermal noise. Due to its granular appearance in an image, speckle noise makes it very difficult to visually and automatically interpret SAR data. Therefore, speckle filtering is a critical preprocessing step for many SAR image-processing tasks, such as segmentation and classification. Wavelet multiresolution analysis has the very useful property of space and scale localization, so it provides great promise for image feature detection at different scales. The recent wavelet thresholding based denoising methods proved promising, since they are capable of suppressing noise while maintaining the high frequency signal details.

However, the local space-scale information of the image is not adaptively considered by standard wavelet thresholding methods. In standard wavelet thresholding based noise reduction methods, the threshold at certain scale is a constant for all wavelet coefficients at this scale. In this paper varies thresholding techniques have been studied for adaptive noise elimination and we presented a new type of thresholding neural network (TNN) structure for adaptive noise reduction, which combines the linear filtering and thresholding methods. This method produced better results than traditional methods.

Keywords:
TNN, wavelet domain, Normal Shrink, RBFN and Soft Thresholding

1.Inroduction

In view of the many theoretical developments that occurred in the last decade, wavelets have found successful applications in a variety of signal processing problems, including image coding and image denoising. Recently, Donoho et al. developed a nonlinear wavelet shrinkage denoising method for statistical applications. The wavelet shrinkage methods rely on the basic idea that the energy of a signal (with some smoothness) will often be concentrated in a few coefficients in wavelet domain while the energy of noise is spread among all coefficients in wavelet domain in [1,2]. Therefore, the nonlinear shrinkage function in wavelet domain will tend to keep a few larger coefficients representing the signal while the noise coefficients will tend to reduce to zero. On the other hand, recent wavelet thresholding based denoising methods proved promising, since they are capable of suppressing noise while maintaining the high frequency signal details in[3,4]. However, the local space-scale information of the image is not adaptively considered by standard wavelet thresholding methods in[5,6]. In standard wavelet thresholding based noise reduction methods, the threshold at certain scale is a constant for all wavelet coefficients at this scale.

2. Speckle Model

Speckle noise in SAR images is usually modeled as a purely multiplicative noise process of the form

$$\begin{aligned} I_S(r, c) &= I(r, c).\ S(r, c) \\ &= I(r, c).\ [1 + T(r, c)] \\ &= I(r, c) + N(r, c) \qquad (1) \end{aligned}$$

The true radiometric values of the image are represented by I, and the values measured by the radar instrument are represented by I_S. The speckle noise is represented by S, and it is modeled as a stationary random process independent of I, with $E[S] = 1$, where $E[\bullet]$ is the expectation operator of [•]. The random process T is zero mean, with variance σ_T and known autocorrelation function $R_{TT} = R_{SS} - 1$. The parameters r and c means row and column of the respective pixel of the image. If

$$T(r, c) = S(r, c) - 1 \qquad (2)$$

$$N(r, c) = I(r, c)\ .\ T(r, c) \qquad (3)$$

we begin with a multiplicative speckle S and finish with an additive speckle N, which avoid the log-transform, because the mean of log-transformed speckle noise does not equal to zero and thus requires correction to avoid extra distortion in the restored image. Eq.(3) represents and additive zero-mean *image-dependent* noise term, which is proportional to the image to be estimated. Since I is non stationary in general, the noise N will be non stationary as well.

Manuscript received December 5, 2008
Manuscript revised December 20, 2008

3. Wavelet-Based Thresholding

Donoho initially proposed denoising based on thresholding in the wavelet domain. Thresholding typically involves a binary decision. The corresponding manipulation of wavelet coefficients usually consists of either "keeping (shrinking)" or "killing" the value of the coefficient. There are two thresholding methods, namely soft and hard thresholding. For each wavelet coefficient, if its amplitude is smaller than a predefined threshold, it will be set to zero (kill); otherwise it will be kept unchanged (hard thresholding), or shrunk in the absolute value by an amount of the threshold (soft thresholding).The key decision in the thresholding technique is the selection of an appropriate threshold. If this value is too small, the recovered image will remain noisy. On the other hand, if the value is too large, important image details will be smoothed out.

The major scheme for recovering original image from its noisy counterpart using the wavelet transform can be summarized by the three primary steps shown in fig. 1.

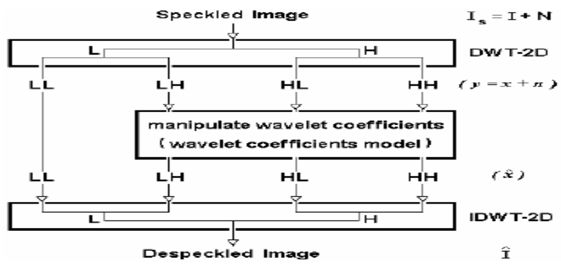

Fig. 1 The wavelet Despeckling procedure with equivalent additive speckle model

1) Calculate the bidimensional DWT (DWT– 2D) of the speckled image
2) Modify the speckled wavelet coefficients according to some rule
3) Compute the inverse of DWT-2D (IDWT-2D) using the modified coefficients

In general manipulating the wavelet coefficients is the most crucial step. One of the major denoising techniques is Thresholding technique.

4. Normalthreshold (Shrink)

A simple and subband adaptive threshold called "*Normal Shrink*" is proposed to address the issue of image recovery form its noisy counterpart. It is based on generalized Gaussian distribution modeling of subband coefficients. The image-denoising algorithm uses soft thresholding to provide smoothness and better edge preservation at the same time. The thresholding with Normal shrink method is computationally more efficient and adaptive because the parameters required for estimating the threshold depends on subband data. The threshold value (T_N) is computed by

$$T_N = \frac{\beta \hat{\sigma}^2}{\hat{\sigma}_y} \quad \text{..........(7)}$$

where the scale parameter β is computed once for each scale using the following equation

$$\beta = \sqrt{\log \frac{L_K}{J}} \quad \text{...........(8)}$$

L_K is the length of the subband at k^{th} scale.

$\hat{\sigma}^2$ is the noise variance which is estimated from the subband HH1, using the formula

$$\hat{\sigma}^2 = \left[\frac{median\ (|Y_{ij}|)}{0.6745} \right]^2, \ Y_{ij} \in \text{subband HH1} \quad (9)$$

and $\hat{\sigma}_y$ is the standard deviation of the subband

This proposed method might be extended to the compression framework, which may further improve the denoising performance.

5. Adaptive Thresholding

The adaptive thresholding method [6,7] is based on the analysis of statistical parameters like arithmetic mean, geometrical mean and standard deviation of the subband coefficients. It is computationally more efficient and adaptive threshold estimation method for image denoising in the wavelet domain based on Generalized Gaussian Distribution modeling of subband coefficients. The noisy image is first decomposed into many levels to obtain different frequency bands. Then soft thresholding method is used to remove the noisy coefficients, by fixing the optimum thresholding value by the proposed method given by

$$T = C\sigma - (|AM - GM|) \quad (10)$$

Here σ is the noise variance of the corrupted image. And is calculated using the formula

$$\hat{\sigma}^2 = \left[\frac{median\ (|Y_{ij}|)}{0.6745} \right]^2 \ Y_{ij} \in \text{subbandHH1} \ (11)$$

Normally in wavelet subbands, as the level increases the coefficients of the subband becomes smoother. For example, HL2 is smoother than the corresponding subband in the first level i.e., HL1. So the threshold value for of HL2 should be smaller than for HL1. the term C is included for this purpose to make the threshold value as decomposition level dependent which is given as

$$C = 2^{(L - k)} \quad (12)$$

Where L is the no. of wavelet decomposition levels, k is the level at which the subband is available (for HH2, k = 2). The term $\left|AM - GM\right|$ is the absolute value of difference between Arithmetic Mean and Geometric Mean of the subband coefficients.

6. Universal Thresholding

Donoho proposed what the wavelet community calls the universal threshold, given by

$$T = \sqrt{2\log(M)}\,\sigma_n \quad (13)$$

Where M is the sample size and σ_n is the noise standard deviation. The universal thresholding technique has been recognized as simple and efficient, but when only a single threshold is used globally, it provides no spatial adaptation during the process of noise suppression. With very large sample size, the universal threshold tends to smear out details.

7. Thresholding Neural Networks

Zhang constructs a type of thresholding neural network (TNN) to perform the thresholding in the transform domain to achieve noise reduction. The neural network structure is shown in fig. 2. The transform in the TNN can be any linear orthogonal transform. The linear transform performed on observed data samples can change the energy distribution of signal and noise samples. By thresholding, the signal energy may be kept while the noise is suppressed. Here, the thresholding functions are employed as nonlinear activation functions of the neural network. The inverse transform is employed to recover the signal from the noise-reduced coefficients in the transform domain. Since most signals have some kind of regularities and the wavelet transform is a very good tool to efficiently represent such characteristics of the signal, the wavelet transform is often a suitable linear transform in TNNs.

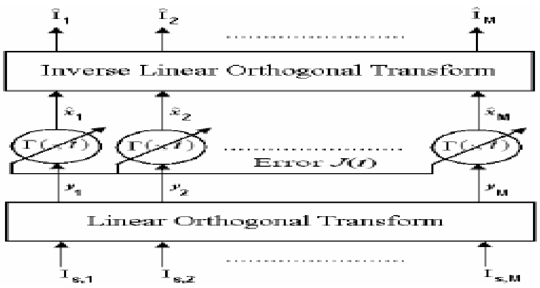

Fig. 2 Thresholding Neural Network

The input to the TNN is noise corrupted signal samples

$$I_{s,i} = I_i + N_i \quad (14)$$

With I = 0, 1, ….,M-1, where I is the true signal and N is the additive noise. The transform in the TNN can be any linear orthogonal transform. For a specific class of signal, the appropriate linear transform may be selected to concentrate signal energy versus noise. Here the thresholding function Γ(x,t) is employed as non-linear activation function of the neural network. The inverse transform is employed to recover the signal from the noise-reduced coefficients in Transform domain. The different thresholds t_j are used in different orthogonal channels and they are independent, i.e., the thresholds of different orthogonal channels can be optimized independently. The TNN is different from conventional multilayer network in the sense that in TNNs, a fixed linear transform is used and the nonlinear activation function is adaptive, while in conventional multilayer neural networks the activation function is fixed and the weights of the linear connection of input signal samples are adaptive.

8. Space-Scale Data Stream Preparation

In the new 2-D adaptive noise reduction method, the 2-D DWT is adopted as the linear transform in TNN and the noise corrupted image *y* is the input of the TNN. To achieve space-scale adaptive noise reduction, we need to prepare the 1-D coefficient data stream, which contains the space-scale information of 2-D images. This is somewhat similar to the "zigzag" arrangement of the DCT (Discrete Cosine Transform) coefficients in image coding applications. In this data preparation step, the 2-D DWT coefficients are rearranged as a 1-D coefficient series in spatial order so that the adjacent samples represent the same local areas in the original image. The DWT of an image consists of four frequency channels: HH, HL, LH and LL. "H" represents high frequency channel and "L" represents low frequency channel. The first letter represents the horizontal direction and the second letter represents the vertical direction. The LL part at each scale is decomposed recursively.

9. Radial Basis Function Network (RBFN)

The RBF Neural Network gained its popularity for its simplicity and speed. RBF is a simple feed forward neural network with only one hidden layer and an output layer. The hidden layer consists of a set of neurons or nodes with radial basis functions as the activation function of the neuron. A Gaussian density function is the most widely used activation function. The output of the layer is a summing unit, which adds up all of the weighted output of the hidden layer.

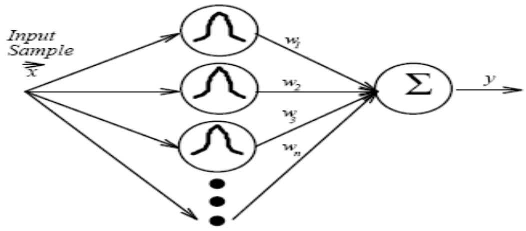

Fig.3 Network structure for RBFNN

Figure above illustrates the RBF Network. The output of the RBF network is given by

$$\hat{y} = f(\vec{x}) = \sum_{k=1}^{N} w_k \phi_k(\vec{x}),$$

where

$$\phi_k(\vec{x}) = (2\pi)^{-p/2} |\Sigma_k|^{-\frac{1}{2}} e^{-\frac{1}{2}(\vec{x}-\vec{c}_k)\Sigma_k^{-1}(\vec{x}-\vec{c}_k)^T}. \quad (15)$$

above N is the number of network nodes, p is the dimensionality of the input space $\vec{x}$ and w_k and $\vec{c}_k$, and Σ_K represent the weight, center and the covariance matrix associated with each node. In the above equation, the output of the network is a scalar quantity and a network can have any number of outputs.

In supervised learning, if ($\vec{x}$, y) is an input output pair, where $\vec{x}$ is the input and y is the desired output, then the network should learn the mapping function f, where

$$y = f(\vec{x}) \quad (16)$$

The training should be done using the M training sample pairs $\left(\vec{x}_1, y_1\right), (\hat{\chi}_2, y_2), \ldots, (\hat{\chi}_M, y_M)$. The output vector containing the M outputs of the network can be written using the matrix form as

$$\hat{\vec{y}} = \phi \vec{w} \quad (17)$$

where, $\hat{\vec{y}}$ is an M dimensional vector, $\vec{w}$ is the N dimensional weight vector. Each column of the ϕ matrix contains the output of a node for all M training samples.

The problem of finding the network weights reduces to finding the vector $\vec{w}$ which makes the network output $\hat{\vec{y}}$ as close as possible to the vector of desired network outputs $\vec{y} = [y_1, y_2, \ldots y_M]^T$. Generally, $\vec{w}$ is determined by finding

$$\Phi \vec{w} = \vec{y} \quad (18)$$

The method for finding the solution to the above equation depends largely on the structure of the network being designed. One popular scheme is to center a node of the network on each of the input training samples ($\vec{c}_k = \vec{x}_k$, k= 1, 2… N). In this case the matrix Φ is a square matrix and the weights are given by $\vec{w} = \Phi^{-1} \vec{y}$.

10. Activation (Threshold) Functions

Currently, three functions are supported: 0. Gaussian = exp(- *v***2 / 2 width**2)

1. Thin Plate Spline = *v***2 * log(*v*)
2. Multi-quadratic = sqrt(*v***2 + width**2)

where **v** is the Euclidian Norm, the distance between the Input vector and the hidden unit Center and is given by

v_i = Sqroot (X - C_i)2
C_i = center vector of ith hidden layer neuron
X = input vector

11. Training

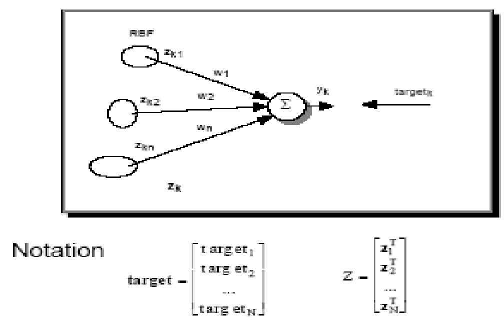

Fig 4. training set notation

The basis weights are set to random values. After the basis weights have been set or trained, a second training phase is required. The learning-rate and momentum settings are used in this phase. Modifying them can have a large effect on the training performance of the network. These values are commonly set from 0.5 to 0.7 for learning rate and 0.0 to 0.9 for momentum. The error tolerance setting controls the final training process. If the data set contains binary targets (0,1), then the tolerance parameter is usually set to 0.1. This means that the output is considered "good" when it is within 0.1 of the desired output (that is, 0.9 for a 1, 0.1 for a 0). When *every* output is within the tolerance range of the desired output value, the network status is changed to LOCKED and weight updates are stopped. The run-time performance of a radial basis network is relatively fast. The input vector is propagated through the network by first multiplying it by basis vectors and then passing it through the basis function. This vector is then multiplied by the output weight matrix. Next it is passed through the selected basis function to produce the network outputs array which is returned to the application program.

12. Experimental Results

The original clean image of "lena.bmp" is used as a test image to illustrate the new method based on the TNN. A noisy image is generated with multiplicative speckle noise. One of them is used as a reference image *y'*. The Daubechies-1 wavelet filter is used. The largest scale of the two-dimensional DWT is set to be 2 in the experiments. The new soft-thresholding function $\eta_\lambda (v,t)$ with λ=0.01 is used. The peak-signal-to-noise-ratio (PSNR) results are shown in Table 1. The first column is the original PSNR of noisy images. The new space-scale adaptive image Denoising method is denoted as "TNN" in the table. For comparison, Table 1 also shows the results of the nonadaptive conventional wavelet denoising schemes. The "VisuShrink" is the universal soft thresholding Denoising technique. The "Normal shrink" represents the Denoising based on generalized Gaussian distribution modeling of sub band coefficients. The Adaptive wavelet thresholding is based on the analysis of statistical parameters like arithmetic mean, geometrical mean and standard deviation of the subband coefficients.

As can be seen, the TNN based space scale adaptive image Denoising has the best performance in terms of PSNR improvement, especially when the PSNR of the original noisy image is high. This can be expected since the amplitudes of the few coefficients representing the signal in transform domain are much higher than those coefficients representing the noise, and then more signal energy can be preserved when cutting off all the coefficients with a threshold.

Table 1

TYPE OF THRESHOLDING	PSNR OF THE NOISY IMAGE (dB)	PSNR OF THE IMAGE AFTER DENOISING (SHRINKING) (dB)
NORMAL SHRINK	13.9899	49.1030
ADAPTIVE WAVELET THRESHOLDING	12.2461	49.1298
UNIVERSAL (VISU SHRINK)	12.2562	49.1343
NEURAL SHRINK	13.8649	49.8442

ORIGINAL IMAGE

NOISY IMAGE

ADAPTIVE WAVELET THRESHOLDING

NORMAL SHRINK

UNIVERSAL THRESHOLD

NEURAL SHRINK

13. Conclusion

In this paper, we have presented a new type of thresholding neural network (TNN) structure for adaptive noise reduction, which combines the linear filtering and thresholding methods. We created a new type of soft and hard thresholding functions to serve as the activation function of TNNs. Unlike the standard thresholding functions, the new thresholding functions are infinitely differentiable. A new practical 2-D space-scale adaptive noise reduction method based on TNN was presented. The learning algorithm proved to be efficient and effective. Numerical examples are given for different noise reduction algorithms including the conventional wavelet thresholding algorithms and linear filtering method. It is shown that the TNN based space-scale adaptive noise reduction algorithm exhibits much superior performance in both PSNR and visual effect.

References

[1] Digital Image Processing using Mat lab – Gonzalez

[2] Tutorial on Wavelets - Robi Polikar

[3] Neural Networks - Simon Haykin

[4] X.-P. Zhang, "Thresholding Neural Network for Adaptive Noise reduction," IEEE Transactions on Neural Networks, vol.12, no. 3, pp.567-584, 2001

[5] Image Denoising using Wavelet Thresholding Lakhwinder Kaur, Savita Gupta, R.C. Chauhan

[6] An Efficient Adaptive Thresholding Technique for Wavelet Based Image Denoising - D.Gnanadurai, and V.Sadasivam; INTERNATIONAL JOURNAL OF SIGNAL PROCESSING VOLUME 2 NUMBER 2 2005 ISSN 1304-4494

[7] D. L. Donoho, "De-noising by soft-thresholding," IEEE Trans. Inform. Theory, vol. 41, no. 3, pp. 613-627, May 1995.

G.Sambasiva Rao received his B.Tech degree in Computer Science from Nagarjuna University Guntur, M.Tech degree in Computer Science from J.N.T.University Anantapur and pursuing his Ph.D from J.N.T.University Kakinada. Currently, he is working as a professor in the Department of Computer Science & Engineering in St. Ann's College of Engineering & Technology, Chirala. He has got 17 years of teaching experience .He has published two research papers in various national and inter national journals and about five research papers in various national and international conferences. He has attended six seminars and workshops. He is member of various professional societies like ISTE, FII and CSI.

C.NagaRaju received his B.Tech degree in Computer Science from J.N.T.University Anantapur, M.Tech degree in Computer Science from J.N.T.University Hyderabad and pursuing his Ph.D in digital Image processing from J.N.T.University Hyderabad . Currently, he is working as a professor in the Department of Computer Science & Engineering in K.L.College of Engineering, Vijayawada. He has got 13 years of teaching experience .He has published six research papers in various national and inter national journals and about twenty four research papers in various national and international conferences. He has attended twenty seminars and workshops. He is member of various professional societies like IEEE, ISTE

Dr.L.S.S.Reddy received his B.Tech degree in Electronics and communication Engineering from J.N.T.University Hyderabad, M.Phil degree in Computer Science from Central University Hyderabad and Ph.D from BITS,PILANI. Currently, he is principal of K.L.College of Engineering, Green Field, Guntur, Andhra Pradesh. He has got more than 20 years of teaching experience .He has published eleven research papers in various national and inter national journals and more than fifty research papers in various national and international conferences. He is Chairman Board of study in Nagarjuna University.He guided four Ph.D scholars and Chairman of CSI Chapter, Vijayawada. He is member of various professional societies like IEEE, ISTE, IETE, IE and CSI.