

# Functional Link Artificial Neural Networks Filter for Gaussian Noise

Guo Yuanhua

School of Automation  
University of Electronic Science and Technology of  
China  
Chengdu, Sichuan, China  
e-mail: gyhua2003@126.com

Huang Chunlun

School of Energy Science and Engineering  
University of Electronic Science and Technology of  
China  
Chengdu, Sichuan, China  
e-mail: clhuang@swu.edu.cn

**Abstract**—In this paper, FLANN(functional link ANN) filter is presented for Gaussian noise. FLANN is a single layer with expanded input vectors and has lower computational cost than MLP(multilayer perceptron). Three types of functional expansion are discussed. BP(back propagation algorithm) for nonlinear activation function and matrix calculation for identical activation function are exploited for training FLANN. Simulation shows that convergence is not guaranteed in BP and related to the initial weight matrix and training images, and that linear FLANN trained by matrix calculation performs better than both nonlinear FLANN trained by BP and Wiener filter in detail region in environment of Gaussian noise

**Keywords**- FLANN, BPNN, Denoising, Gaussian noise

## I. INTRODUCTION

Digital images are usually corrupted by Gaussian noise due to errors arising from the thermal activity of photoelectronic systems. Unlike impulsive noise, it affects all pixels. The presence of noise degrades, blurs images and affects the subsequent image segmentation, feature extraction and recognition, quantitative analysis. So far, various techniques have been developed for Gaussian noise reduction, e.g. linear prediction filters(FIR, Wiener[1]), Wavelet filters[2], filters based partial differential equation[3], anisotropic diffusion filters[4], bilateral filters[5], low pass filters, and so on.

ANN's(artificial neural networks) have become a powerful tool to perform complex tasks in nonlinear environment because of learning capability and approximation of nonlinear functions. ANN's are capable of generating complex mapping between the input and the output space. A large number of filters based on neural networks have been proposed to remove the noise in images[6,7]. The functional link ANN(FLANN) proposed by Pao[8,9] consists of a network without hidden layers exploiting appropriate functional expansions of the input. FLANN may be conveniently used for function approximation and pattern classification with faster convergence rate and lesser computational load than MLP(multilayer perceptron). Sicuranza and Carini[10] introduce a recursive FLANN and present a sufficient stability condition for bounded-input-bounded-output stability. Sicuranza and Carini [11] also propose an extension FLANN including suitable cross-terms. Some applications of FLANN may be found in [12] and [13].

This paper presents FLANN filter for Gaussian noise, the eight elements vector derived from  $3 \times 3$  filter window is functionally expanded as input vector, and the single output is correction of center gray value. Two methods are introduced for training FLANN: BP algorithm and matrix calculation, and trigonometric, exponential and power expansion functions are used. The performance of FLANN has been compared with BPNN( two layers back propagation neural networks) filter and famous Wiener filter.

## II. FUNCTIONAL LINK ANN(FLANN)

### A. Input Vector

Let  $k = [k_1, k_2]$  denote pixel location, and  $x(k)$  denote the luminance value of the noisy image.  $[x_1(k), x_2(k), \dots, x_8(k)]$  is an eight-element observation vector that is the elements of a  $3 \times 3$  window centered around  $x(k)$  (excluding  $x(k)$ ), and  $x_i(k)$  is correspond to a left-to-right, top-to-bottom mapping from the  $3 \times 3$  window to the 1-D vector, as shown in Fig.1. Defining  $X(k)$  an eight-element vector such that

$$X(k) = [x_1(k) - x(k), x_2(k) - x(k), \dots, x_8(k) - x(k)]. \quad (1)$$

For simplicity, writing (1) as

$$X(k) = [x_1(k), x_2(k), \dots, x_8(k)] \quad (2)$$

$x_i(k)$  in (2) is equal to  $x_i(k) - x(k)$  in (1).

$x_1(k)$	$x_2(k)$	$x_3(k)$
$x_4(k)$	$x(k)$	$x_5(k)$
$x_6(k)$	$x_7(k)$	$x_8(k)$

Figure 1. Elements of  $3 \times 3$  window

### B. Structure of FLANN

The functional link ANN(FLANN), originally proposed by Pao[5,6] is a single layer artificial neural network structure capable of performing complex decision regions by functional expansion, and the need of hidden layer is removed. The functional expansion effectively increases the

dimensionality of the input vector and hence FLANN has greater discrimination capability in the input pattern space.

Let us consider a set of basis functions  $B = \{\phi_i \in L(A)\}_{i \in I}$  with the following properties: 1)  $\phi_1 = 1$ , 2) the subset

$B_j = \{\phi_i \in B\}_{i=1}^j$  is linearly independent set, i.e., if  $\sum_{i=1}^j w_i \phi_i = 0$ ,

then  $w_i = 0$  for all  $i = 1, 2, \dots, j$ , and 3)  $\sup_j [\sum_{i=1}^j \|\phi_i\|_A^2]^{1/2} < \infty$ .

Let  $B_N = \{\phi_i\}_{i=1}^N$  be a set of basis functions to be considered for the FLANN. Thus, the FLANN consists of  $N$  basis functions  $\{\phi_1, \phi_2, \dots, \phi_N\} \in B_N$  with the following input-output relationship for the  $j$ th output

$$\hat{y}_j = \rho(S_j); S_j = \sum_{i=1}^N w_{ji} \phi_i(X) \quad (3)$$

where  $X \in A \subset R^n$ , i.e.,  $X = [x_1 \ x_2 \ \dots \ x_n]^T$  is the input vector,  $\hat{y} \in R^m$ , i.e.,  $\hat{y} = [\hat{y}_1 \ \hat{y}_2 \ \dots \ \hat{y}_m]^T$  is the output vector and  $w_j = [w_{j1} \ w_{j2} \ \dots \ w_{jN}]$  is the weight associated with the  $j$ th output of the FLANN. The activation function in second layer is denoted by  $\rho(\cdot)$ .

Considering the  $m$ -dimensional output vector, (3) may be written as

$$S = W\phi \quad (4)$$

where  $W$  is an  $m \times N$  weight matrix of the FLANN given by  $W = [w_1 \ w_2 \ \dots \ w_m]^T$ ,  $\phi = [\phi_1(X) \ \phi_2(X) \ \dots \ \phi_N(X)]^T$  is the basis function vector, and  $S = [S_1 \ S_2 \ \dots \ S_N]^T$  is a matrix of linear sums in second layer. The  $m$ -dimensional output vector  $\hat{y}$  may be given by

$$\hat{y} = \rho(S) \quad (5)$$

Let  $K$  be the number of input vectors applied to the network in a sequence repeatedly, training sequence is denoted by  $\{X(k), y(k)\}$ , and corresponding expansion vector is denoted by  $\phi(X(k))$  or  $\phi(k)$ .

In our simulation the functional expansion includes trigonometric, exponential and power series. The  $k$ th enhanced input vector is obtained by using a trigonometric function as

$$\phi(k) = [1 \ x_1(k) \sin(x_1(k)) \cos(x_1(k)) \sin(\pi x_1(k)) \cos(\pi x_1(k)) \dots] \quad (6)$$

Using exponential expansion and power expansion the enhanced will be

$$\phi(k) = [1 \ x_1(k) \exp(x_1(k)) \exp(2x_1(k)) \dots x_2(k) \exp(x_2(k)) \exp(2x_2(k)) \dots], \quad (7)$$

$$\phi(k) = [1 \ x_1(k) \ x_1(k)^2 \ x_1(k)^3 \dots x_2(k) \ x_2(k)^2 \ x_2(k)^3 \dots]. \quad (8)$$

### C. Learning Algorithm

If the activation function associated with nodes with second layer is the tanh function given by  $\rho(S) = \tanh(S) = (1 - e^{-2S}) / (1 + e^{-2S})$ . The partial derivative of

$\rho(S)$  with respect to  $S$  is denoted by  $\rho'(S)$  and is given by  $\rho'(S) = (1 - \rho^2(S))$ . Adopting the BP algorithm for a single layer, the update rule for the weights of FLANN is given by

$$w_{ij}(k) = w_{ij}(k-1) - \mu(1 - S_i(k)^2)e_i(k)\phi_j(k) \quad (9)$$

where  $\mu$  denote learning rate, and  $e_i(k)$  is the  $i$ th output error such as  $\hat{y}_i - y_i$ .

If the activation function in second layer is identical mapping, the weights can be evaluated through matrix calculation. Error function may be defined as

$$E(W) = \frac{1}{2} \sum_{k=1}^K \sum_{i=1}^m (\sum_{j=1}^N w_{ij} \phi_j(k) - y_i(k))^2 \quad (10)$$

Minimum of  $E(W)$  is obtained by the method of least squares. According to necessary condition of multi-function, we have

$$\frac{\partial E(W)}{\partial w_{ij}} = 0, i = 1, \dots, m, j = 1, \dots, N, \quad (11)$$

$$\frac{\partial E(W)}{\partial w_{ij}} = \sum_{k=1}^K (\sum_{l=1}^N w_{il} \phi_l(k) - y_i(k)) \phi_j(k), \quad (12)$$

$$\sum_{k=1}^K (\sum_{l=1}^N w_{il} \phi_l(k) - y_i(k)) \phi_j(k) = 0 \quad (13)$$

or

$$\sum_{l=1}^N w_{il} (\sum_{k=1}^K \phi_l(k) \phi_j(k)) = \sum_{k=1}^K y_i(k) \phi_j(k), \quad i = 1, \dots, m; l, j = 1, \dots, N \quad (14)$$

Writing in matrix form

$$W \cdot (\phi^T \phi) = Y^T \cdot \phi \quad (15)$$

where

$$W = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1N} \\ w_{21} & w_{22} & \dots & w_{2N} \\ \dots & \dots & \dots & \dots \\ w_{m1} & w_{m2} & \dots & w_{mN} \end{bmatrix}, \quad (16)$$

$$\phi = \begin{bmatrix} \phi_1(1) & \phi_2(1) & \dots & \phi_N(1) \\ \phi_1(2) & \phi_2(2) & \dots & \phi_N(2) \\ \dots & \dots & \dots & \dots \\ \phi_1(K) & \phi_2(K) & \dots & \phi_N(K) \end{bmatrix}, \quad (17)$$

$$Y = \begin{bmatrix} y_1(1) & y_2(1) & \dots & y_m(1) \\ y_1(2) & y_2(2) & \dots & y_m(2) \\ \dots & \dots & \dots & \dots \\ y_1(K) & y_2(K) & \dots & y_m(K) \end{bmatrix}. \quad (18)$$

$\phi^T \phi$  is an  $N \times N$  matrix, if not singular matrix, denoting

$$\phi^+ = \phi \cdot (\phi^T \phi)^{-1}. \quad (19)$$

Then

$$W = Y^T \cdot \phi^+. \quad (20)$$

If  $\phi^T \phi$  is singular, defining

$$\phi^+ = \lim_{\varepsilon \rightarrow 0^+} \phi.(\phi^T \phi + \varepsilon I)^{-1} \quad (21)$$

where  $I$  is an identity matrix, it has been proved that the limit always exists.

### III. SIMULATION

Images with Gaussian additive white noise were employed to train the neural network and test. In this noise environment the add or subtract of values in pixel is independent of the values taken by image. Famous test images lena, boats and sail, with  $256 \times 256$  sizes, which are shown in Fig.1, are adopted.  $3 \times 3$  filter window is used in FLANN and BPNN, image borders are ignored, then total number of network input vector is  $253 \times 253$ . Lena is selected for training image for Lena is rich in different patterns.

We compare the proposed network with classical BPNN(multilayer feed forward networks trained with back propagation). We set BPNN to be  $\{9-4-1\}$ , the activation functions of first and second layer are logsig function and tanh function, respectively, the learning rate is 0.05 and iteration number is 50. We adopt the back propagation algorithm for training BPNN. In FLANN (2) is expanded two times using different expansion, so the input vector has 25 elements including threshold input, the structure is  $\{25-1\}$  and the learning rate is also 0.05. In FLANN if the activation function is tanh or identical mapping, we adopt the back propagation algorithm or matrix calculation, respectively. We also compare FLANN with Wiener filter and median filter, and use Matlab function wiener2() and medfilt2() to realize them. In all train and test gray values were normalized to fall with in the interval of  $[0,1]$ . Gaussian noise with zeros mean and 0.005 variance was added, and Fig. 2(a) shows the degraded Sail image. Table 1 shows the simulation result in term of PSNR. For each ANN

highest PSNR in Table 1 was selected from 10 independent training. FLANN1, FLANN2 and FLANN3 stand for FLANN with tanh activation function having trigonometric expansion, exponential expansion and power expansion, respectively. FLANN4, FLANN5 and FLANN6 stand for FLANN with identical function having trigonometric expansion, exponential expansion and power expansion, respectively. Convergence of  $W$  in FLANN with tanh function is not guaranteed and related to training image and initial  $W$ . Simulation shows that  $W$  closed to zero matrix is more likely to convergent. Each FLANN shows better result than BPNN and median filters, FLANN1-FLANN3 with identical activation function better than FLANN4-FLANN6 with nonlinear activation function, as can be noticed that Fig. 2(b) is more blurred than Fig. 2(c), and FLANN with exponential expansion poorer than FLANN with trigonometric and power expansion. FLANN performs better than Wiener filter in boats and poorer in sail, so FLANN is largely equivalent to Wiener in term of PSNR. Wiener filter in wiener2() is a locally adaptive linear filter[14], referenced widely, and can be expressed as:

$$y(k) = \bar{x}(k) + \frac{\sigma(k)^2 - \sigma^2}{\sigma(k)^2} (x(k) - \bar{x}(k)) \quad (22)$$

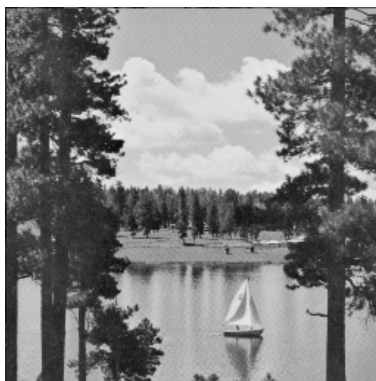
where  $\bar{x}(k)$ ,  $x(k)$ ,  $\sigma(k)^2$  and  $\sigma^2$  stand for the local mean, the center pixel value, the local variance and noise variance, respectively. It is evident that for  $\sigma_k^2 \approx \sigma^2$ ,  $y(k) \approx \bar{x}(k)$ . Alternatively, for  $\sigma_k^2 \gg \sigma^2$ ,  $y(k) \approx x(k)$ . Consequently Wiener filter provides heavy filtering in smooth regions and light filtering in edge regions., hence the noise in the neighborhood of edges is almost preserved. Compared to Wiener, FLANN deals with edge regions well, as can be noticed between Fig. 2(c) and Fig. 2(e), in Fig. 2(e) leaf edge regions details are preserved but the noise is visible.

	Noise	FLANN1	FLANN2	FLANN3	FLANN4	FLANN5	FLANN6	BPNN	Wiener	median
Boats	22.5026	26.0622	25.3083	26.1907	26.5853	26.5790	26.4807	24.7257	26.4118	24.5013
Sail	22.4394	25.5055	24.4767	25.6593	26.1184	26.0022	26.1121	24.8591	26.3882	23.9140

Table 1. PSNR results filtered by various filters for images degraded by Gaussian noise with 0.005 variance



(a) Lena

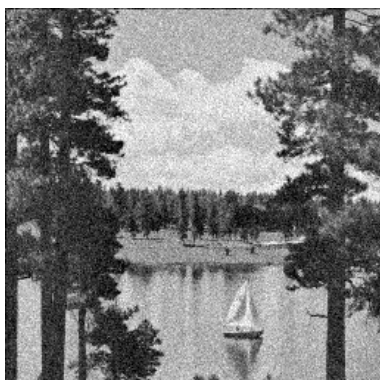


(b) Sail



(c) Boats

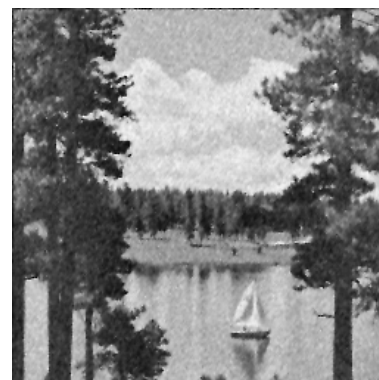
Figure 1. Training and test images



(a) Degraded Sail



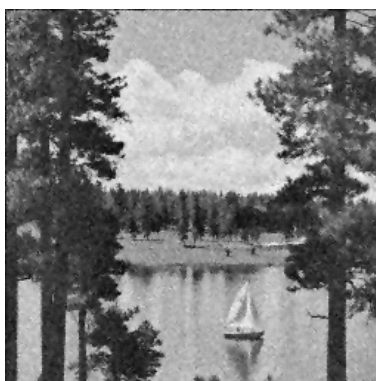
(b) (a) filtered by FLANN1



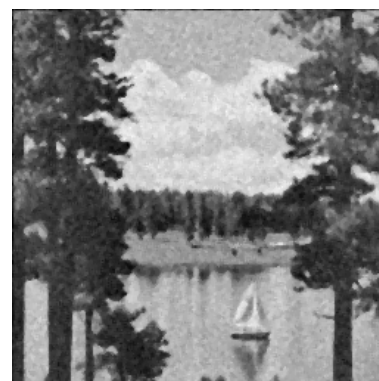
(c) (a) filtered by FLANN4



(d) (a) filtered by BPNN



(e) (a) filtered by Wiener filter



(f) (a) filtered by median filter

Figure 2. Degraded Sail and filtered results

#### IV. CONCLUSIONS

Here FLANN is proposed for denoising of degraded images with Gaussian noise. FLANN is single layer with expanded input vector. The functional expansion increases

the dimensionality of input vector, which is analogous to the hidden layer of BPNN, hence FLANN has greater discrimination capability. Here trigonometric expansion, exponential expansion, power expansion, tanh() activation function and identical activation function are tested.

Nonlinear and linear weight matrix  $W$  are obtained by BP algorithm and matrix calculation, respectively. Simulation in Gaussian environment shows that identical function has lower computational expense, is more effective and more stable than  $\tanh()$  function. Each FLANN is more effective than BPNN and median filter. FLANN with identical function is near equal to Wiener filter in term of PSNR and performs better in detail regions.

#### REFERENCES

- [1] J. S. Goldstein, I.S. Reed and L. L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Transactions on Information Theory*, vol.44, no.7, pp.2943-2959, Nov. 1998.
- [2] S.G.Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. PAMI*, vol.11, no.7, pp.674-693, July, 1989.
- [3] M. Lysaker, A. Lundervold and X.C. Tai, "Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time," *IEEE Trans. on Image Processing*, vol.12, no.12, pp.1579-1590, Dec. 2003.
- [4] M.J.Black, G.Sapiro, D.H.Marimont and D.Heeger. "Robust anisotropic diffusion," *IEEE Trans. on Image Processing*, vol.7, no.3, pp. 421-432, Mar. 1998.
- [5] C.Tomasi and R.Manduchi, "Bilateral filtering for gray and color images," in *Proc. Of the 1998 IEEE Int. Con. on Com. Vision*, Bombay, India, pp.839-846.
- [6] L. Corbalan, G.Osella Massa, C.Russo, L.Lanzarini and A. De Giusti. "Image recovery using a new nonlinear adaptive filter based on neural networks," in *28th Int. Conf. Information Technology Interfaces*, Cavtat, Croatia, June 19-22, 2006, pp.355-360.
- [7] T.A.Cheema, I.M.Qureshi and A.Hussain, "Blind image deconvolution using space-variant neural network approach," *Electronics Letters*, vol.41, no.6, pp.308-309, Mar. 2005.
- [8] Y.H.Pao. *Adaptive Pattern Recognition and Neural Networks*. Reading, MA: Addison-Wesley, 1989.
- [9] Y.H.Pao, S.M.Phillips and D.J.Sobajic, "Neural-net computing and the intelligent control systems," *Int. J. Contr.*, vol.56, no.2, pp.263-289, 1992.
- [10] G. L.Sicuranza and A. Carini, "Adaptive recursive FLANN filters for nonlinear active noise control," in *ICASSP 2011*, pp.4312-4315.
- [11] G. L.Sicuranza and A. Carini, "A generalized FLANN filter for nonlinear active noise control," *IEEE Trans. on Audio, Speech and Language Processing*, vol.19, no.8, Nov. 2011.
- [12] W.D.Wang and C.T.Yen, "Reduced-decision feedback FLANN nonlinear channel equalizer for digital communication systems," *IEE Proc. Commun.*, vol.151, no.4, pp.305-311, Aug. 2004.
- [13] J.C. Patra, N.C.Thanh and P.K.Meher, "Computationally efficient FLANN-based intelligent stock price prediction system," in *Proceedings of International Joint Conference on Neural Networks*, Atlanta Georgia, USA, June 14-19, 2009, pp.2431-2438.
- [14] J.S.Lee. "Digital image enhancement and noise filtering by use of local statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-2, pp.165-168, Mar. 1980.

Guo Yuanhua received Ph.D degree in automated reasoning from east china normal university and now does research about image noise suppression. His interests include mechanical theorem-proving, linux kernel, artificial intelligence, image filtering, image enhancement and other research areas about digital image processing.

Huang Chunlun is currently pursuing the Ph.D. degree at University of Electronic Science and Technology of China, and he does research about image noise suppression and pattern recognition.