Research Article

A PCA and ELM Based Adaptive Method for Channel Equalization in MFL Inspection

Zhenning Wu,[1] Huaguang Zhang,[1] Jinhai Liu,[1] Zongjie Qiu,[2] and Mo Zhao[1]

[1] *School of Information Science and Engineering, Northeastern University, Shenyang, Liaoning 110004, China*
[2] *CNOOC (China) Co., Ltd., Beijing 100010, China*

Correspondence should be addressed to Zhenning Wu; wuzn2003@hotmail.com

Magnetic flux leakage (MFL) as an efficient method for pipeline flaw detection plays important role in pipeline safety. This nondestructive test technique assesses the health of the buried pipeline. The signal is gathered by an array of hall-effect sensors disposed at the magnetic neutral plane of a pair of permanent magnet in the pipeline inspection gauge (PIG) clinging to the inner surface of the pipe wall. The magnetic flux measured by the sensors reflects the health condition of the pipe. The signal is influenced by not only the condition of the pipe, but also by the lift-off value of the sensors and various properties of electronic component. The consistency of the position of the sensors is almost never satisfied and each sensor measures differently. In this paper, a new scheme of channel equalization is proposed for MFL signal in order to correct sensor misalignments, which eventually improves accuracy of defect characterization. The algorithm proposed in this paper is adaptive to the effects of error on the disposition of the sensor due to manufacturing imperfections and movements of the sensors. The algorithm is tested by data acquired from an experimental pipeline. The results show the effectiveness of the proposed algorithm.

1. Introduction

MFL is a widely used nondestructive testing (NDT) methods for pipeline inspection. The inspection machine is usually called PIG. A PIG using MFL consists of several pairs of strong permanent magnets which magnetize the pipe along the axial direction of the pipe. Each pair of the strong permanent along with the yoke iron and the hall sensors and also brush is called a carrier. The carrier with the pipe consists of a magnetic circuit. Details of the PIG can be found at some famous inspection companies website, ROSEN, PII, and so forth. A lot of research work has been done to analyze the signal of MFL. Carvalho et al. [1], Christen and Bergamini [2], and Xiang and Tso [3] purposed neural network based methods to detect flaws. Mukhopadhyay and Srivastava [4] proposed wavelet based technique to denoise the signal of the MFL inspection. Mukherjee et al. proposed wavelet based inverse mapping system [5]. Kathirmani et al. [6] using PCA [7] and wavelet [8–11] technique to compress data of the MFL signal.

But there is still one problem that needs to be studied. The sensing arrangement, that is, each sensor, its mechanical support, and the underlying electronics for acquiring the magnetic leakage flux data, commonly referred to as a channel, suffers mismatch among each other. A lot of factors can cause channel-to-channel mismatch, including the lift-off value between the pipeline, and position of hall and coil sensors and the various properties of electronic component. Other factors influence the factors mentioned above can also impact the output of the signal, such as the difference of the sensors location caused by assembly, the shake when the detector running in the pipeline, and so forth. All these factors make the output of the signal different even under same testing condition and testing object. This will lower the capacity of the detector especially during the post processing of the signal. Such kind of mismatch may also exist in other multisensor data processing. Commonly, it can be equalized by using adaptive techniques. An adaptive channel equalization algorithm to deal with the problem of channel-to-channel mismatch of MFL signals is given in

[12]. In [12], they assume that at least one sensor out of the sensor array is ideal and can be used as a reference. For implementation, this assumption imposes serious limitations on the performance of post processing algorithm as tolerance and misalignment of an individual sensor is not deterministic and needs to be accounted for in a stochastic framework for choice of a clear cut candidate qualifying as a reference channel. To solve the problem, Mukherjee et al. [13] gives an adaptive method which does not need to choose a reference channel. In [13], the reference channel required for channel equalization is replaced with the baseline estimation. The baseline estimation reflects the background leakage flux. But the baseline estimation is not available under a defect or feature. It is estimated by first order forward or back prediction of the neighboring MFL data.

In this paper, a new adaptive channel equalization algorithm to minimize channel-to-channel mismatch is proposed for MFL signals. In contrast to [12] our algorithm does not need to choose any reference channel. Because the ideal reference channel is not easy to get and the character of each channel may have little difference, equalizing the channels adaptively with no reference channel is reasonable. The signal of all channel is learned by a neural network. The training data set is selected as a clean pipe (almost no flaw), which reflects the character of the pipe. And distinguished from [13], we mainly focus on the signal around and including the flaw. In [13], the signal around the flaw is only estimated by first order forward or backward prediction of the neighboring MFL data, which may cause distortion of the flaw signal. As flaw evaluation needs the exact shape of the signal around and including the flaw, our algorithm has more advantage. A PCA based flaw detection algorithm is given in this paper to find the location of the flaw signal. A median corrected algorithm is also given from the engineering point of view. The simulation results show that the algorithm proposed in this paper is efficient.

The paper is organized as follows. In Section 2, an ELM based method is given to dynamically compensate each channel, and also with a PCA based statistic method to separate normal and flaw signal and extract signal characters. The details of our algorithm proposed in this paper are stated too. A median corrected algorithm is given, and simulation results are shown in Section 3. Section 4 concludes the paper indicating major achievements and future scope of this work.

2. PCA and ELM Based Channel Equalization

2.1. Channel Equalization Using ELM Neural Networks. Neural networks are very efficient and popular tool to do regression and classification. The advantages of the neural networks are mainly two points: one is that the model of the data does not need to be known, the other is its high capability to deal with nonlinear problems. There exist many types of neural networks; however, feedforward neural networks may be one of the most popular neural networks. The feedforward neural network usually consists of one input layer receiving the stimulin from external environments, one or multihidden layers, and one output layer sending the
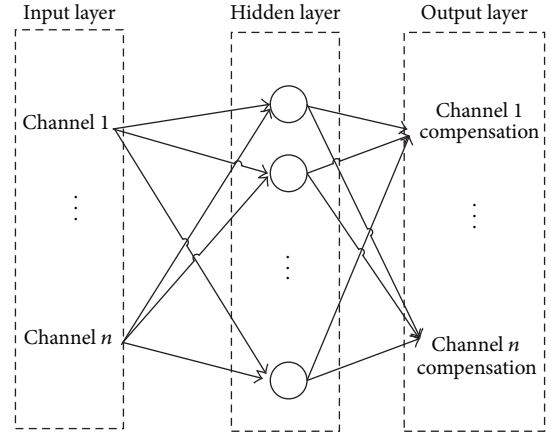

FIGURE 1: Neural network model for channel equalization.

network output to external environments. Widely used neural networks include backpropagation (BP) neural network [14], radial basis function (RBF) neural network [15], and support vector machine (SVM) [16]. Three main approaches are usually used to train feedforward networks including gradient-descent based method (e.g., BP neural networks), least-square based method (e.g., RBF network learning) and standard optimization method based method (e.g., SVM). Different from traditional learning algorithms, ELM [17] tends to reach not only the smallest training error but also the smallest norm of output weights. According to the neural network theory, for feedforward neural networks, smaller training error results in smaller norm of weights and better generalization performance. Since the hidden layer needs not be tuned in ELM and the hidden layer parameters can be fixed, the output weights can then be resolved using the least-square method. The model of channel equalization using ELM is given as Figure 1.

The output function of single-hidden layer feedforward networks (SLFNs) with L hidden nodes can be represented by

$$f(x) = \Sigma_{i=1}^{l} \beta_i g_i(x), \tag{1}$$

where $g_i(x)$ denotes the output function of the ith hidden node.

For N arbitrary distinct samples (x_i, t_i), SLFNs with L hidden nodes are mathematically modeled as

$$\Sigma_{i=1}^{l} \beta_i g_i(x_j) = y_j, \quad j = 1, \dots, N. \tag{2}$$

That SLFNs can approximate these N samples with zero error means that

$$\Sigma_{j=1}^{l} \left\| y_j - t_j \right\| = 0. \tag{3}$$

The parameters in $g_i(x)$ can be trained according to (2).

This can be written as

$$G\beta = T, \tag{4}$$

where

$$G = \begin{bmatrix} G(x_1) \\ \vdots \\ G(x_N) \end{bmatrix} = \begin{bmatrix} g(w_1, b_1, x_1) & \cdots & g(w_l, b_l, x_1) \\ \vdots & \ddots & \vdots \\ g(w_1, b_1, x_N) & \cdots & g(w_l, b_l, x_N) \end{bmatrix}.$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_l^T \end{bmatrix}_{l \times m}, \qquad T = \begin{bmatrix} T_1^T \\ \vdots \\ T_l^T \end{bmatrix}_{N \times m}. \tag{5}$$

It is proved in [17] that given any small positive value $e > 0$, activation function $g : R \rightarrow R$ which is infinitely differentiable in any interval and N arbitrary distinct samples (x_i, t_i), there exists $l \leq N$ such that for any $\{w_i, b_i\}_{i=1}^l$ (parameters need to be trained in $g_i(x)$) randomly generated from any intervals of $R^d \times R$ according to any continuous probability distribution, with probability one, $\|H_{N \times l}\beta_{l \times m} - T_{N \times m}\| < e$. And from the interpolation point of view the maximum number of hidden nodes required is not larger than the number of training samples. In fact, if $L = N$, the training errors can be zero.

Though the ELM has good regression ability, there is still one obverse problem that the results of the ELM rely on the training data set. But the flaw difference is from not only length and width, but also depth, which may cause the MFL signal variance. It is impossible to include all flaw signal in the training data set. A better way is to use a small training data set to train the ELM which can generate good compensation results. One solution is given in this paper in Section 2.2.

2.2. PCA Based Flaw Exclusion.

To train the ELM and solve the problem mentioned in Section 2.1, one solution is given.

Because the property of pipe is learned using ELM, we can use the channel with no flaw to predict the channel with flaw when flaw is detected. By using the predicted result to substitute the channels which detect flaw, the compensation result can be obtained using ELM. And then, using the ELM result, the compensation is given to each channel. This avoids training ELM with every type of flaw. To exclude the flaw, a PCA based algorithm is stated as follows, which detects which channels and which sampling points detest flaw signal.

PCA [18] as an efficient statistical learning algorithm is useful to deal with multivariable problems. The PIG has n sensors surrounding the vessel. Consider one sampling of all sensors as $x = (x_1, x_2, \ldots, x_n)^T$. The linear transform of the sensors result can be written as

$$y_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = a_1^T x$$
$$y_2 = a_{11}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = a_2^T x$$
$$\vdots$$
$$y_n = a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = a_n^T x \tag{6}$$

y_1 is called the ith principle component. The computation steps is as follows.

First, m samples from each sensor are collected and written as a matrix $X \in R^{m \times n}$. The matrix is scaled to zero mean, and in addition to unit variance.

The second step is to compute the singular values. The covariance matrix is computed as

$$\Sigma \cong \frac{1}{m-1} X^T X. \tag{7}$$

An SVD (singular value decomposition) is used to compute the principal components and the associated singular vectors as

$$\frac{1}{m-1} X^T X = T \Lambda T^T$$
$$\Lambda = \begin{bmatrix} \Lambda_{\text{pc}} & 0 \\ 0 & \Lambda_{\text{res}} \end{bmatrix}, \tag{8}$$

where

$$\Lambda_{\text{pc}} = \text{diag}\left(\sigma_1^2, \ldots, \sigma_l^2\right) \in R^{l \times l}, \quad \sigma_1 \geq \cdots \sigma_l \geq \cdots \sigma_n$$

$$\Lambda_{\text{res}} = \text{diag}\left(\sigma_{l+1}^2, \ldots, \sigma_n^2\right) \in R^{(n-l) \times (n-l)}$$

$$T = \begin{bmatrix} T_{\text{pc}} & T_{\text{res}} \end{bmatrix} \in R^{(n \times n)}$$

$$T_{\text{pc}} \in R^{n \times l}, \qquad T_{\text{res}} \in R^{n \times (n-l)} \tag{9}$$

$$TT^T = T_{\text{pc}}T_{\text{pc}}^T + T_{\text{res}}T_{\text{res}}^T = I_{n \times n}$$

$$T_{\text{pc}}T_{\text{pc}}^T \begin{bmatrix} T_{\text{pc}}^T \\ T_{\text{res}}^T \end{bmatrix} \begin{bmatrix} T_{\text{pc}} & T_{\text{res}} \end{bmatrix}$$

l is the number satisfying

$$\frac{\Sigma_{i=1}^l \sigma_i^2}{\Sigma_{i=1}^n \sigma_i^2} \geq \alpha, \quad 0 \leq \alpha \leq 1 \tag{10}$$

$\sigma_i^2 / \Sigma_{i=1}^n \sigma_i^2 \geq \alpha$ is called the significance level. And (10) is called the cumulated significance level, which shows how much the first l principle components can reflect data X. The Hotelling T^2 statistic is used to detect fault

$$T_i^2 = \Sigma_{j=1}^l \frac{y_{ij}^2}{\lambda_j}. \tag{11}$$

With a set threshold, flaw signal can be separated from normal signal. Suppose there are only two sensors. Take 500 sampling. The result is shown in Figure 2. The linear transform of y is also shown in this figure. Using the Hotelling T^2 statistic, which is shown as formula (11), the flaw data can be detected.

2.3. Details of PCA and ELM Based Algorithm for Channel Equalization.

The data gathered using a PIG can be described as $X \in R^{m \times n}$, where m is the sampling number and n is the number of sensors. The algorithm proposed in this paper treats data with two points of view, one is from the sampling view and the other is from the sensor view.
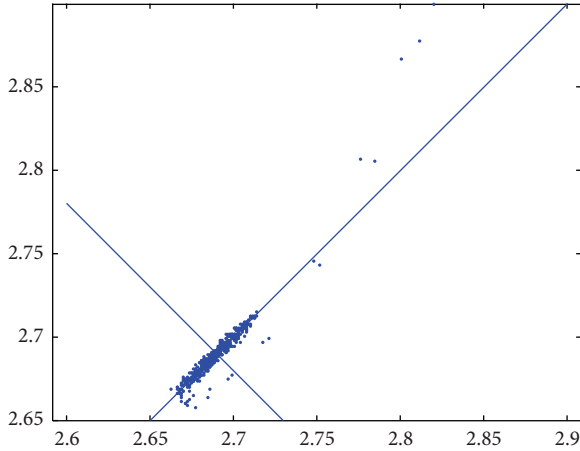
FIGURE 2: PCA analysis of 500 sampling using 2 sensors.

From the sampling view, the PCA algorithm is used to detect flaw, which determines at which sampling points the flaw locates. For example, take m sampling, using PCA, flaw locates at $[k, k + \Delta]$, where $1 \leq k \leq m$ and $1 \leq k + \Delta \leq m$ can be detected with a preset T^2 statistic. The normal part is tested using a trained ELM neutral network to remove channel mismatches, which is called ELM 1. And the flaw part is treated from the sensor point of view. The normal part is added to the training set dynamically in order to learn more character of this part of pipe.

From the sensor view, the flaw data is also treated using PCA to determine which sensors detect flaw. For example, only sensors s to $s + \Delta s$ detect flaw, where $1 \leq s \leq s + \Delta s \leq n$. The normal channel is used as input data, using ELM trained with channels of normal from training data set, the flaw part of signal can be forecasted as normal part. This step is to revert the flaw part to its normal condition, in order to determine how much each channel needs to be compensated with ELM 1. And using ELM 1, the channel mismatch of the flaw part can be removed. It is clear that each flaw needs an ELM neural network to compute the compensation, which is called the ELM p in Figure 3.

The flow chart of the algorithm is shown in Figure 3 with steps and details stated as follows.

Step 1. An ELM is trained using training data set to learn the character of the normal condition of the pipe for channel equalization. The target is each channels compensation. The ELM trained is marked as ELM 1.

Step 2. Using PCA with $x = (x_1, x_2, \ldots, x_n)^T$ represents sensors to detect flaw which is stated in Section 2.2 with a threshold. The T^2 statistic up over the threshold denotes the flaw signal. The signal will be separated into normal parts and flaw parts in Step 3.

Step 3. The signal of flaw detected from Step 2 is tested using PCA with $x = (x_1, x_2, \ldots, x_m)^T$ represents sampling points. A threshold is also set automatically. The T^2 statistic up over the threshold denotes the channels which detects flaw. It makes

the flaw signal detected from Step 2 separated into two parts, the signal of normal channels and the signal channel of flaw.

Step 4. Signal of normal parts acquired from Step 2 is tested using ELM 1 trained in Step 1.

Step 5. For the flaw signal detected in Steps 2 and 3, another ELM is trained with training data set. The normal channels are used as inputs of the ELM, and the flaw channels are used as output of the ELM. For example, the flaw is detected at sampling point from t to $t + \Delta t$, and channels from s to $s + \Delta s$. The training data set from channel 1 to $s - 1$ and $s + \Delta s + 1$ to n is used to train this neural network. The target of the ELM is set as the training data set with channels from s to $s + \Delta s$. Each flaw has an ELM. The ELM is marked as ELMp with p representing the number of flaw.

Step 6. The flaw signal from t to $t + \Delta t$, and channels from s to $s + \Delta s$ is replaced temporarily by the test result with ELMp.

Step 7. The signal segment in Step 6 is tested using ELM 1. The output is compensated to the original signal from t to $t + \Delta t$ and channels from s to $s + \Delta s$.

Step 8. Update the training data set with normal data acquired in Step 2.

3. Experiment Results

The PIG used to collect data in this paper consists of 15 carriers with 5 axial sensors on each carrier. An 8-inch seamless steel pipe with length of about 14 meters is used to do this experiment. 9 exterior flaws were made on this pipe. The pipeline in our experiment is connected with two flanges and one weld. The sampling is controlled by an odometer wheel with the sampling frequency of 1 sampling per 2 mm. Several experiments were done with different load angel of the PIG.

In order to train our algorithm, some signal of MFL of normal condition pipeline with no flaw is needed. The training data is obtained in two way. (i) The first way is to obtain from the original training data. One test data is selected as original training data with flaw signal excluded manually. (ii) The second way of getting the training data is generated from each test using algorithms stated in Sections 2.1 and 2.2. The algorithm used in this paper treats data batch by batch. One batch of data treated using the algorithm can separate this batch of data into normal data and flaw data. The normal part of data is added to the training data set in order to reinforce the training result. By adding new training data, the training set is updated. New character of the normal condition is studied. This means as the process goes, the result of the algorithm proposed in this paper gets better. To avoid the training data set getting too large, the length of the data set is set to a certain length. When length of the training data set grows up to its limits, the earlier training data will be erased and new training data will be added to replace the vacancy.

And in order to show the efficiency of our algorithm, the length of the original training data is reduced to only less than 1/5 of one test, though theoretically a bigger training
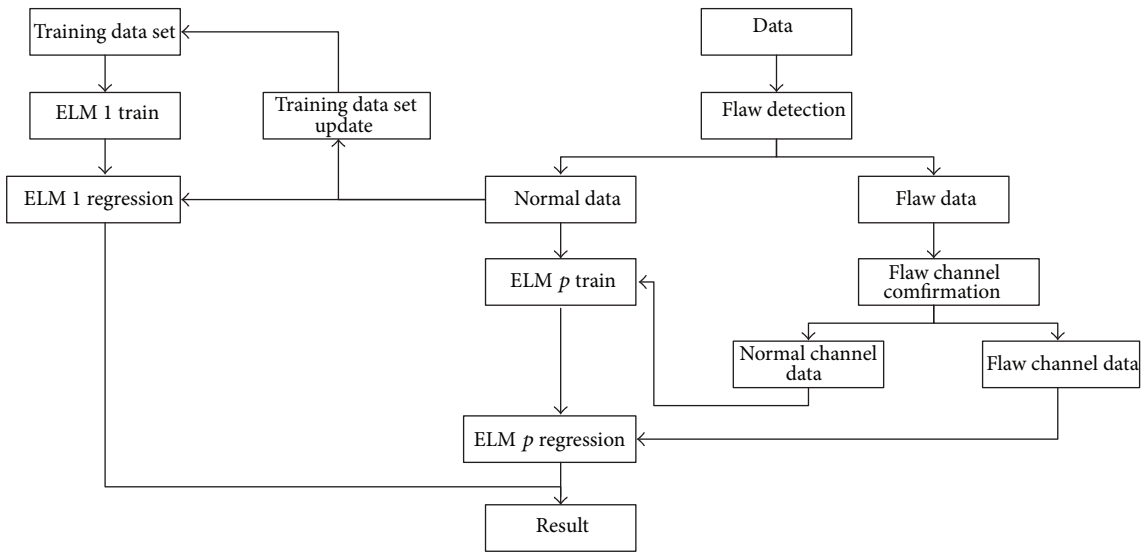
Figure 3: Flow chart of algorithm proposed.

data set will get better result. Also restricted to the experiment environment, it is not possible to build a pipeline with enough length of hundreds of meters or even miles. The results with less original training data also show that the algorithm proposed in this paper generates satisfying results.

And algorithm of median corrector is also adopted to compare with our results. Among many factors that influence channel-to-channel mismatch, the lift-off value and the difference of the baseline play the main role. The influence caused by difference of the lift-off value among sensors can be reduced by improving the assembly skills. And the influence of the baseline (zero output) of sensors can be reduced by calculating the average of sensors as baseline. The channel equalization can be computed as three steps. Assume X as data gathered by sensing a clean pipe (almost no flaw). First, calculate each channels average $\overline{x_i}$. Second, calculate x_{mean} the average of $\overline{x_i}$. Third, compensate $x_{\mathrm{mean}} - \overline{x_i}$ to each channel. But as the MFL data is processed automatically, this method needs to be operated manually. And it is not that easy to find such steady state signal. To overcome these, we use median corrector to do rough channel equalization in engineering. Instead of calculating the average of each channel in the first step, $\overline{x_i}$ is each channel's median value. Other steps are same as the average method.

All data of one test is treated using the algorithm proposed in this paper. Figure 4 shows the PCA result of Step 2 in Section 2.3. The threshold is automatically generated with F distribution parameter of 95%. All the components and flaws are described in Figure 4. Details of raw signal of 3 flaws are shown to illustrate the following steps in Figure 5. The dashed line shows the flaw sampling point intervals detected using PCA as shown in Figure 5. By applying the sensor view of PCA stated as Step 3 in Section 2.3, the flaw signal channels are marked within the solid line in Figure 5. The sensor view PCA result is shown in Figure 6, with F distribution
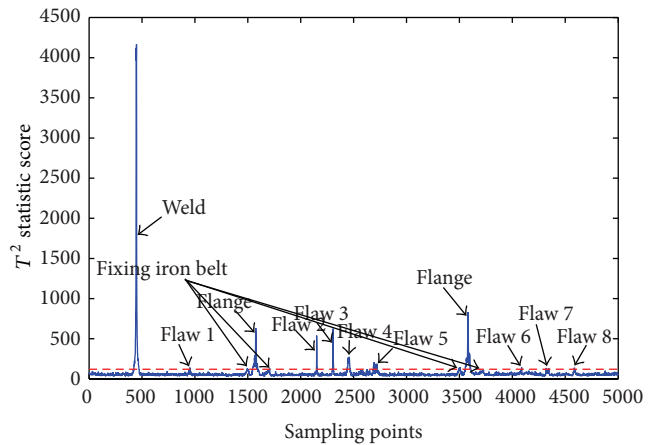

Figure 4: PCA flaw exclusion result of sampling view.
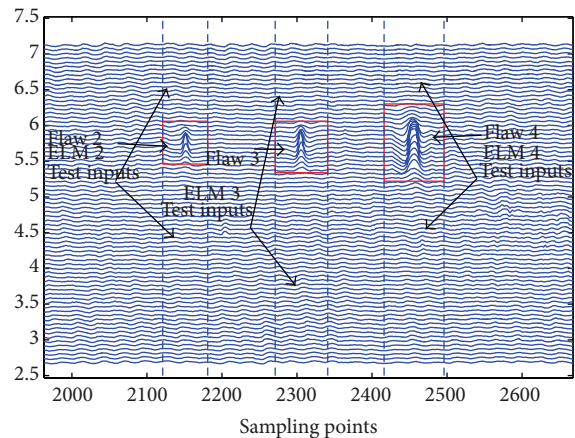

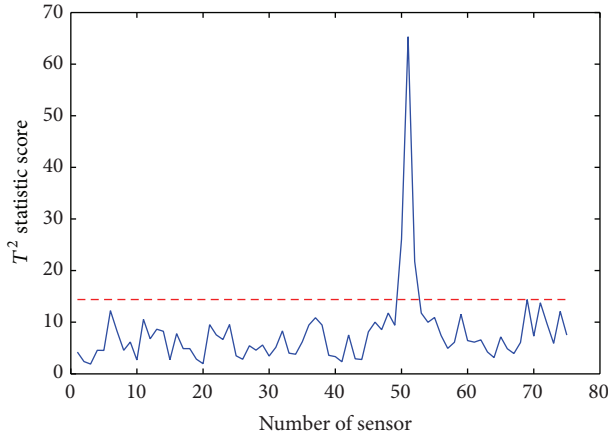Figure 5: Detail of flaw signal.

Figure 6: PCA flaw exclusion result of sensor view.
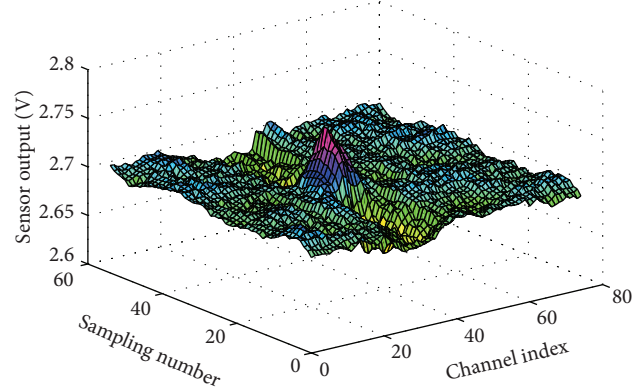

Figure 9: PCA and ELM based algorithm treated data of flaw 1.
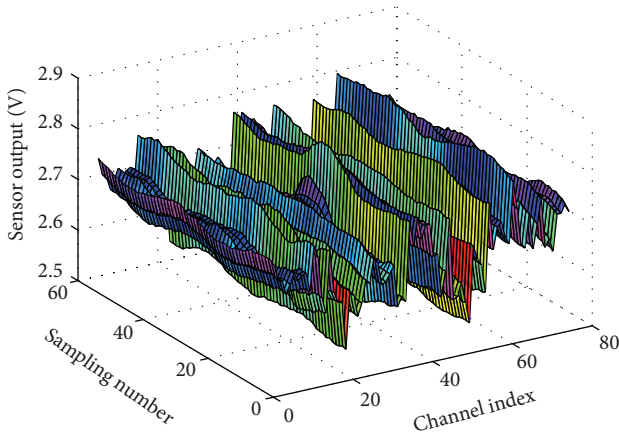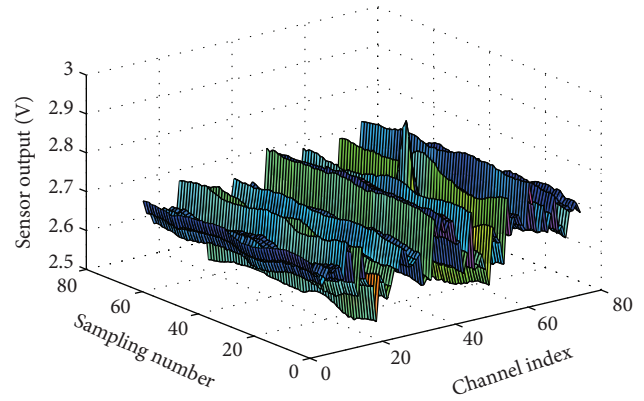

Figure 7: Raw data of flaw 1.
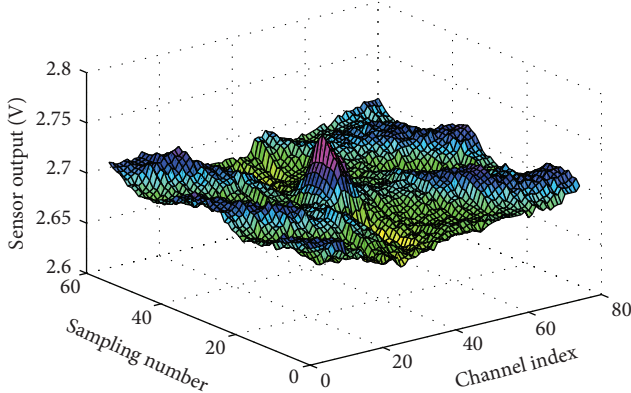

Figure 10: Raw data of flaw 2.

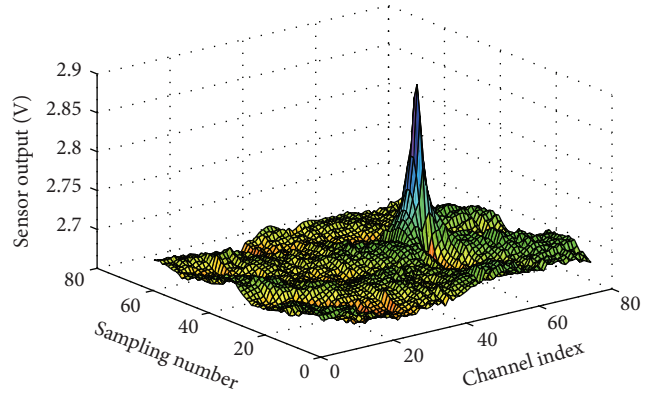Figure 8: Median treated data of flaw 1.

Figure 11: Median treated data of flaw 2.

parameter of 95%. The normal channels is used as input of ELM p, and test result is shown from Figures 7–12.

To show the result, the standard deviation (SD) and peak signal to noise ratio (PSNR) are adopted. The results are shown in Tables 1 and 2. Lower standard deviation reflects that the signal is cleaner with less channel mismatch. To illustrate the result, two segments of flaw signal are plotted.

The baseline estimation algorithm proposed in [13] is also compared in Tables 1 and 2. From Table 1, it can be seen that the flaw in show is typical small size flaw. The PSNR results in Table 2 also indicate that our algorithm minimizes channel-to-channel mismatches.

From Figures 7 and 10, it is clear that the signal has great channel mismatches which makes it impossible to evaluate the flaw size. Using median algorithm corrected data shown as Figures 8 and 11, the flaw signal is prominent and the signal of normal channel is smooth, but still some ripple exists.

TABLE 1: SD of raw data and corrected data.

Index of flaw	SD of raw data	SD of median corrected data	SD of baseline estimation equalized data	SD of adaptive channel equalized data	flaw size (length * width * depth, unit mm)
1	0.0544	0.0129	0.0121	0.0093	$40 \times 20 \times 1$
2	0.0548	0.0138	0.0136	0.0108	$\phi 6$ perforation
3	0.0565	0.0175	0.0169	0.0158	$\phi 8$ perforation
4	0.0617	0.0324	0.0318	0.0288	$20 \times 40 \times 4$
5	0.0564	0.0182	0.0179	0.0160	$40 \times 20 \times 4$
6	0.0560	0.0143	0.0143	0.0142	$20 \times 20 \times 2$
7	0.0544	0.0115	0.0104	0.0089	$40 \times 20 \times 2$
8	0.0573	0.0174	0.0163	0.0141	$20 \times 40 \times 2$

TABLE 2: PSNR of raw data and corrected data.

| Index of flaw	PSNR of raw data	PSNR of median corrected data	PSNR of baseline estimation equalized data	PSNR of adaptive channel equalized data	flaw size (length * width * depth, unit mm)
1	16.8393	25.6999	26.0056	26.7182	$40 \times 20 \times 1$
2	18.6216	30.4281	31.0981	32.5947	$\phi 6$ perforation
3	19.0502	29.0965	29.2674	30.0454	$\phi 8$ perforation
4	18.5951	24.0668	24.1328	24.6181	$20 \times 40 \times 4$
5	17.5086	25.5968	25.8254	26.5034	$40 \times 20 \times 4$
6	16.8988	21.7448	22.9136	24.0683	$20 \times 20 \times 2$
7	16.4687	26.5349	26.8903	28.1939	$40 \times 20 \times 2$
8	17.5993	26.5228	26.8162	27.8077	$20 \times 40 \times 2$
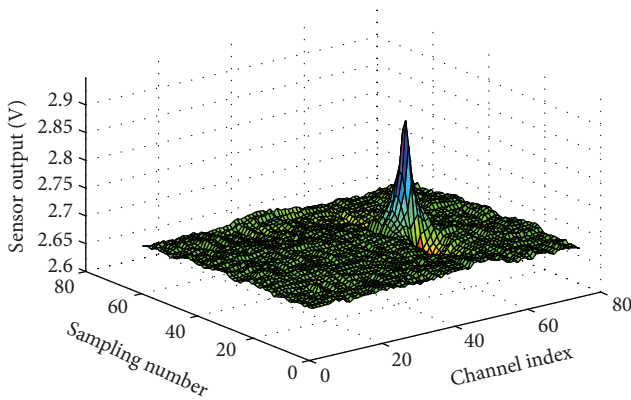
FIGURE 12: PCA and ELM based algorithm treated data of flaw 2.

Figures 9 and 12 show the results of algorithm proposed in this paper, the flaw signal is more prominent and the normal part is smoother, which results easy to evaluate the flaw. The SD shown in Table 1 also indicates that data treated by our algorithm has less channel mismatches and noise.

4. Conclusion

In this paper, a new adaptive channel equalization is proposed for processing of MFL signal prior to flaw characterization. The scheme performs channel equalization by using single layer neural networks, and the fast learning algorithm of ELM is used to give excellent processing speed. Focusing on the signal of flaw, a PCA based flaw detect algorithm is given to locate the flaw signal. The algorithm proposed in this paper minimizes the channel-to-channel mismatch and reduces the distortion of the signal of the flaw. Both theory analysis and simulation results show the efficiency of our algorithm. For the flaw signal, the algorithm proposed in this paper needs to locate the flaw first. For shallow and small flaw, how to locate it and make less false detection is still a problem in both theory and engineering.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] A. A. Carvalho, J. M. A. Rebello, L. V. S. Sagrilo, C. S. Camerini, and I. V. J. Miranda, "MFL signals and artificial neural networks applied to detection and classification of pipe weld defects," *NDT & E International*, vol. 39, no. 8, pp. 661–667, 2006.

[2] R. Christen and A. Bergamini, "Automatic flaw detection in NDE signals using a panel of neural networks," *NDT and E International*, vol. 39, no. 7, pp. 547–553, 2006.

[3] Y. Xiang and S. K. Tso, "Detection and classification of flaws in concrete structure using bispectra and neural networks," *NDT and E International*, vol. 35, no. 1, pp. 19–27, 2002.

[4] S. Mukhopadhyay and G. P. Srivastava, "Characterization of metal loss defects from magnetic flux leakage signals with discrete wavelet transform," *NDT and E International*, vol. 33, no. 1, pp. 57–65, 2000.

[5] D. Mukherjee, S. Saha, and S. Mukhopadhyay, "Inverse mapping of magnetic flux leakage signal for defect characterization," *NDT & E International*, vol. 54, pp. 198–208, 2013.

[6] S. Kathirmani, A. K. Tangirala, S. Saha, and S. Mukhopadhyay, "Online data compression of MFL signals for pipeline inspection," *NDT and E International*, vol. 50, pp. 1–9, 2012.

[7] Q. He, R. Yan, F. Kong, and R. Du, "Machine condition monitoring using principal component representations," *Mechanical Systems and Signal Processing*, vol. 23, no. 2, pp. 446–466, 2009.

[8] R. Yan, R. X. Gao, and X. Chen, "Wavelets for fault diagnosis of rotary machines: a review with applications," *Signal Processing*, vol. 96, pp. 1–15, 2014.

[9] B. Li and X. Chen, "Wavelet-based numerical analysis: a review and classification," *Finite Elements in Analysis and Design*, vol. 81, pp. 14–31, 2014.

[10] Z. K. Zhu, R. Yan, L. Luo, Z. H. Feng, and F. R. Kong, "Detection of signal transients based on wavelet and statistics for machine fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 23, no. 4, pp. 1076–1097, 2009.

[11] R. Yan and R. X. Gao, "An efficient approach to machine health diagnosis based on harmonic wavelet packet transform," *Robotics and Computer-Integrated Manufacturing*, vol. 21, no. 4-5, pp. 291–301, 2005.

[12] Y. Zhang, Z. Ye, and X. Xu, "An adaptive method for channel equalization in MFL inspection," *NDT & E International*, vol. 40, no. 2, pp. 127–139, 2007.

[13] D. Mukherjee, S. Saha, and S. Mukhopadhyay, "An adaptive channel equalization algorithm for MFL signal," *NDT and E International*, vol. 45, no. 1, pp. 111–119, 2012.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[15] D. Lowe, "Adaptive radial basis function nonlinearities, and the problem of generalisation," in *Proceeding of the 1st IEE International Conference on Artificial Neural Networks*, pp. 171–175, London, UK, October 1989.

[16] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[17] G. Huang, L. Chen, and C. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Transactions on Neural Networks*, vol. 17, no. 4, pp. 879–892, 2006.

[18] G. Li, C. F. Alcala, S. J. Qin, and D. Zhou, "Generalized reconstruction-based contributions for output-relevant fault diagnosis with application to the Tennessee Eastman process," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 5, pp. 1114–1127, 2011.