ELSEVIER

NDT&E International 35 (2002) 519–528

NDT&E international

www.elsevier.com/locate/ndteint

Automatic identification of different types of welding defects in radiographic images

Gang Wang, T. Warren Liao*

Department of Industrial and Manufacturing Systems Engineering, Louisiana State University, Baton Rouge, LA 70803, USA

Received 14 August 2001; revised 5 May 2002; accepted 22 May 2002

Abstract

Radiographic testing is a well-established non-destructive testing method to detect subsurface welding defects. In this paper, an automatic computer-aided identification system was implemented to recognize different types of welding defects in radiographic images. Image-processing techniques such as background subtraction and histogram thresholding were implemented to separate defects from the background. Twelve numeric features were extracted to represent each defect instance. The extracted feature values are subsequently used to classify welding flaws into different types by using two well-known classifiers: fuzzy *k*-nearest neighbor and multi-layer perceptron neural networks classifiers. Their performances are tested and compared using the bootstrap method.

Keywords: Radiographic images; Weld inspection; Image processing; Fuzzy *k*-nearest neighbors; Neural networks

1. Introduction

Inspection of welded structures is essential to ensure that the quality of welds meets the requirements of the design and operation, thus to assure safety and reliability. A variety of NDT methods are available for the inspection of welding defects [5]. Visual inspection is the primary evaluation method of many quality control programmes. It can be easily carried out, inexpensive, and usually does not require special equipment other than magnifying glasses, boroscopes, or television camera systems. It is used most effectively for the inspection of welds where quick detection and correction of flaws or process related problems can result in significant cost savings. For more critical welded structures such as high-pressure vessels, the nature, location, and magnitude of the flaws must be mapped in order to determine their acceptability by further mechanics analyses. To this end, more sophisticated NDT methods such as ultrasonic testing (UT) and radiographic testing (RT) are needed. Both UT and RT methods have their limitations, e.g. defects lying in certain plains can be picked up on one and not the other and vice versa. The two techniques complement one another and should be used together wherever possible and where price allows.

Ultrasonic inspection uses sound waves of short wavelength and high frequency to detect flaws. Usually pulsed beams of high frequency ultrasound are used via a hand-held transducer which is placed on the specimen. Any sound from that pulse that returns to the transducer like an echo is shown on a screen which gives the amplitude of the pulse and the time taken to return to the transducer. Defects anywhere through the specimen thickness reflect the sound, back to the transducer. Flaw size, distance and reflectivity can be interpreted. Automated UT systems are now commercially available (http://www.globalxray.com/auto_ut.html).

RT is the other commonly used NDT method for detecting internal welding flaws. It is based on the ability of X-rays or γ-rays to pass through metal and other materials opaque to ordinary light, and produce photographic records by the transmitted radiant energy [10]. Because different materials absorb either X-ray or γ-rays to different extent, penetrated rays show variations in intensity on the receiving films. That provides a means to examine the internal structure of a weld.

Traditionally, experienced workers are required to evaluate the weld quality based on radiographic images. Therefore, the results very much depend upon the capability and experience of the operator. Unfortunately, the manual interpretation process is time-consuming and the results could be very subjective, inconsistent, and sometimes

* Corresponding author. Tel.: +1-225-578-5365; fax: +1-225-578-5109.
E-mail address: ieliao@lsu.edu (T.W. Liao).

0963-8695/02/$ - see front matter
PII: S0963-8695(02)00025-7

biased. Therefore, it is desirable to develop a computer system to increase the objectivity, consistency, accuracy, and efficiency of RT inspection. To the best of our knowledge, no commercially viable automated RT system exists today. Nevertheless, several efforts have been devoted to the development of such a system. In our view, an automated RT system should have three major functions: segmenting welds from the background, identifying the flawed segments in the weld, and classifying different types of welding flaws. These developments rely mostly on techniques such as image processing, feature extraction, and pattern recognition. Past studies in each one of the above-mentioned areas are briefly reviewed in the following.

Liao and Ni [16] proposed a methodology for the extraction of welds from digitized radiographic images. The method was based on the observation that the intensities of pixels in the weld area distribute more as a Gaussian distribution than other areas in the image. This method had been proved effective, however, only to segment linear welds. Subsequently, Liao and Tang [17] applied a multilayer perceptron (MLP) neural network based procedure for the same application. This method was successfully applied to segment both linear and curved welds. Another study [14] employed fuzzy classifiers, specifically fuzzy *K*-NN and fuzzy *c*-means, instead of neural networks as the pattern classifier. This method can also be applied to segment curved welds, and can handle varying number of welds in one radiographic image.

After welds are successfully extracted, of interest is to identify the defect areas in welds. Daum et al. [3] proposed a defect segmentation algorithm based on a background subtraction algorithm. This algorithm was proved effective regardless the defect types. But it had difficulties in detecting small defect regions (4–6 pixels). Gayer et al. [6] described a two-step process for the automatic recognition of welding defects through radiography. A fast search for defect regions was followed by identifying and locating defects. The fast search could be accomplished either by comparing defects with known defects' templates, or by thresholding the image. This method tried to imitate the way a human inspector inspects radiographs: first, a general glance with coarse resolution, followed by fine focusing on defective regions. Hyatt et al. [9] presented a multiscale method for segmenting flaw indications from the background radiographic images. The method was designed to remove the overall background structure while reserving the defect details. Another welding flaw detection method was presented by Liao and Li [15]. This method was based on the observation that welding flaws usually result in distortions in the overall line profile of a weld. The whole process consisted of four parts: preprocessing, curve fitting, profile-anomaly detection, and postprocessing. Test results indicate that their method has high successful detection rate and an acceptable false alarm rate. Liao et al. [13] proposed another approach using fuzzy clustering methods. Twenty-five features were selected for each line of the radiographic image. The results showed that fuzzy *K*-NN outperformed fuzzy *c*-means. Murakami [18] offered a simple algorithm for defects detection, which was achieved by conducting local arithmetical operation to a limited region and by thresholding.

To date, there is not much research done on automatically identifying the type of a welding defect. Murakami [18] classified defect types with an expert system. The features used in the expert system include the shape, position and intensity level of the defect pattern. However, results from this method strongly depended on the types of defects. Using their system detecting blowholes is relatively easy; but detecting cracks is difficult. Kato et al. [11] proposed another expert system for identifying different types of welding defects. The identification rules were based on the interviews with expert inspectors. Six features were extracted from each welding defect, which cover information about shape, intensity and location. The performance of this method was tested and found even better than the judgments from human expert inspectors. Aoki and Suga [1] used a three-layer artificial neural network to identify defect types. Ten discrimination features were automatically generated from each defect by image processing techniques. The whole algorithm achieved a successful rate of 92.6%.

This paper presents a system developed for classifying welding defect types as a continuation of our research in this area.

2. Methodology

Fig. 1 shows the major steps of our welding defect classification system. This system was realized using three major techniques: digital image processing, feature extraction, and pattern classification. Digital image processing techniques are used to extract the principal objects, which are welding defects in this research, from radiographic images. Usually, defects in the original X-ray image are low in number comparing with its background information, and mixed with noises coming from various processes in the formation of X-ray images. Digital image processing techniques are employed to lessen the noise effects and to improve the contrast, so that the principal objects in the image can be more apparent than the background. Feature extraction is necessary to obtain a set of features that can describe the characteristics of welding defects. These features should be small in number and high in discriminatory power. Pattern classification methods are needed to analyze feature data and make a prediction of the defect type. Pattern classification algorithms might differ in efficiency and accuracy. Therefore, two renowned supervised algorithms: fuzzy *k*-nearest neighbor (*K*-NN) and MLP neural networks are investigated.

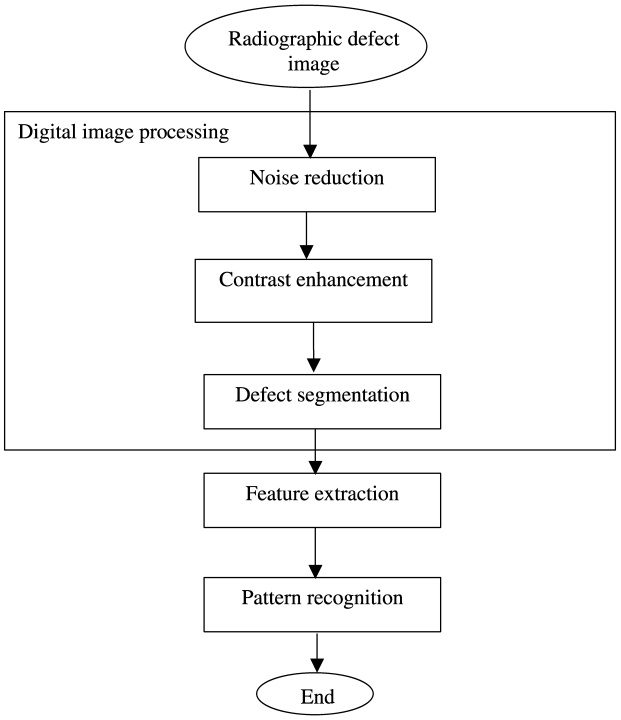


Fig. 1. Procedure for the automatic classification of welding defect types.

2.1. Noise reduction

The gray level values of the noise pixels are much higher than those of their immediate neighbors. As noise is characterized as high frequency values, low-pass filtering methods, like a median filter, can be used to effectively remove the noise pixels. The median filter deploys a small mask template, which is usually a 3×3 or 5×5 structuring element. The template operation may be calculated by either correlation or convolution operators. The median filter replaces a pixel's gray level with the median value of its neighborhood. This operation improves on other filtering methods by preserving edges while suppressing noises. The algorithm is presented below

$$g'(x,y) = \text{median}\{g(x_1,y_1)|(x_1,y_1) \text{ is in } N(x,y)\} \quad (1)$$

where $N(x,y)$ is the immediate neighbors of the pixel (x,y).

2.2. Contrast enhancement

Radiographic images usually have poor contrast and a lack of detail. The aim of the contrast enhancement is to improve the quality of radiographic images by highlighting the useful information while leaving the unimportant information intact. One reason for poor contrast in radiographic images is the limited range of intensities the image capture device is able to accommodate [8]. In the original radiographic image, the distribution of gray levels is highly skewed towards the darker side. Defects in these areas can hardly be recognized. Therefore, it is desirable to stretch the histogram distribution to an evenly distributed one, like a rectangular shape instead of a skewed shape. The histogram equalization algorithm proportionally modifies the original gray level values to the range of 0 and 255, as shown in Eq. (2) below:

$$g'_n = g_{\min} + (g_{\max} - g_{\min})C_n \quad (2)$$

In the above equation g'_n is the modified gray level value for a pixel, and C_n is calculated by

$$C_n = \sum_{k=0}^{i} g_k / \sum_{k=0}^{j} g_k \quad (3)$$

where i is the gray level value of the current pixel, j is the maximum gray level value, which is 255 in this case, in the available range. This algorithm spreads the gray levels to a wide range, thus showing details in areas with a high brightness gradient.

2.3. Defect segmentation

2.3.1. Background subtraction method

Background subtraction method (BSM) is one of the commonly used algorithms for image segmentation. The background is defined to be the regions in an image that are not significant to the analyst [2]. Because defects are superimposed on other image structures like the shape of the welding bead; defects in an image can be acquired by subtracting the background from the original image. Daum et al. [3] noticed that the defect indications in an image could be characterized by high spatial frequencies; while the normal welding bead with its reinforcement causes only a gradual gray level change (low spatial frequencies). Therefore, a gradual gray level change was simulated by a two-dimensional (2D) background model $f_B(x,y)$ and subtracted from the original image $f(x,y)$. This method can be applied to all types of defects and is followed in this paper. Fig. 2 shows how the background subtraction method ideally works. The background model $f_B(x,y)$ was estimated by smoothing the brightness distribution, as shown in Fig. 2b. Fig. 2c gives the result of the background subtraction, in which only the defect area is left in the image. Note that Fig. 2b was manually constructed; an actual model would not be nearly as ideal.

The background model $f_B(x,y)$ was estimated by the surface-fitting algorithms implemented by an analytical software—TableCurve3D from SPSS. Each weld image is imported in the format of 3D data, which indicate pixel position across the weld bead, along the weld bead, and pixel gray level. Given a weld image (Fig. 3a) with the corresponding 3D data display shown in Fig. 3b, TableCurve3D tests the selected functions in its library and ranks them based on the least-squared error. Fig. 3c shows the background model constructed based on a polynomial function, $Z = a + bx + cx^2$, with a r^2 value of 0.86. Fig. 3d is the 3D display of Fig. 3c. This simple function with constant y was chosen to ensure that defect is always

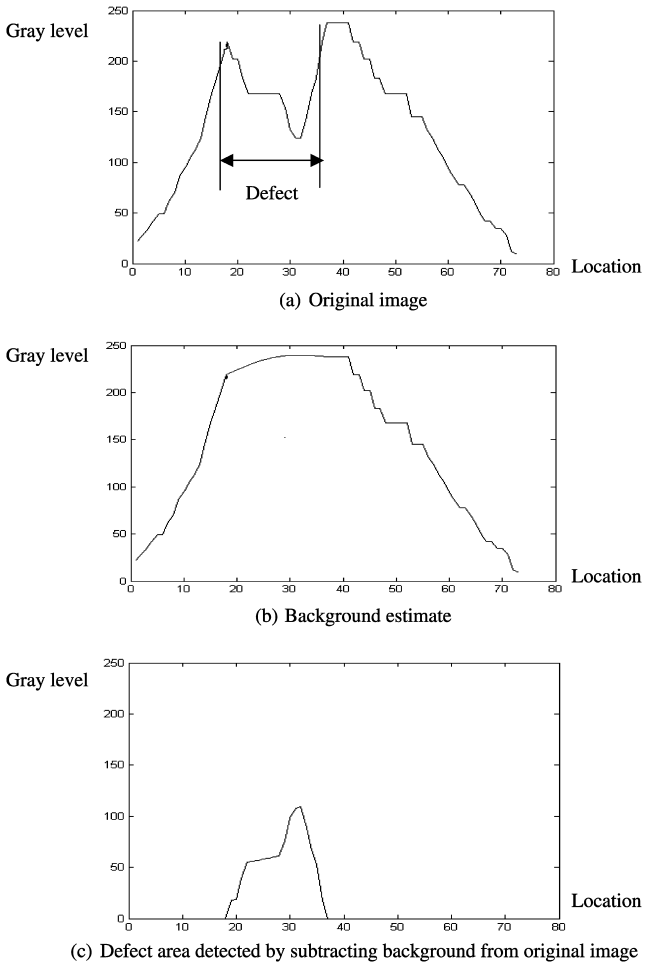


Fig. 2. Background subtraction method. (a) Original image; (b) background estimate; (c) defect area detected by subtracting background from original image.

preserved as much as possible. Higher noises are thus unavoidable as evident in the subtracted image (Fig. 4a). This leads to the discussion of the next operation.

2.3.2. Histogram thresholding method

After the background is subtracted from the image, defects in the resultant images still need to be separated from noises by a thresholding operation. Aoki and Suga [1] implemented a histogram thresholding method to discriminate between defects and the background. Since the frequency of background and noises is much higher than that of defects, a sharp gradient exists between the noise class and the defect class. In order to classify the two classes, a tangential fitting line is calculated by the least square method (LSM). With LSM, we can fit a straight line, $y = a + bx$, for a given set of data points. The sum of squares of the distances from those data points to this straight line is a minimum. The parameters, a and b, can be determined by the following equations:

$$a = (N\sum xy - \sum x\sum y)/(N\sum x^2 - (\sum x)^2) \tag{4}$$

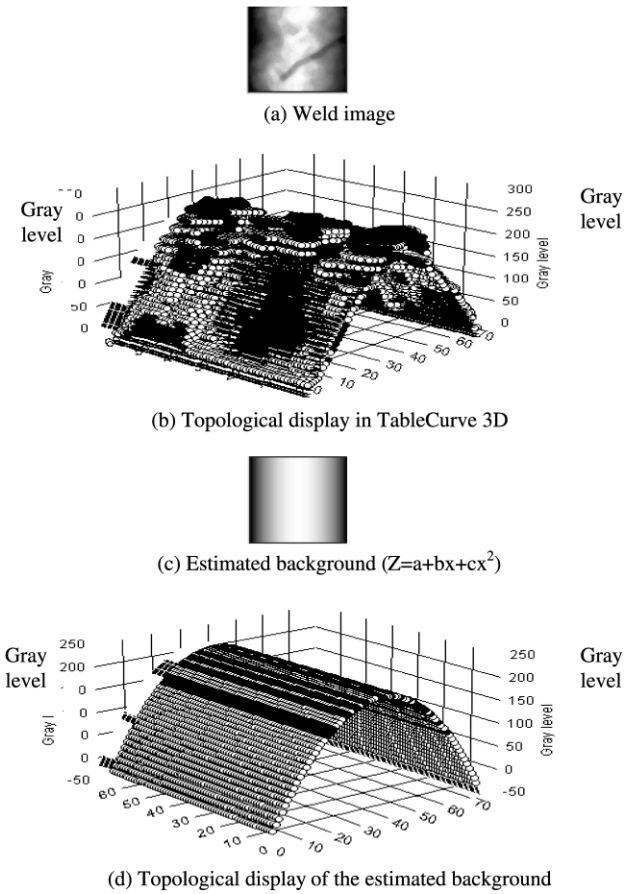


Fig. 3. Fitting background with TableCurve3D. (a) Weld image; (b) topological display in TableCurve3D; (c) estimated background ($Z = a + bx + cx^2$); (d) topological display of the estimated background.

$$b = (\sum y\sum x^2 - \sum x\sum xy)/(N\sum x^2 - (\sum x)^2) \tag{5}$$

The threshold is determined as the crossing point of the fitted line and the horizontal axis. Those pixels with gray levels that are higher than the threshold value are considered as noises and eliminated. An example is shown in Fig. 4.

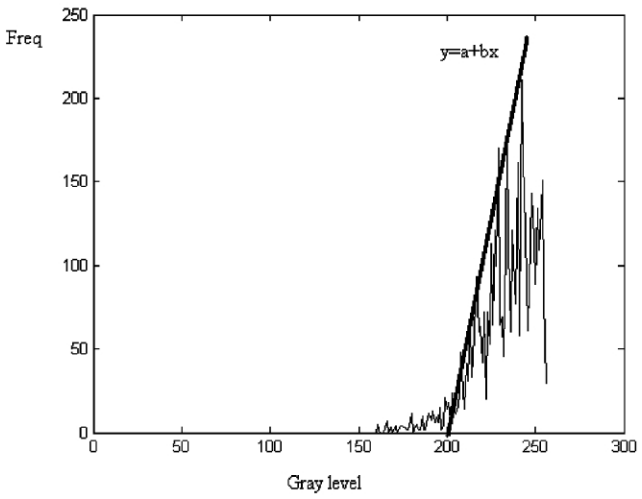
2.4. Feature extraction

In this study, features describing the shape, size, location, and intensity information of welding defects were extracted.

1. *Distance from center*. More specifically, the distance between the center of defect and the central-line of the welding-bead (Fig. 5).
2. *Radius mean, standard deviation, and circularity*. This set of features measures the circularity of defects. Circularity is defined as the standard deviation, σ_r, divided by the mean radius, μ_r

$$\mu_r = \frac{\sum_{(x,y)\text{ in }A_b}[(x-\bar{x})^2 + (y-\bar{y})^2]^{1/2}}{\#(A_b)} \tag{6}$$

(a) Result of BSM



(b) Histogram of defect image after BSM

(c) Binary image after threshold

Fig. 4. Result of histogram thresholding. (a) Result of BSM; (b) histogram of defect image after BSM; (c) binary image after threshold.

$$\sigma_r^2 = \frac{\sum([(x-\bar{x})^2 + (y-\bar{y})^2]^{1/2} - \mu_r)^2}{\#(A_b)} \tag{7}$$

where A_b is the set of boundary points of defect, $\#(A_b)$ is the number of boundary points, x and y are coordinates of boundary pixels around the defect, $\bar{x}$ and $\bar{y}$ are coordinates of the center of mass of defect object. $\bar{x}$ and $\bar{y}$ are calculated as follows

$$\bar{x} = \frac{\int\int xK(x,y)\mathrm{d}x\,\mathrm{d}y}{\text{area}} \tag{8}$$

$$\bar{y} = \frac{\int\int yK(x,y)\mathrm{d}x\,\mathrm{d}y}{\text{area}} \tag{9}$$

$$\text{area} = \int\int K(x,y)\mathrm{d}x\,\mathrm{d}y \tag{10}$$

where $K(x,y)$ is 1 for the pixel (x,y) that is in the defect, and 0 elsewhere.

3. Compactness: This feature measures the object shape

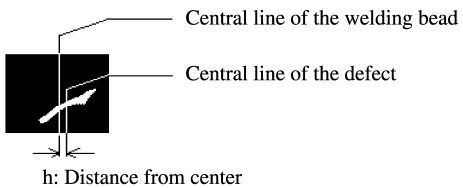
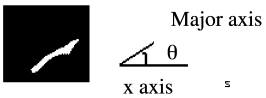


Fig. 5. Distance from center.



θ: The angle between major axis and horizontal line

Fig. 6. Major axis of the defect object (θ is the angle between the major axis and horizontal line).

that is calculated by

$$\text{compactness} = \frac{\text{perimeter}^2}{\text{area}} \tag{11}$$

where perimeter is the number of boundary points around the defect area. A circular object has a smaller compactness value than a non-circular object.

4. *Major axis*. The orientation of the defect object stretching in the weld. It is calculated as the angle between the orientation of defect and the horizontal line (Fig. 6).
5. *Width and length*. Length is the projection of defect along the major axis whereas width is the projection along the minor axis, which is perpendicular to the major axis (Fig. 7).
6. *Elongation*. Eq. (12) computes the ratio between the length and width of the defect object, which takes a value between 0 and 1

$$\text{elongation} = \frac{\text{width}}{\text{length}} \tag{12}$$

7. *Heywood diameter*. The diameter of a circle having an equivalent area to that of the defect (Fig. 8).
8. *Average intensity and standard deviation of intensity*. It gives the information about the brightness of the defect.

2.5. Pattern classification

2.5.1. Fuzzy k-nearest neighbor

The fuzzy k-nearest neighbor (K-NN) algorithm is a supervised classifier. It requires the use of training data, which has to be comprised of representative examples from different classes.

The fuzzy k-nearest neighbor (K-NN) algorithm [12] has the following steps:

1. Initialize k, where $1 \leq k \leq n$.
2. Find the k-nearest neighbor for each x by comparing the distance between x and each training instance. Label the k--nearest neighbor as $x_1, \ldots, x_j, \ldots, x_k$.
3. Assign the membership value of x belonging to class i

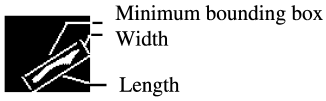


Fig. 7. Width and length of the defect object.

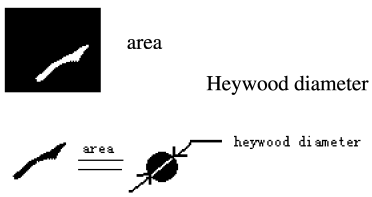


Fig. 8. Heywood diameter.

using the following equation

$$u_i(x) = \frac{\sum_{j=1}^{k} u_{ij}(1/\|x - x_j\|^{2/(m-1)})}{\sum_{j=1}^{k} (1/\|x - x_j\|^{2/(m-1)})} \tag{13}$$

Repeat steps 2 and 3 for all test data.

The variable m in Eq. (13) determines how heavily the distance is weighted when calculating each neighbor's contribution to the membership value. As m increases, the neighbors are more evenly weighted, and their relative distances from the point being classified have less effect [13]. The commonly used m value is 2. The symbol '$\|\cdot\|$' denotes Euclidean distance in multi-norm, which is defined as

$$\|x - x_j\| = \sqrt{\sum_{q=1}^{Q} (x_q - x_{qj})^2} \tag{14}$$

where x_q is the qth feature of vector x.

The membership value u_{ij}, which is the membership of x_j in class i, is computed by the following equation:

$$u_{ij} = \begin{cases} 1, & \text{if } i \text{ is the class that } x_j \text{ belongs to} \\ 0, & \text{if } i \text{ is not the class that } x_j \text{ belongs to} \end{cases} \tag{15}$$

In short, Fuzzy K-NN first uses a training dataset to get membership values of how training vectors belong to each class. After training, fuzzy K-NN algorithm can determine the class of a test instance based on its distance from nearest neighbors and each neighbor's membership in each class. Generally, the test instance is assigned the class that is represented by the majority of its k-nearest neighbors in the training dataset.

2.5.2. Multi-layer perceptron neural network

The MLP neural network is one of the best-known supervised feedforward networks. Fig. 9a shows the structure of an MLP with three layers: an input layer, and output layer, and a hidden layer between input and output. Neurons in the hidden layer, as shown in Fig. 9b, sum up values from input nodes after weighting them with appropriate weights W_{ji} and compute the output Y_j as a function f of the summation.

The backpropagation (BP) algorithm [19], a gradient descent algorithm, is the most commonly adopted MLP training algorithm. It computes the change ΔW_{ji} on the weight of a connection between an input neuron X_i and a neuron j as follows:

$$\Delta W_{ij} = \eta \delta_j X_i \tag{16}$$

where η is the learning rate and δ_j is a factor determined by the type of neuron j. If j is a neuron in the hidden layer

$$\delta_j = \left(\frac{\partial f}{\partial \text{net}_j}\right) \sum_q W_{qj}\delta_q \tag{17}$$

If j is a neuron in the output layer

$$\delta_j = \left(\frac{\partial f}{\partial \text{net}_j}\right)(Y_j^{(t)} - Y_j) \tag{18}$$

where net_j is the weighted sum of input nodes connected with neuron j; $Y_j^{(t)}$ is the target output for neuron j; and δ_q is obtained from the neuron q connected to the output neuron j already. Therefore, the BP algorithm starts with the output layer and iteratively computes the δ values for neurons in all layers. The weights can be updated right after one training pattern or after the whole set of training patterns.

It is common to add a momentum factor into the BP algorithm to increase its learning speed. The momentum factor determines how much the previous weight change influences the new weight change. The new equation for ΔW_{ji} is shown as follows

$$\Delta W_{ji}(k+1) = \eta \delta_j X_i + \mu \Delta W_{ji}(k) \tag{19}$$

where μ is the 'momentum' coefficient with value $0 \leq \mu \leq 1$.

The MLP neural network is to build a mapping relationship between input neurons and output neurons by

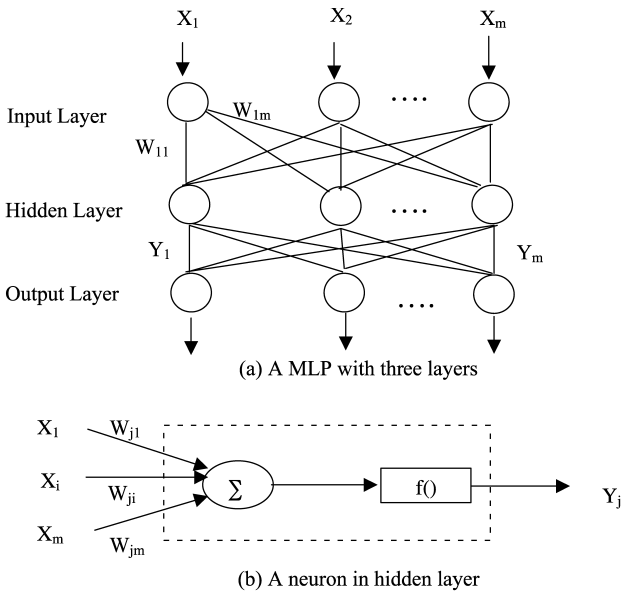


Fig. 9. Structure of MLP neural networks. (a) A MLP with three layers; (b) a neuron in hidden layer.

updating the weights associated with all the connections between layers.

2.6. Performance evaluation

The performance of a classifier is measured by its accuracy, which expresses the degree of success in correctly classifying new instances. Three methods are commonly applied for estimating the accuracy. They are holdout with random re-sampling, cross-validation, and bootstrap [4]. Because there is a relatively small dataset in the subject application, the bootstrap method was applied to obtain a more exact accuracy for a classification algorithm [20].

The bootstrap method randomly selects instances to form a training set from the raw dataset with replacement. The remaining instances become the testing set. θ_i' represents the accuracy estimator for the ith testing set. The process is repeated b times. The 0.632 bootstrap estimator θ_B' is defined as

$$\theta_B' = \frac{1}{b}\sum_{i=1}^{b}(0.632\theta_i' + 0.368\theta_c') \tag{20}$$

where δ_c' is the optimum accuracy estimator, which is calculated by using the initial data set for both training and testing purposes. This value is always equal to one for fuzzy K-NN and MLP neural network, because the testing data can easily achieve 100% accuracy simply by memorizing the training data.

3. Test results and discussion

3.1. Source of data

Four X-ray film strips of about 3.5 in. wide by 17 in. long each were digitized using the NDT SCAN II digitizer. The strips were digitized at 70 μm resolutions and saved as an image with 5000-pixel by 6000-line resolution. Every image thus contains four welds and each weld may contain either no defects or a number of different defects. Also, there are reference objects in the image to identify the position of defects, or to calibrate the X-ray. The digitized images were all initially stored in VICOM file format.

Human experts provided a priori knowledge about the location and type of defects in each image. The focus of this paper is to find the relationship between the characteristics of a weld defect and the type it belongs to. Therefore, each segment of a radiographic weld image, which contains a defect, was extracted manually along the edges of welding bead from the raw image. Totally 147 image segments covering six types of defects were extracted from 88 radiographic weld images. Fig. 10 shows the distribution of defect types. Sample defects of each type are shown in Fig. 11.

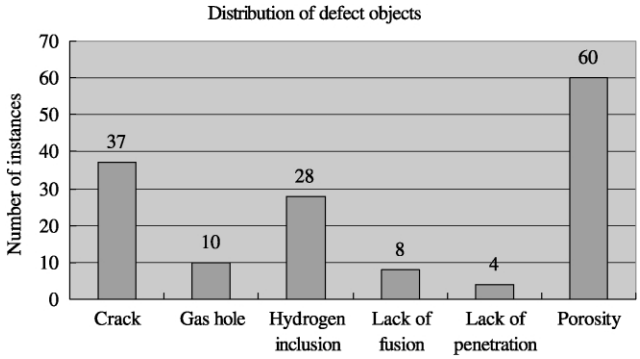


Fig. 10. Distribution of welding defects in the test data.

3.2. Results of MLP neural networks

Hundred and eight training data were selected from the entire set of 147 data and used to train the MLP neural network. Special attention was paid to ensure that instances of all different types are included. The remaining 39 data

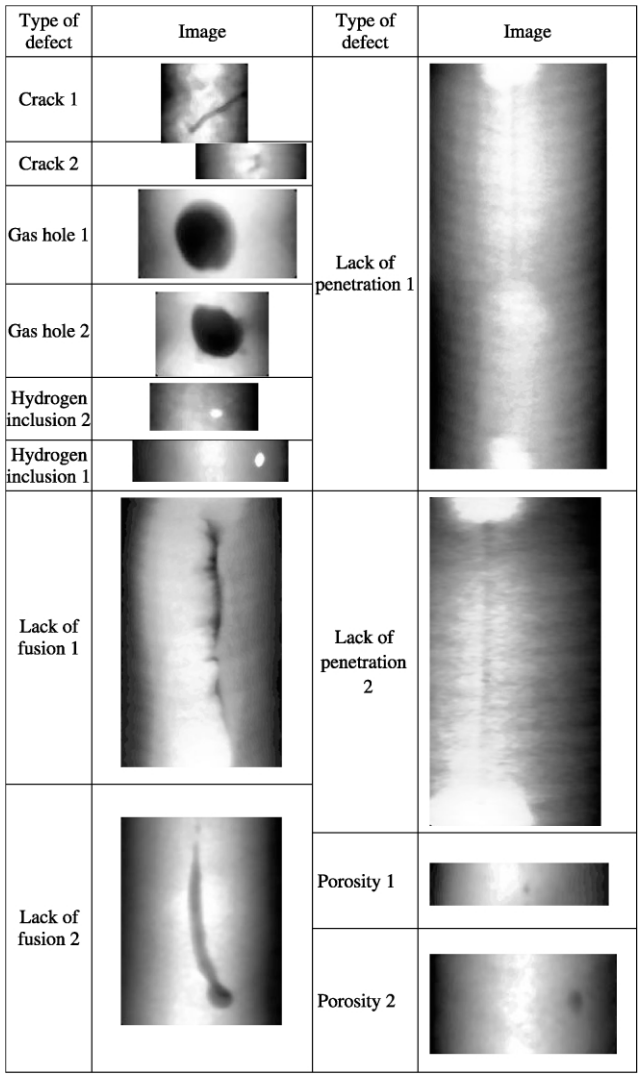


Fig. 11. Selected six types of welding defects.

Table 1
Test result for number of hidden nodes in MLP neural network

# of hidden nodes	Random seed	Training accuracy (%)	Testing accuracy (%)	Validation accuracy (%)
12	6	100.00	91.67	88.89
	12	98.15	91.67	88.89
	18	80.56	91.67	92.59
	24	84.26	83.33	92.59
	Average	90.74	89.58	90.74
24	6	100.00	91.67	88.89
	12	100.00	91.67	92.59
	18	100.00	83.33	92.59
	24	100.00	91.67	81.48
	Average	100.00	89.58	88.89
36	6	100.00	91.67	96.30
	12	100.00	91.67	88.89
	18	100.00	83.33	88.89
	24	100.00	91.67	85.19
	Average	100.00	89.58	89.81
48	6	100.00	91.67	88.89
	12	100.00	91.67	92.59
	18	100.00	91.67	96.30
	24	100.00	83.33	96.30
	Average	100.00	89.58	93.52
60	6	100.00	83.33	92.59
	12	100.00	91.67	92.59
	18	100.00	91.67	88.89
	24	100.00	83.33	96.30
	Average	100.00	87.50	92.59

were split into 12 and 27 for testing and validation. The testing and validation data were used to test the capability of trained neural networks. The output layer of all MLP neural networks has six nodes; each node corresponds to one defect type. The random seed and the number of nodes in the hidden layer are known to affect the performance of neural networks. Therefore, tests were performed to determine the optimum parameter values. Learning rate, which determines how large an adjustment MLP will make when it trains data, was set at the default value, 1.000. The default value of training tolerance, 0.100, was also consistently used. It means the output must be within 10% of the pattern range (5 for the subject application) to be considered 'good' in the training process. The testing tolerance was consistently set at 0.499, which means that a testing datum is considered 'good' if the predicted pattern is in the interval of $[0.501, 1.499]$ (or the error is <0.499). The results, as given in Table 1, indicate that among all those tested, 48 hidden nodes perform the best.

The final accuracy of the MLP neural network classifier is estimated by the statistical bootstrap method. In bootstrap, training set, testing set and validation set are required to be randomly selected from the original dataset for b times. Based on the accuracies from these b runs, the final accuracy is estimated. The results are shown in Table 2 for five runs. These results were obtained using MLP models with 48 hidden nodes. By plugging the average validation accuracy θ_i' for each run into Eq. (20), the accuracy

Table 2
Results of validation accuracy from MLP neural networks

Bootstrap	Random seed (%)				Average validation accuracy (θ_i') (%)	$0.632\theta_i'$	$0.368\theta_c'$	$\sum$
	6	12	18	24				
#1	81.48	85.19	81.48	88.89	84.26	0.533	0.368	0.901
#2	96.30	92.59	96.30	85.19	92.59	0.585	0.368	0.953
#3	88.89	88.89	85.19	88.89	87.96	0.556	0.368	0.924
#4	85.19	88.89	92.59	85.19	87.96	0.556	0.368	0.924
#5	88.89	85.19	88.89	85.19	87.04	0.550	0.368	0.918
						Accuracy estimate:		92.39%

θ_c' : optimistic accuracy is equal to 1.

Table 3
Accuracy for different k values in fuzzy K-NN

k	Testing accuracy					Validation accuracy				
	No. 1	No. 2	No. 3	No. 4	No. 5	No. 1	No. 2	No. 3	No. 4	No. 5
1	0.9167	0.8333	0.9167	0.9167	0.9167	0.8148	0.9259	0.8519	0.8889	0.8519
5	0.9167	0.8333	0.9167	0.9167	0.9167	0.7778	0.9259	0.8889	0.8889	0.8519
10	0.9167	0.8333	0.9167	0.8333	0.9167	0.6667	0.8889	0.8148	0.8148	0.8148
15	0.8333	0.6667	0.9167	0.8333	0.8333	0.7037	0.8889	0.8148	0.8519	0.7778
20	0.8333	0.6667	0.8333	0.8333	0.8333	0.7037	0.8889	0.8148	0.8519	0.7778
30	0.8333	0.6667	0.8333	0.8333	0.8333	0.7037	0.8519	0.7778	0.8519	0.7778

estimate of MLP neural networks is accordingly determined to be 92.39%.

3.3. Results of fuzzy K-NN

Similar to the MLP neural networks, the original data set of 147 feature vectors was randomly divided into training, testing, and validation subsets, with each having 108, 12, and 27 instances, respectively. Using a different number of nearest neighbors (parameter k) most likely will produce a different classification accuracy. Therefore, different K values were tested. The results are shown in Table 3 for testing accuracy and validation accuracy. The results indicate that smaller k values (1 or 5) give higher accuracy. Note also that the results of validation accuracy for $K = 15$ and 20 are identical.

The bootstrap accuracy estimate method was applied to fuzzy K-NN as well. The training and validation datasets in each run were identical to those used in MLP neural networks for fair comparison. In each run, the k value of 5 was consistently used. The final accuracy estimate for fuzzy K-NN based on the bootstrap method is 91.57%, as shown in Table 4.

3.4. Discussion

The test results indicate that the proposed methodology achieves a classification accuracy of $\sim 92\%$ for both classifiers used: MLP neural networks and fuzzy K-NN. Since that the a priori knowledge about these defect types were established by the human experts for this study, this performance can be interpreted as the 'likely' difference between the automated system and the human experts. Further study is necessary to determine the real difference.

Table 4
Bootstrap accuracy estimate on fuzzy K-NN

Bootstrap	Validation accuracy (%)	$0.632\theta_i'$	$0.368\theta_c'$	$\sum$
1	77.78	0.492	0.368	0.8596
2	92.59	0.585	0.368	0.9532
3	88.89	0.562	0.368	0.9298
4	88.89	0.562	0.368	0.9298
5	85.19	0.538	0.368	0.9064
			Accuracy estimate: 91.57%	

How is the performance of our system compared with previous studies? Unfortunately, no one knows the answer because different sets of radiographic images were used in different studies. This calls for the need to establish a benchmark image set in order to perform a sensible comparison. Is $\sim 92\%$ classification accuracy good enough? The answer depends on the intended use of the welded structure and the type, location, and size of the flaws missed, as well as on how costly a repair work is. Ideally, one would like to have a perfect automated RT system that gives 100% classification accuracy. Unfortunately, a realistic system will always fall short of that. Therefore, we should strive to develop an automated RT deemed considered trust worthy by human experts. Such a system might be imperfect and requires crosschecks by human experts some times.

4. Conclusions

The results obtained and the knowledge learned in this research can be summarized as follows:

1. Digital image processing techniques simulate the functions of human visions, which can derive useful information from images. In some aspects such as enhancement, digital image processing outperforms human visions by improving the quality of images.
2. The 3D fitting function algorithms implemented in TableCurve3D are proved to be effective for the construction of the background model for the background subtraction method. These algorithms give reasonable results with greater speed.
3. For the representation of defect objects, twelve features extracted describing the location, shape, size, and intensity information of defects are shown to be useful although perfect classification was not attained.
4. For the fuzzy K-NN classification algorithm, the k value could affect the accuracy as much as 15%. The proper selection of the k value is thus important. Unfortunately, there is no solution to determine the optimal value of parameter k for a given data set. A trial and error process

was implemented to find the best k value among those selected for testing.

5. For MLP neural networks, the number of hidden neurons is very critical to the training process. The determination of this parameter was based upon a trial and error process of several candidate values. In addition, different values of initial connection weights (created by using different vales of random seeds) could have large impacts on the training and testing accuracy. The average accuracy is thus often obtained by initializing MLP neural networks with several (4 in this study) random seeds.
6. MLP neural networks generally outperform the fuzzy K-NN algorithm in this application. Based on the bootstrap method, MLP neural networks trained with 108 defects achieved 92.39% classification accuracy (on an average 25 out of 27 defects for validation were correctly classified). On the other hand, fuzzy K-NN achieved 91.57% accuracy, which is slightly worse than MLP neural networks.

To enable comparison and to further advance this area of research, there is a need for establishing a benchmark image set.

References

[1] Aoki L, Suga Y. Application of artificial neural network to discrimination of defect type in automatic radiographic testing of welds. ISIJ Int 1999;39(10):1081–7.
[2] Chackalackal MS, Basart JP. NDE X-ray image analysis using mathematical morphology. Quant Nondestruct Eval 1990;9:721–8.
[3] Daum W, Rose P, Heidt H, Builtjes JH. Automatic recognition of weld defects in X-Ray inspection. Br J NDT 1987;29(2):79–82.
[4] Diamantidis NA, Karlis D, Giakoumakis EA. Unsupervised stratification of cross-validation for accuracy estimation. Artif Intell 2000;116:1–16.
[5] Edward GR. Inspection of welded joints. ASM handbook, welding, brazing and soldering, vol. 6. Materials Park, OH: ASM International; 1993. p. 1081–8.
[6] Gayer A, Saya A, Shiloh A. Automatic recognition of welding defects in real-time radiography. NDT Int 1990;23(3):131–6.
[8] Hall EL, Kruger RP, Dwyer SJ, Hall DL, Mclaren RW, Lodwick GS. A survey of preprocessing and feature extraction techniques for radiographic images. IEEE Trans Comput 1971;20(9):1032–44.
[9] Hyatt R, Kechter GE, Nagashima S. A method for defect segmentation in digital radiographs of pipeline girth welds. Mater Eval 1996;P:925–8.
[10] Hayes C. ABC's of nondestructive weld examination. Weld J 1997;76(5):46–51.
[11] Kato Y, Okumura T, Matsui S, Itoga K, Harada T, Sugimoto K, Michiba K, Iuchi S, Kawano S. Development of an automatic weld defect identification system for radiographic testing. Weld World 1992;30(7/8):182–8.
[12] Keller JM, Gray MR, Gigens Jr. JA. A fuzzy k-nearest neighbor algorithm. IEEE Trans Syst Man Cyber 1985;15(4):580–5.
[13] Liao TW, Li DM, Li YM. Detection of welding flaws from radiographic images with fuzzy clustering methods. Fuzzy Sets Syst 1999;108(2):145–58.
[14] Liao TW, Li DM, Li YM. Extraction of welds from radiographic images using fuzzy classifiers. Inform Sci 2000;126:21–42.
[15] Liao TW, Li YM. An automated radiographic NDT system for weld inspection. Part II. Flaw detection. NDT & E Int 1998;31(3):183–92.
[16] Liao TW, Ni J. An automated radiographic NDT system for weld inspection. Part I. Weld extraction. NDT & E Int 1996;29(3):157–62.
[17] Liao TW, Tang K. Automated extraction of welds from digitized radiographic images based on MLP neural networks. Appl Artif Intell 1997;11:197–218.
[18] Murakami K. Image processing for non-destructive testing. Weld Int 1990;4(2):144–9.
[19] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representation by error propagation. In: Rumelhart DE, McClelland JL, editors. Parallel distributed processing: explorations in the microstructure of cognition. Foundations, vol. 1. Cambridge, MA: MIT Press; 1986.
[20] Weiss SM, Kulikowski CA. Computer systems that learn: classification and prediction methods from statistics. Neural networks, machine learning and expert systems. San Mateo, CA: Morgan Kaufmann; 1991.