# Weld defect classification in radiographic images using unified deep neural network with multi-level features

Lu Yang[1] · Hongquan Jiang[1,2]

## Abstract

Deep neural network (DNN) exhibits state-of-the-art performance in many fields including weld defect classification. However, there is still a large room for improving the classification performance over the generic DNN models. In this paper, a unified deep neural network with multi-level features is proposed for weld defect classification. Firstly, we define 11 weld defect features as inputs of our proposed classification model. Not limited to geometric and intensity features, 4 features based on the intensity contrast between weld defect and its background are proposed in this paper. Secondly, we construct a novel deep learning framework: a unified deep neural network, where multi-level features of each hidden layer are fused by the last hidden layer to predict the type of weld defect comprehensively. In addition, we investigate pre-training and fine-turning strategies to get better generalization performance with small dataset. Comparing with other classification methods like SVM and generic DNN model, our framework takes full advantage of multi-level features extracted from each hidden layer, an outstanding performance is shown where the classification accuracy is improved by 3.18% and 4.33% on the test dataset, to reach 91.36%.

**Keywords** Non-destructive testing · Weld defect classification · Deep neural network · Multi-level features fusion · Stacked auto-encoder

## Introduction

Radiography testing (RT), one of the most important non-destructive testing (NDT) methods, plays an important role in the inspection of welds which are used in petroleum, nuclear and power generation industries (Nacereddine et al. 2019). Computer vision and machine learning methods are always used to detect and classify weld defects in an automated RT system. In general, there are five kinds of weld defects: porosity (PO), slag inclusion (SL), lack of penetration (LP), lack of fusion (LF) and crack (CR), as shown in Fig. 1. And the differences between some of them are not obvious, such as PO and SL, LP and LF. So improving the efficiency and accuracy of weld defects classification in radiographic images becomes an important technical problem in an automated RT system.

Instead of human eyes, an automated RT system which includes two parts are used to inspect weld defects, as shown in Fig. 2: (a) radiographic image processing, (b) weld defect pattern recognition (Da Silva and Mery 2007a, b). In part (a), image denoising and other image enhancement methods are implemented in radiographic image to get better image quality, and then weld defects are segmented from the improved radiographic image. In part (b), features are extracted from the segmented weld defect region, and the features are used for its classification. Our study mainly focuses on part (b): weld defect pattern recognition, which includes feature extraction and weld defect classification.

In recent years, various algorithms based on machine learning and statistics were used to classify weld defects from radiographic images. For example, artificial neural

✉ Lu Yang
  yanglu8823@163.com

  Hongquan Jiang
  jhqxjtu@163.com

1  State Key Laboratory for Manufacturing System Engineering, Xi'an Jiaotong University, Xi'an, China

2  Laboratory for Manufacturing and Productivity, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge 02139, MA, USA
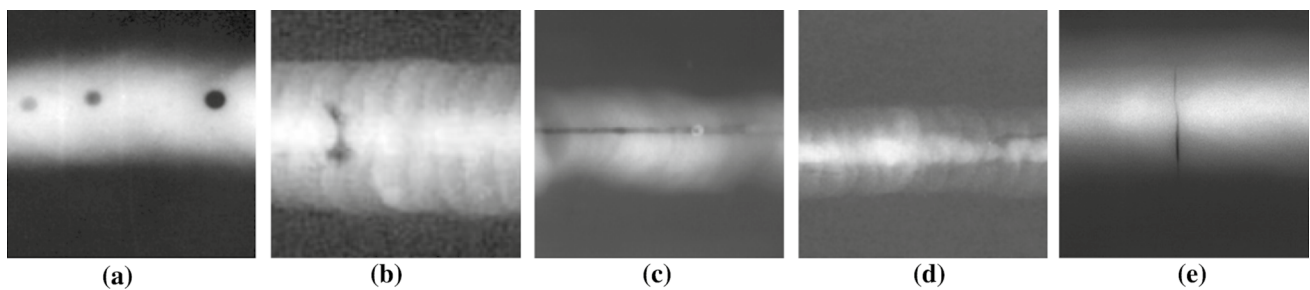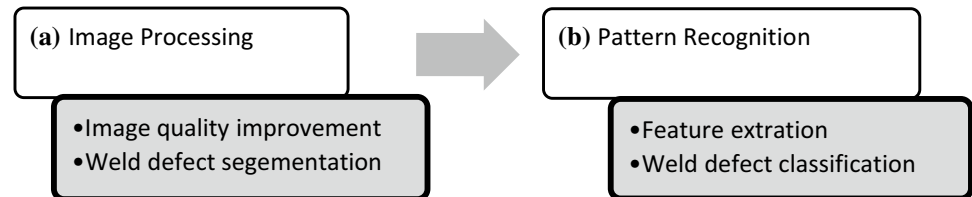
**Fig. 1** Weld defects: **a** porosity (PO); **b** slag inclusion (SL); **c** lack of penetration (LP); **d** lack of fusion (LF); **e** crack (CR)

**Fig. 2** Workflow of an automated RT system



**(a)** Image Processing
- Image quality improvement
- Weld defect segemention

**(b)** Pattern Recognition
- Feature extration
- Weld defect classification

network (ANN) with one hidden layer was used to classify weld defects, and principal component analysis (PCA) was used to reduce the number of features in the automated RT system (Vilar et al. 2009; Zapata et al. 2012). In Wang and Guo (2014), potential defects were detected, then support vector machine (SVM) was used to distinguish real defects from the potential ones. SVM was also used to classify weld defects, direct multiclass support vector machine (DMSVM) with higher accuracy and faster computation speed was proposed to classify defects (Shen et al. 2010). When there were a large amount of features, the SVM classifier for automatic weld defect classification was combined with PCA by which the number of features were reduced (Mu et al. 2013). Moreover, fuzzy reasoning methods were also popular for classifying weld defects. For instance, fuzzy k-nearest neighbor (kNN) with feature selection methods was proposed to improve classification accuracy (Liao 2009). In Liao (2003), a fuzzy expert system approach is proposed for the classification of different types of welding defects. In Baniukiewicz (2014), a complex classifier composed of ANN and a fuzzy logic system was proposed to classify weld defects. With the development of computer processing capability, deep neural network (DNN) with more hidden layers were widely used for pattern recognition, and it performed well in classifying weld defects (Hou et al. 2018). Among the machine learning models mentioned above, we notice that DNN classifier exhibits state-of-the-art performance. Comparing with other classifiers, DNN takes advantage of its architecture with large number of trainable variables, it gets higher accuracy both in training and testing especially than ANNs with shallow architectures. Consequently, we mainly focus on DNN classifier in this paper.

DNN has deep architectures containing multiple hidden layers like the human neural systems, and it has been shown perfect classification performance in many fields when the training examples are enough (Richardson et al. 2015; Yao et al. 2020). However, there are two big problems to train DNNs: one is the vanishing gradients with the increment of the hidden layer number, the other is the traps of poor local minimums (Schmidhuber 2015; Lecun et al. 2015). In order to solve the above problems, greedy layer-wised pre-training methods like restricted Boltzmann machine (RBM) (Hinton and Salakhutdinov 2006; Hinton et al. 2006) and stacked auto-encoder (SAE) (Yuan et al. 2018) was proposed. SAE, the unsupervised pre-training method, encodes the input data from a high-dimensional space into a low-dimensional space and then decodes the low-dimensional space data into a high-dimensional space stack by stack (Jia et al. 2016). Moreover, pre-training method like SAE along with fine-tuning processes also tackle the problem: DNN classifier gets poor classification performance when the training dataset is small (Feng et al. 2019). In our study, the resolution of radiographic image for weld seam is $7500 \times 2048$, so it's hard to collect and assemble big dataset for weld defects. In this paper, we investigate SAE for pre-training and fine-turning strategies to train a DNN aiming at getting better classification performance with small weld defects dataset. DNNs contain rich information in hidden layers, concrete information in low-level hidden layers and abstract one in high-level hidden layers. Multi-level features aggregating framework achieved a better classification performance than generic ones (Zhao et al. 2015; Wang et al. 2015; Zhang et al. 2017). In our work, we construct a unified deep neural network to improve the classifier's performance, which

extracts multi-level features from each hidden layer and fuses them to characterize weld defects comprehensively.

In this paper, a unified deep neural network with multi-level features is proposed for weld defect classification, pre-training methods like SAE and fine-tuning strategies are investigated to improve the classifier's performance with small dataset. The workflow of classification is given in detail in the following parts. Firstly, we'll talk about features extraction in Sect. 2. 11 weld defect features are extracted in radiographic image as inputs of our unified deep neural network, where 4 of them based on the intensity contrast than its background are proposed in this paper. Secondly, in Sect. 3, the unified deep neural network which we propose is introduced, where multi-level features extracted from each hidden layer are combined by the last hidden layer to predict the type of weld defect comprehensively. Moreover, we investigate pre-training and fine-turning strategies to get better generalization performance. Thirdly, a case study about weld defect classification is shown to illustrate our work in Sect. 4. Finally, in Sect. 5, conclusions and suggestions are given for future research.

## Features extraction

In previous works, features extracted from weld defect are mainly based on its geometric and intensity properties (Da Silva and Mery 2007b). For a segmented weld defect region, the geometric features are related to its size, shape and contour, while the intensity features are related to its gray-level distribution. We know that the characteristics based on intensity contrast are very useful to classify weld defects due to different types of weld defects tend to have a different gray value distribution than the background behind them. However, they are omitted for a long time. Consequently, 4 features based on the intensity contrast between weld defect and its background

are proposed to contribute for classification. Together with 7 weld defect features which perform well in previous studies (Shen et al. 2010; Jiang et al. 2019), we totally define 11 features as inputs of our unified deep neural network for classification in this paper. The first 7 features are selected from previous works and the last 4 features are newly defined as shown in Table 1.

To extract the contrast features, we define $W_d$ as a window of the weld defect minimum enclosing rectangle and $W_b$ as a window which is a rectangle obtained by enlarging the $W_d$ by a factor $\theta$ in all directions as shown in Eq. (1). Figure 3 shows the details, the red rectangle is $W_d$ and the green rectangle is $W_b$. In this paper, we set that $\theta$ equals to 1.2.
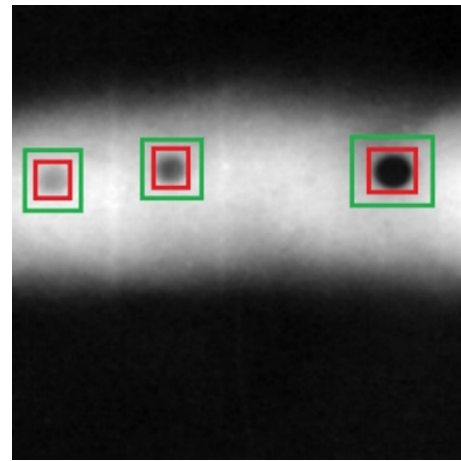
$$\frac{|W_b|}{|W_d|} = \theta^2 \tag{1}$$



**Fig. 3** Weld defect window and background window: red one is weld defect window, green one is background window (Color figure online)

**Table 1** Defect features

|  | Feature | Feature type | Definition |
|---|---|---|---|
| W | Length | Geometric | The maximum distance between two points in the defect edge |
| $A_r$ | Aspect ratio | Geometric | Describe the ratio between the width and length of the defect |
| $S_p$ | Sharpness | Geometric | The keenness of the defect at the two ends along the major axis |
| $R_e$ | Roughness of defect edge | Geometric | The variation (corresponding to the profile's high frequencies that are closely related to the types of the defect) of the defect profile |
| $R_r$ | Roughness of defect region | Intensity | The intensity variation of the pixels in the internal region of the defect |
| $S_k$ | Skewness | Intensity | The symmetry of the topographic map of the defect region along the major axis |
| $K_u$ | Kurtosis | Intensity | Describe the wedge angle of the topographic map of the defect region |
| $H_c$ | Histogram contrast | Contrast | Chi square distance of gray-level histograms between defect region and its background |
| $R_c$ | Roughness contrast | Contrast | The roughness ratio of defect region to its background |
| $S_c$ | Skewness contrast | Contrast | The skewness ratio of defect region to its background |
| $K_c$ | Kurtosis contrast | Contrast | The kurtosis ratio of defect region to its background |

where $\theta$ is the enlarging factor, $|\cdot|$ is the number of pixels, $W_b$ is the background window, $W_d$ is the weld defect window.

It is important to note that weld defect window is different from weld defect region, the former one is a rectangle and only used to get background window, while the latter one can be arbitrary appearance depends on segmentation results of weld defect and is used to calculate the 11 features mentioned above. Above all, we propose 4 new features based on intensity contrast as follows.

(1) Histogram contrast ($H_c$): Histogram contrast calculates Chi square distance of gray-level histograms between defect region and its background, and it is defined as below:

$$H_c = \chi^2\big(h(R_d), h\big(W_b\big(W_d, \theta\big)\big)\big) \qquad (2)$$

where $R_d$ is the segmented area of weld defect, $h$ is the gray-level histogram, $\chi^2$ is the Chi square distance.

(2) Roughness contrast ($R_c$): the roughness ratio of defect region to its background, and it is defined as:

$$R_c = R_r/R_r' \qquad (3)$$

where the $R_r$ is the roughness of defect region, $R_r'$ is the roughness of its background.

(3) Skewness contrast ($S_c$): the skewness ratio of defect region to its background, and it is defined as:

$$S_c = S_k/S_k' \qquad (4)$$

where the $S_k$ is the skewness of defect region, $S_k'$ is the skewness of its background.

(4) Kurtosis contrast ($K_c$): the kurtosis ratio of defect region to its background, and it is defined as:

$$K_c = K_u/K_u' \qquad (5)$$

where the $K_u$ is the kurtosis of defect region, $K_u'$ is the kurtosis of its background.

## The method of classification

DNNs contain rich information in each hidden layer. In our model, we extract multi-level features and fuse them to predict the type of weld defects comprehensively. In addition, supervised learning methods (SAE) and unsupervised learning methods are combined, we investigate pre-training and fine-turning strategies for training to improve the classifier's performance in this paper. In this section, firstly, a novel architecture of the unified deep neural network with multi-level features is discussed. Secondly, pre-training and fine-turning strategies for training is introduced. Finally, workflow of classification is shown in the last part.

## A unified deep neural network with multi-level features

DNN is state-of-the-art architecture for many applications, such as speech recognition (Hinton et al. 2012), face recognition (Mai et al. 2017), pose estimation (Ahn et al. 2018) and so on. It is a large-scale nonlinear system composed by many neural cells like human brain and it has many hidden layers between input layer and output layer as shown in Fig. 4. In DNN models, forward propagation is used to get the output value, while backward propagation is used to optimize model's parameters.

In our work, we extract rich information of weld defects in each hidden layer and fuse them in the last hidden layer through the novel architecture (a unified deep neural network with multi-level features) we proposed as shown in Fig. 5. In our model, we totally have 6 layers (11–6–5–5–10–5) including one input layer, one output layer and 4 hidden layers. Different hidden layer contains abstract features in different levels, we fuse them in the last hidden layer for classification to improve the classifier's performance. The 11 extracted features which are introduced in Sect. 2 are normalized. The features are contributed to the input layer, and then transmitted to hidden layer 1 by forward propagation algorithm. Each neuron in the next layer depends on its previous layer and can be calculated according to Eqs. (6) and (7). In particular, the purple neurons of hidden layer 4 can be calculated according to Eqs. (6) and (8), and the red neurons of hidden layer 4 can be calculated according to Eqs. (6) and (9), and so on. On one hand, the neurons in hidden layer 1 propagate to hidden layer 2, on the other, neurons from hidden layer 1 also propagate to hidden layer 4 directly. In the same way, neurons in hidden layer 2 propagate to hidden layer 3 and hidden layer 4. In the last hidden layer (hidden layer 4), ten neurons with three levels are combined to predict the type of weld defect comprehensively. Neurons of the output layer range from 0 to 1, the maximum of which represent the hypothesis of weld defect type: porosity (PO), slag inclusion (SL), lack of penetration (LP), lack of fusion
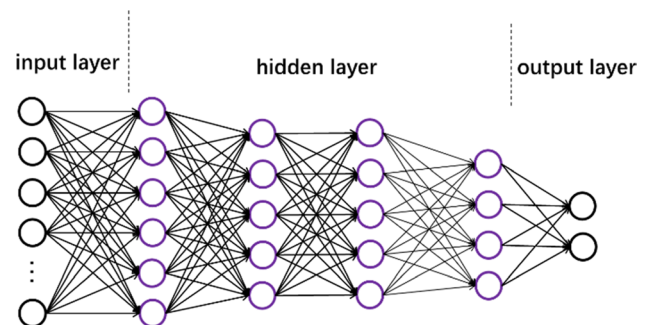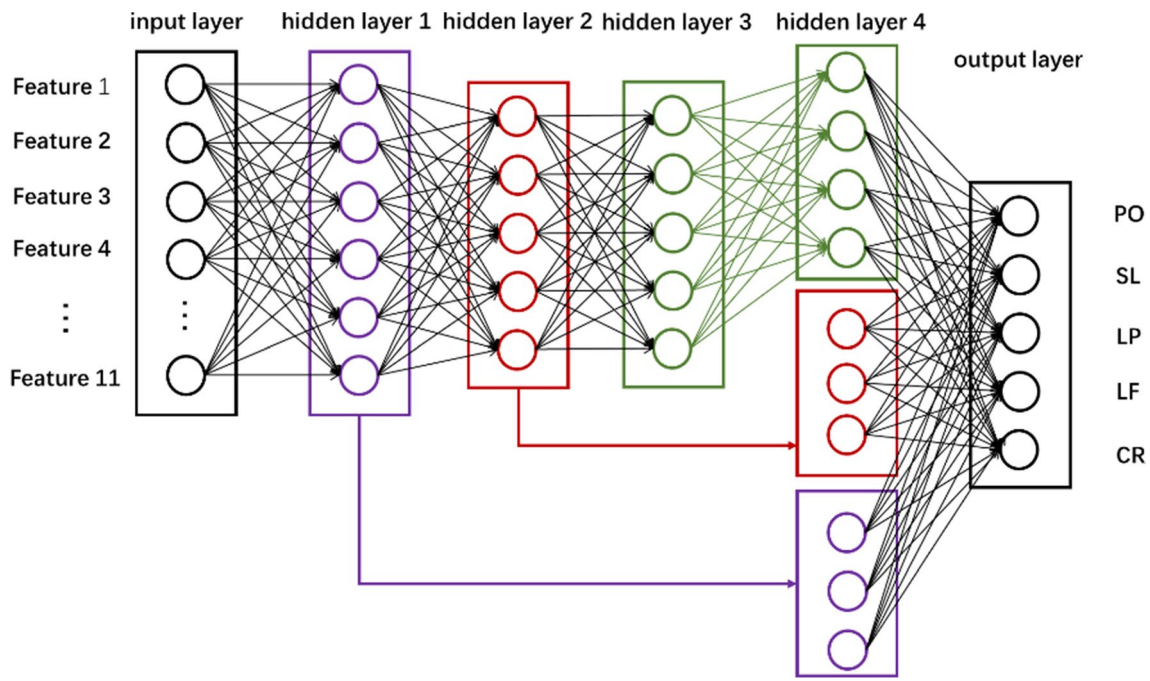


**Fig. 4** DNN architecture

**Fig. 5** A unified deep neural network with multi-level features

(LF), crack (CR). The deep neural framework we propose takes full advantage of multi-level features extracted from each hidden layer for fusion to classify weld defect more accurately.

$$a_j^{(l+1)} = g\left( \sum_i z_{ij} + b_j^{(l+1)} \right) \quad (6)$$

$$z_{ij} = a_i^{(l)} w_{ij}^{(l,l+1)} \quad (7)$$

where $a_j^{(l)}$ is the $j$th neuron of the $l$th layer, $g(x)$ is the activation function, $w_{ij}^{(l,l+1)}$ is the weight and $b_j^{(l+1)}$ is the bias between the $l$th layer and the $(l+1)$th layer.

In Fig. 5, to calculate the purple neurons of hidden layer 4, Eq. (7) should be replaced by Eq. (8). Similarly, to calculate the red neurons of hidden layer 4, Eq. (7) should be replaced by Eq. (9).

$$z_{ij} = a_i^{(l-2)} w_{ij}^{(l-2,l+1)} \quad (8)$$

$$z_{ij} = a_i^{(l-1)} w_{ij}^{(l-1,l+1)} \quad (9)$$

DNNs are always trained with backpropagation algorithm which constitute the best example of a successful gradient-based learning technique (Lecun et al. 1998). In our work, we construct a special deep learning framework with forward propagation algorithm and train it with backpropagation algorithm. Firstly, we define the cost function for our unified deep neural networks with regularization term, and it is given as Eq. (10). Then we initialize model parameters and do some pre-train, after that, we implement the backpropagation algorithm to compute the gradient for the deep neural network cost function. Thirdly, we use gradient descent algorithm to minimize the cost function. We update model parameters according to Eqs. (11) and (12), then compute the gradient again to minimize the cost function until convergence.

$$J(W,b) = \frac{1}{2m} \sum_{i=1}^{m} \sum_{k=1}^{K} \left[ \left( h_\theta(x^i) \right)_k - y_k^{(i)} \right]^2 + \frac{\lambda}{2m} \sum_l \omega_2^2 \quad (10)$$

where $J(W,b)$ is the loss, $m$ is the number of training sets, $k$ is the number of types of weld defects, $x^i$ is the input features of $i$th training example, $\left( h_\theta(x^i) \right)_k$ is the $k$th neuron of the output layer which is the hypothesis for the input $x^i$, $y_k^{(i)}$ is the $i$th element of label $y_k$, $\lambda$ is regularization factor, $W$ is the weight matrix, $\omega$ is the weight vector for each layer, $b$ is the bias vector, $l$ represent the layer of model.

$$\omega \to \omega - \alpha \frac{\partial J}{\partial \omega} \quad (11)$$

$$b \to b - \alpha \frac{\partial J}{\partial b} \quad (12)$$

where $\alpha$ is learning rate.

In parameter initialization process, pre-training strategy (SAE) are investigated to escape from local optimal solution, and to avoid the vanishing gradients with the increment of the hidden layer number. After pre-training, we do some fine-tunings to train the proposed models. Pre-training and fine-tuning processes make the classifier get better performance with small training set. The details are introduced in the next part.

## Pre-training and fine-turning

Pre-training can improve generalization performance and reduce the possibility of overfitting. It is utilized to initialize deep learning framework with optimized model parameters and it helps the fine-tuning step escape from local optimal solution. SAE is a kind of unsupervised pre-training algorithm. Figure 6 shows how to pre-train a DNN by SAE in detail.

In Fig. 6, the upper half shows a DNN's structure and the lower half shows 2 auto-encoders' structures. For the first auto-encoder on the left, its input vector and the neuron number of hidden layer is the same as the DNN's, the neuron number of its output layer is the same as its input layer. By initializing the weights and bias of the auto-encoder, its neuros of hidden layer is calculated according to Eq. (13) and its output is calculated by Eq. (14). The weights and bias of the auto-encoder could be optimized by minimizing the loss as shown in Eq. (15). An auto-encoder aims to construct a mapping which can change the inputs from one form to another without loss as far as possible. After optimization, auto-encoder's weights and bias between its input layer and its hidden layer are used to initial the DNN's corresponding parameters. The previous auto-encoder's hidden layer provides the next auto-encoder with the input. In general, each

trained auto-encoder devotes its weights and biases to the DNN's corresponding layer for initialization.

$$h = g(Wx + b) \tag{13}$$

$$\hat{x} = g'\left(W'h + b'\right) \tag{14}$$

$$L\left(W, b, W', b'\right) = \frac{1}{m} \sum_{i=1}^{m} x - \hat{x}_2^2 \tag{15}$$

where $x$ is the input vector of an auto-encoder, $\hat{x}$ is the output vector, $h$ is the hidden layer vector, $g(x)$ and $g'(x)$ are activation function, $L\left(W, b, W', b'\right)$ is the loss function, $m$ is the number of input examples.

Generally speaking, our model pre-trained by SAE is in the same way as DNNs. 11 features mentioned above which are normalized are treated as inputs of the first auto-encoder, and the other auto-encoders are trained in order. However, for pre-training, there is a little difference in the last hidden layer which combines features from three levels. Three corresponding trained auto-encoder provides its weights and biases to the last hidden layer respectively. After our model is pre-trained, it is fine-tuned by the gradient descent algorithm in next step of training.

## Workflow of classification

The workflow of weld defect classification proposed in this paper is shown as Fig. 7. Our work mainly focused on the steps of the workflow which are encircled by the blue rectangular box. Firstly, the weld defects are segmented after radiographic image improvement. Secondly, 11 weld defect features are extracted for pre-processing (normalization). Thirdly, the feature dataset is divided



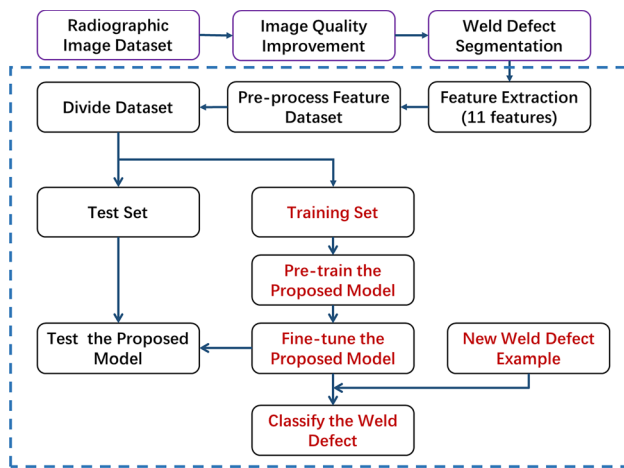**Fig. 6** Stacked auto-encoders for DNNS pre-training

**Fig. 7** Workflow of weld defect classification
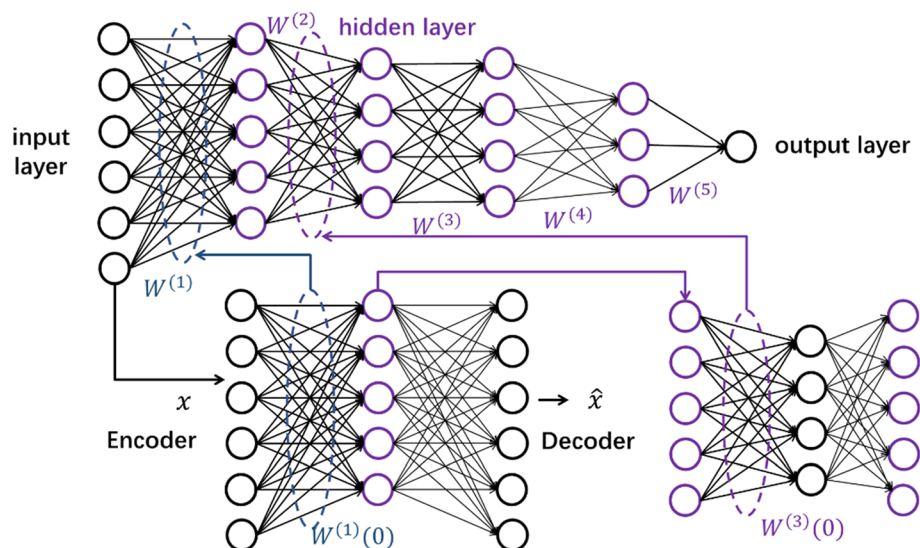


**Fig. 8** Workflow of case study section

into two parts: the training set and the test set. Finally, our model is pre-trained by SAE and fine-tuned with training set. And the test set is utilized for testing. When there is a new weld defect example, it could be classified by our trained model with its 11 extracted features. In short, the workflow of weld defect classification are from features extraction to new weld defect classification in this paper, and it is described in a more simple way in the next paragraph.

The main steps of our workflow of classification is given as follows:

(1) Feature extraction: 11 weld defect features.
(2) Feature pre-processing: normalization.
(3) Dataset Division: training set and test set.
(4) Train the proposed model: pre-training and fine-tuning.
(5) Test the proposed model.
(6) Classification: new weld defect example.

## Case study

Dataset description, weld defect classification with our model and models' performance comparison are introduced in this section as shown in Fig. 8. Firstly, features are extracted and pre-processed for dataset assembling, then dataset are divided into training set and test set. Secondly, our proposed model are pre-trained and fine-tuned with training set, and weld defects in test set are classified to test the model's performance. Finally, classification performance of the three models: DNNs, SVMs and our model are compared.
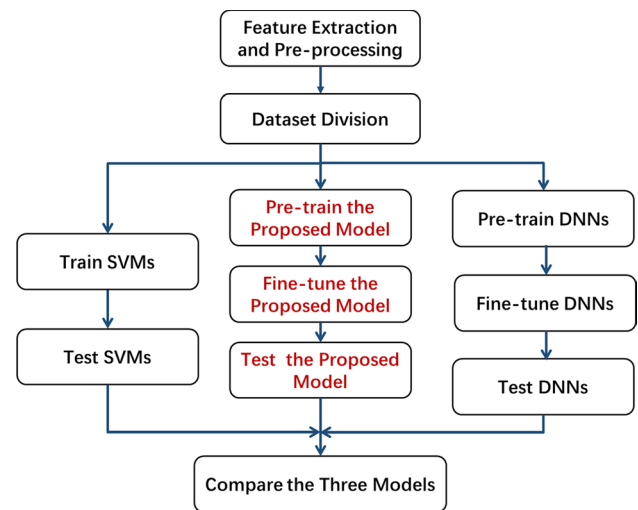
## Dataset description

In our work, dataset including 220 samples with 5 types of weld defect are collected to train and test the classifier which we propose. To ensure there are no overlap between training and testing datasets, cross validation strategy is implemented. The training and testing results are achieved by a fivefold cross-validation. We divide the whole dataset randomly into 5 subsets. Each subset takes its turn as the test set while the remaining subsets are combined for training. And each training set and test set have the same proportion of the 5 type weld defects as the whole dataset. As a result, 20% samples of the dataset are working as the testing set and the rest 80% samples are used as the training set. As shown in Table 2, the whole dataset includes of 220 examples: 50 PO, 50 SL, 50 LP, 35 LF and 35 CR. For each round of cross validation, 176 samples are used for training with the proportion: 40 PO, 40 SL, 40 LP, 28 LF and 28 CR, 44 samples are used for the test set. Within the dataset, 11 features mentioned in Sect. 2 are extracted from each weld defect example, and each feature is normalized.

## Classification performance of our model

In our work, Rectified Linear Unit (ReLU) as shown in Eq. (16) is used as the activation function for each hidden layer, which solves the problem of gradient elimination and has a fast convergence speed (Krizhevsky et al. 2017). Softmax function as shown in Eq. (17) and its loss function (cross entropy loss) are utilized for the final output layer. A fivefold cross-validation method is implemented to train and test our model, and we carry out the classifying experiments for five rounds. In each run, there are no overlap between training set and test set. Firstly,

**Table 2** Weld defects dataset distribution

|  | Total | PO | SL | LP | LF | CR |
|---|---|---|---|---|---|---|
| Total dataset | 220 | 50 | 50 | 50 | 35 | 35 |
| Training set | 176 | 40 | 40 | 40 | 28 | 28 |
| Test set | 44 | 10 | 10 | 10 | 7 | 7 |

**Table 3** Confusion matrix for training

| Confusion matrix | | | | | |
|---|---|---|---|---|---|
| Predictive value | True value | | | | |
|  | PO | SL | LP | LF | CR |
| PO | 40 | 0 | 0 | 0 | 0 |
| SL | 0 | 40 | 0 | 4 | 0 |
| LP | 0 | 0 | 40 | 0 | 0 |
| LF | 0 | 0 | 0 | 24 | 0 |
| CR | 0 | 0 | 0 | 0 | 28 |

**Table 4** Confusion matrix for testing

| Confusion matrix | | | | | |
|---|---|---|---|---|---|
| Predictive value | True value | | | | |
|  | PO | SL | LP | LF | CR |
| PO | 10 | 0 | 0 | 0 | 0 |
| SL | 0 | 9 | 0 | 2 | 0 |
| LP | 0 | 0 | 10 | 0 | 0 |
| LF | 0 | 1 | 0 | 5 | 0 |
| CR | 0 | 0 | 0 | 0 | 7 |

samples in training set are utilized to pre-train our model with SAE layer by layer, and then they are utilized for fine-tuning. After training, we test our model by indicator which is commonly utilized for classifier performance evaluation such as classification accuracy. For each type of weld defect, classification accuracy defined as Eq. (18) is utilized to evaluate a 2-class classifier: one vs all, in addition, it is also used to evaluate the overall classification performance for a multi-class classifier.

$$g(x) = \max(0, x) \tag{16}$$

$$S_i = \frac{e^i}{\sum_j e^j} \tag{17}$$

where $g(x)$ is activation function, $x$ is the input variable, $i$ is i-th type of weld defect, $j$ is j-th type of weld defect and belongs to [1, 5], $S_i$ is the hypothesis of i-th type weld defect.

$$\text{Accuacy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{18}$$

where $TP$ is the number of true positive samples, $TN$ is the number of true negative samples, $FP$ is the number of false positive samples, $FN$ is the number of false negative samples.

Confusion matrix for training and testing are utilized to calculate the accuracy. In one round of cross validation, it is created as shown in Tables 3 and 4.

For each type of weld defect, we calculate its training and testing accuracy based on confusion matrix mentioned above according to Eq. (18). Overall classification accuracy is also calculated in each run of the experiment. At last, we get average accuracy as shown in Table 5.

For our whole model, average training accuracy is 97.95%, average testing accuracy is 91.36%. It shows that our model achieves perfect training performance. From Table 5, PO, LP and CR get 100% training accuracy. In addition, the results indicate that our model get a high testing performance, PO and CR even get 100% test accuracy. In general, the classification results show our model performs very well in both training and testing with a 220-sample dataset. Due to our model takes advantage of multi-level features extracted from each hidden layer and investigates SAE in pre-training, it classifies weld defect with high precision and gets perfect generalization performance with small dataset.

## Models' performance comparison

In this part, a fivefold cross-validation method is carried out to train and test classifiers, SVM and DNN models are utilized to classify weld defects with the same dataset at each round as it mentioned above including 176 training samples and 44 testing samples. The SVM model utilizes Gauss kernel function as kernel function for training and testing. The DNN model is a generic one with 6 layers (11–6–5–5–4–5) including one input layer, one output layer and 4 hidden layers, pre-training method (SAE) and fine-tune strategy are utilized for training as shown in Fig. 8. Table 6 shows average classification accuracy of each type of weld defect for training and testing with the three models in five rounds. Table 7 shows classification performance comparison among the three models. The values marked in red are the highest scores for each type of indicator.

**Table 5** Training and testing accuracy for each type of weld defect and the overall model

| Experiment | Training accuracy (%) | | | | | | Testing accuracy (%) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PO | SL | LP | LF | CR | Total | PO | SL | LP | LF | CR | Total |
| # 1 | 100 | 100 | 100 | 85.71 | 100 | 97.73 | 100 | 90 | 100 | 71.43 | 100 | 93.18 |
| # 2 | 100 | 97.5 | 100 | 82.14 | 100 | 96.59 | 100 | 80 | 100 | 71.43 | 100 | 90.91 |
| # 3 | 100 | 92.5 | 100 | 89.29 | 100 | 96.59 | 100 | 80 | 90 | 85.71 | 100 | 90.91 |
| # 4 | 100 | 97.5 | 100 | 100 | 100 | 99.43 | 100 | 80 | 100 | 71.43 | 100 | 90.91 |
| # 5 | 100 | 100 | 100 | 96.43 | 100 | 99.43 | 100 | 80 | 90 | 85.71 | 100 | 90.91 |
| Average | 100 | 97.5 | 100 | 90.71 | 100 | 97.95 | 100 | 82 | 96 | 77.14 | 100 | 91.36 |

**Table 6** Accuracy of each type of weld defect for training and testing with different models

| Experiment | PO | | | SL | | | LP | | | LF | | | CR | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SVM | DNN | Ours | SVM | DNN | Ours | SVM | DNN | Ours | SVM | DNN | Ours | SVM | DNN | Ours |
| Training accuracy (%) | | | | | | | | | | | | | | | |
| # 1 | 100 | 100 | 100 | 97.5 | 87.5 | 100 | 100 | 100 | 100 | 85.71 | 100 | 85.71 | 100 | 100 | 100 |
| # 2 | 97.5 | 100 | 100 | 95 | 97.5 | 97.5 | 100 | 100 | 100 | 85.71 | 60.71 | 82.14 | 100 | 100 | 100 |
| # 3 | 97.5 | 100 | 100 | 90 | 100 | 92.5 | 100 | 100 | 100 | 78.57 | 60.71 | 89.29 | 100 | 100 | 100 |
| # 4 | 100 | 100 | 100 | 90 | 97.5 | 97.5 | 100 | 100 | 100 | 82.14 | 100 | 100 | 100 | 100 | 100 |
| # 5 | 97.5 | 100 | 100 | 95 | 92.5 | 100 | 100 | 100 | 100 | 82.14 | 85.71 | 96.43 | 100 | 100 | 100 |
| Average | 98.5 | 100 | 100 | 93.5 | 95 | 97.5 | 100 | 100 | 100 | 82.85 | 81.43 | 90.71 | 100 | 100 | 100 |
| Testing accuracy (%) | | | | | | | | | | | | | | | |
| # 1 | 100 | 100 | 100 | 70 | 50 | 90 | 100 | 90 | 100 | 71.43 | 71.43 | 71.43 | 100 | 100 | 100 |
| # 2 | 100 | 100 | 100 | 80 | 80 | 80 | 100 | 100 | 100 | 57.14 | 71.43 | 71.43 | 100 | 100 | 100 |
| # 3 | 90 | 80 | 100 | 90 | 90 | 80 | 100 | 100 | 90 | 85.71 | 85.71 | 85.71 | 100 | 100 | 100 |
| # 4 | 100 | 90 | 100 | 80 | 100 | 80 | 90 | 100 | 100 | 71.43 | 71.43 | 71.43 | 85.71 | 85.71 | 100 |
| # 5 | 100 | 100 | 100 | 50 | 60 | 80 | 100 | 90 | 90 | 71.43 | 57.14 | 85.71 | 100 | 100 | 100 |
| Average | 98 | 94 | 100 | 74 | 76 | 82 | 98 | 96 | 96 | 71.43 | 71.43 | 77.14 | 97.14 | 97.14 | 100 |

**Table 7** Different models' classification performance comparison

| Average accuracy | SVM (%) | DNN (%) | Ours (%) |
|---|---|---|---|
| Training | 95.45 | 95.91 | 97.95 |
| Testing | 88.18 | 87.03 | 91.36 |

As shown in Table 7, our model's training accuracy is 97.95% and testing accuracy is 91.36%. The results show that our model gets the perfect classification performance both in training and testing among the three models. For each type of weld defect, our model also gets the perfect classification performance in training as shown in Table 6, in addition, except for LP, our model gets the highest average accuracy in testing. Especially for PO and CR, our model gets a 100% score of average accuracy in training and testing. Generally speaking, comparing with SVM and DNN model, our classification model shows an outstanding performance where the classification accuracy is improved by 2.5% and 2.04% for training, it is improved by 3.18% and 4.33% for testing.

## Conclusion

In this paper, a unified deep neural network with multi-level features is proposed for weld defect classification. Firstly, 11 weld defect features are extracted for pre-processing (normalization), where 4 of them based on the intensity contrast than its background are proposed. Secondly, cross validation strategy is implemented, we divide the whole dataset randomly into 5 subsets, each subset takes its turn as the test set while the remaining subsets are combined for training. For each round of cross validation, the feature dataset including 220 samples is divided into two parts: the training set with 176 samples and the test set with 44 samples as shown in Table 2. Thirdly, we construct our unified deep neural network with multi-level features. It is pre-trained by SAE and fine-tuned with training set, then it is tested by the trained model with test set. Training and testing results are recorded. Finally, for comparison, SVM and DNN models are utilized to classify weld defects with the same dataset. The case study result shows that our model could fuse multi-level features to predict the type

of weld defect comprehensively, pre-training (SAE) and fine-turning strategies make our model get better generalization performance with small dataset. Comparing with SVM and generic DNN model, our classification model shows an outstanding performance where the classification accuracy is improved by 3.18% and 4.33% for testing, to reach 91.36%.

Our work has achieved good performance in the classification of weld defects. For future research, more works should be done to improve the classification performance. Three aspects of future study are discussed as follows:

(1) Further research on systematic optimization of our model is necessary to get higher accuracy and faster convergence speed. Due to the structure of our model, the cross validation process and parameter initialization rely on user's experience, and they all affect the accuracy and speed of classification. Consequently, research on systematic optimization of our model is needed.
(2) Normalization methods used in data pre-processing step should be studied to tackle the 'out of range' problem of 'test data'. In practice, 'test data' are normalized before importing into our classification model, however, the parameters for normalization relays on training sets and it may lead to the 'out of range' problem of 'test data'. So normalization methods require to be further studied.
(3) To realize weld defect classification automatically, methods about obtaining weld defect region candidates need to be studied. In this paper, weld defect classification is implemented when the weld defect candidates already exists. In real application, weld defect regions of interest should be obtained automatically for the test data. Research is needed on the method of obtaining weld defect region candidates automatically.

# References

Ahn, B., Choi, D., Park, J., & Kweon, I. S. (2018). Real-time head pose estimation using multi-task deep neural network. *Robotics and Autonomous Systems, 103,* 1–12. https://doi.org/10.1016/j.robot.2018.01.005.

Baniukiewicz, P. (2014). Automated defect recognition and identification in digital radiography. *Journal of Nondestructive Evaluation, 33*(3), 327–334. https://doi.org/10.1007/s10921-013-0216-6.

Da Silva, R. R., & Mery, D. (2007a). The state of the art of weld seam radiographic testing: Part I-image processing. *Materials Evaluation, 65*(6), 643–647.

Da Silva, R. R., & Mery, D. (2007b). The state of the art of weld seam radiographic testing: Part II-pattern recognition. *Materials Evaluation, 65*(9), 833–838.

Feng, S., Zhou, H., & Dong, H. (2019). Using deep neural network with small dataset to predict material defects. *Materials and Design, 162,* 300–310. https://doi.org/10.1016/j.matdes.2018.11.060.

Hinton, G. E., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-R., Jaitly, N., et al. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine, 29*(6), 82–97. https://doi.org/10.1109/MSP.2012.2205597.

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation, 18*(7), 1527–1554. https://doi.org/10.1162/neco.2006.18.7.1527.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science, 313*(5786), 504–507. https://doi.org/10.1126/science.1127647.

Hou, W., Wei, Y., Guo, J., & Jin, Y. (2018). Automatic detection of welding defects using deep neural network. *Journal of Physics Conference Series, 933,* 012006. https://doi.org/10.1088/1742-6596/933/1/012006.

Jia, F., Lei, Y., Lin, J., Zhou, X., & Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing, 72,* 303–315. https://doi.org/10.1016/j.ymssp.2015.10.025.

Jiang, H., Wang, R., Gao, Z., Gao, J., & Wang, H. (2019). Classification of weld defects based on the analytical hierarchy process and Dempster–Shafer evidence theory. *Journal of Intelligent Manufacturing, 30*(4), 2013–2024. https://doi.org/10.1007/s10845-017-1369-4.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386.

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324. https://doi.org/10.1109/5.726791.

Liao, T. W. (2003). Classification of welding flaw types with fuzzy expert systems. *Expert Systems with Applications, 25*(1), 101–111. https://doi.org/10.1016/s0957-4174(03)00010-1.

Liao, T. W. (2009). Improving the accuracy of computer-aided radiographic weld inspection by feature selection. *NDT and E International, 42*(4), 229–239. https://doi.org/10.1016/j.ndteint.2008.11.002.

Mai, G., Cao, K., Yuen, P. C., & Jain, A. K. (2017). Face image reconstruction from deep templates. *IEEE Transactions on Pattern Analysis & Machine Intelligence*. https://doi.org/10.1109/tpami.2018.2827389.

Mu, W., Gao, J., Jiang, H., Wang, Z., Chen, F., & Dang, C. (2013). Automatic classification approach to weld defects based on PCA and SVM. *Insight-Non-Destructive Testing and Condition Monitoring, 55*(10), 535–539. https://doi.org/10.1784/insi.2012.55.10.535.

Nacereddine, N., Goumeidane, A. B., & Ziou, D. (2019). Unsupervised weld defect classification in radiographic images using multivariate generalized Gaussian mixture model with exact computation of mean and shape parameters. *Computers in Industry, 108,* 132–149. https://doi.org/10.1016/j.compind.2019.02.010.

Richardson, F., Reynolds, D., & Dehak, N. (2015). A unified deep neural network for speaker and language recognition. In *Conference*

*of the international speech communication association* (Vol. 1–5, pp. 1146–1150).

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks, 61,* 85–117. https://doi.org/10.1016/j.neunet.2014.09.003.

Shen, Q., Gao, J., & Li, C. (2010). Automatic classification of weld defects in radiographic images. *Insight-Non-Destructive Testing and Condition Monitoring, 52*(3), 134–139. https://doi.org/10.1784/insi.2010.52.3.134.

Vilar, R., Zapata, J., & Ruiz, R. (2009). An automatic system of classification of weld defects in radiographic images. *NDT and E International, 42*(5), 467–476. https://doi.org/10.1016/j.ndteint.2009.02.004.

Wang, Y., & Guo, H. (2014). Weld defect detection of X-ray images based on support vector machine. *IETE Technical Review, 31*(2), 137–142. https://doi.org/10.1080/02564602.2014.892739.

Wang, L., Lu, H., Ruan, X., & Yang, M. (2015). Deep networks for saliency detection via local estimation and global search. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3183–3192). https://doi.org/10.1109/cvpr.2015.7298938.

Yao, Z., Li, J., Guan, Z., Ye, Y., & Chen, Y. (2020). Liver disease screening based on densely connected deep neural networks. *Neural Networks, 123,* 299–304. https://doi.org/10.1016/j.neunet.2019.11.005.

Yuan, X., Huang, B., Wang, Y., Yang, C., & Gui, W. (2018). Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE. *IEEE Transactions on Industrial Informatics, 14*(7), 3235–3243. https://doi.org/10.1109/TII.2018.2809730.

Zapata, J., Vilar, R., & Ruiz, R. (2012). Automatic inspection system of welding radiographic images based on ANN under a regularisation process. *Journal of Nondestructive Evaluation, 31*(1), 34–45. https://doi.org/10.1007/s10921-011-0118-4.

Zhang, P., Wang, D., Lu, H., Wang, H., & Ruan, X. (2017). Amulet: Aggregating multi-level convolutional features for salient object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 202–211). https://doi.org/10.1109/iccv.2017.31.

Zhao, R., Ouyang, W., Li, H., & Wang, X. (2015). Saliency detection by multi-context deep learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1265–1274). https://doi.org/10.1109/cvpr.2015.7298731.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.