ORIGINAL PAPER

Deep learning-based weld defect classification using VGG16 transfer learning adaptive fine-tuning

Samuel Kumaresan[1] · K. S. Jai Aultrin[2] · S. S. Kumar[3] · M. Dev Anand[4]

Abstract

Welding is a vital joining process; however, occurrences of weld defects often degrade the quality of the welded part. The risk of occurrence of a variety of defects has led to the development of advanced weld defects detection systems such as automated weld defects detection and classification. The present work is a novel approach that proposes and investigates a unique image-centered method based on a deep learning model trained by a small X-ray image dataset. A data augmentation method able to process images on the go was used to offset the limitation of the small X-ray dataset. Fine-tuned transfer learning techniques were used to train two convolutional neural network based architectures with VGG16 and ResNet50 as the base models for the augmented sets. Out of the networks we fine-tuned, VGG16 based model performed well with a relatively higher average accuracy of 90%. Even though the small dataset was spread across 15 different classes in an unbalanced way, the learning curves showed acceptable model generalization characteristics.

✉ Samuel Kumaresan
   sam.profl@gmail.com

   K. S. Jai Aultrin
   jaiaultrin@niuniv.com

   S. S. Kumar
   kumarss@niuniv.com

   M. Dev Anand
   ananddev254@gmail.com

1 Department of Aerospace Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Kanyakumari District, Tamil Nadu 629180, India

2 Department of Marine Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Kanyakumari District, Tamil Nadu 629180, India

3 Department of Electronics and Instrumentation Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Kanyakumari District, Tamil Nadu 629180, India

4 Department of Mechanical Engineering, Noorul Islam Centre for Higher Education, Kumaracoil, Kanyakumari District, Tamil Nadu 629180, India

1 Introduction

Welding is an integrated industrial process that is inherent to most of the production sectors; specialized welding processes are widely used in high-performance industries such as aerospace, design of machines, power generation sectors, and marine and automotive industries. Welding is complex by nature, involving multiple parameters that influence the quality of the weld joint. Apart from welding parameters, random events that take place in the manufacturing processes also lead to weld defects. Thus, quality assurance and testing are crucial for critical applications. As a result, research into detecting and identifying weld defects has become a necessity. Weld defects are usually detected by non-destructive techniques, which are preferred due to their passive nature when physically involving the specimen. Popular non-destructive techniques are vibration analysis, screening by eddy currents, and X-ray identification [1, 2]. These techniques have their limitations. The X-ray detection method results in harmful side effects for humans under prolonged exposure; using eddy currents is limited only to testing metallic specimens. However, image processing has been an important topic of AI technology in the recent decade and has been extensively implemented in weld fault

detection. Image data acquisition, image pre-processing, extraction of features, and classification are the components of image processing [3]. Previously, weld feature extraction and classification methods relied on either surface or geometrical characteristics from weld defect imagery. However, this was inadequate to determine acceptable weld faults. Many academics have suggested various fault detection models that incorporate neural networks. Many types of research have been conducted on the extraction of features and categorization of weld faults. To categorize distinct weld flaws based on textural and geometric characteristics, Valavanis and Kosmopoulos [4] used an artificial neural network and 'k' nearest neighbor methods. Yin et al. [5] formulated a novel approach to extract geometric characteristics from Lissajous imagery and error diagnostics by adapting machine learning methods. Boaretto and Centeno [6] employed the double wall double image (DWDI) illumination approach to determine defects by constantly identifying them by the multilayer perception method. An evolutionary platform-based fuzzy interference method for determining weld defects by establishing the form and defect location as classifying variables was explored by Pérez et al. [7]. To detect unique flaws, Zahran et al. [8] used ANN to compare characteristics obtained by analyzing the density of the power spectrum in the split weld regions. El-Abbasy et al. [9] in their work on using artificial neural network models to forecast the conditions of marine pipelines carrying oil and gas. They developed the models by ANN and highlighted that several existing models focus only on corrosion models and their disadvantages in detecting other conditions of the specimen. The authors developed the models based on the pipeline data acquired from existing offshore pipelines, and the model achieved 97% validity when applied to the dataset. Jiang et al. [10] and Mu et al. [11] in their work explored methods of efficiently converting weld defect imagery to confirmatory factor regions. Carvalho et al. [12] worked on detecting and categorizing the weld defects on pipes by utilizing magnetic flux leak signals and artificial neural networks (ANNs). Pattern recognition was developed from an intelligent animal, and ANN was used to differentiate the weld defects and non-defect weld patterns. The authors utilized Fourier, Wavelet analyses, Moving-average, and Savitzky-Golay filters. They utilized signal processing technology to enhance the performance of the differentiation. They determined the external corrosion and internal corrosion rates with 71% success. Shen et al. [13] in their work on the automatic categorizing of weld defects reported two problems that impeded categorization. They were poor generalizations of the categorization method and poor separation of features, both caused by limited training samples. They proposed a novel method based on a multiclass support machine to determine the weld defects to offset poor generalization and a feature-based image classifier to characterize weld defects; thereby effectively enhancing the separability of the

image group. They reported a significant computation time cut of 22 s and a greater efficiency in defect classification. He et al. [14] investigated the correlation between the floating potential and welding flaws by using finite element modeling. They developed an ultrasonic duplex finite element model to detect weld flaws in a butt-weld joint. A floating potential was used to conveniently describe the geometrical parameters of the weld flaw. Their simulation analysis emulated an ultrasonic-guided wave testing procedure to test the flaws and identify the variant geometrical traits. They argued that the floating potential data exhibited a significant advantage in relating the geometrical features of the weld defects such as their location and orientation against the energy vanishing point and decay rate. They argued that their proposed methodology showed promise in highlighting the weld flaw geometries in practical use cases. When Krizhevshy et al. [15] utilized AlexNet to win the ImageNet image classification challenge, deep learning exhibited unimaginable potential. The ZFNet, VGGNet, GoogLeNet, and ResNet systems for large-scale image categorization were suggested by the authors. Unlike typical feature extraction approaches, these methods do not require any pre-selected picture characteristics to be implemented; they can use supervised learning to learn the operator's high-level characteristics from the test data [16]. Deep learning algorithms have recently become popular for detecting weld flaws. Pattnaik et al. [17] in their work on developing a framework for categorizing pests in a tomato farm. They reported that recent studies had focused on utilizing CNNs and the myriad methods of adapting them to specific processes. The authors reported that the use of CNN's has resulted in significant improvement in image recognition quality. They proposed a transfer learning-based CNN framework that was pre-trained for classifying the tomato plants on a farm. The authors reported that the use of transfer learning-based CNN had resulted in significant improvement in classification accuracy and obtained results with 88% accuracy through the use of the DenseNet169 model. They reported that the effectiveness of transfer learning deep CNN had demonstrated good effectiveness in the detection of pests and classification of plants. Using the notion of learning algorithms, Hou et al. [18] proposed an automatic detection scheme with three-stage detection for weld defects in X-ray imagery. The images were pre-processed to determine the weld area, and a classification model incorporating cropped X-ray images was used for training. A sliding window approach was used in distinguishing the images on the training model. They reported that their proposal was effective in detecting welding quality.

Park et al. [19] employed transfer learning to increase the accuracy of weld flaw classification in radiographic pictures. They applied a pre-trained CNN to a limited collection of weld fault photos and fine-tuned it. They obtained 96.9% classification accuracy with insufficient training data,

illustrating the efficiency of transfer learning in enhancing classification accuracy. Du et al. [20] identified weld defects from X-ray images of aluminium castings of automobile parts by ROI alignment and pyramid network methods. To solve the classification challenges, the aforesaid methods frequently necessitate the creation of deep CNN architectures and the application of SoftMax. Yet, when there were fewer training samples, SoftMax performed poorly and the features were unclear. A smart recognition technique based on deep learning was presented to overcome the difficulties stated. Furthermore, the random forest was considered the best classification method since it had minimal generalization errors when interacting with classification problems [21]. Base models such as VGG net-16 and ResNet50 were used in the augmented image datasets to perform fine-tune-based transfer learning techniques. Park et al. [22] presented a deep CNN-based ensemble learning system for the automatic classification of weld flaws in radiographic pictures. They attained 96.6% classification accuracy, illustrating ensemble learning's usefulness in enhancing classification accuracy. Dubey and Jain [23] in their work on programmed facial recognition modeling, utilized the VGG16-based transfer learning technique. They reported that the complex nature of recognizing myriad human facial features and expressions had immense implications in multiple fields, such as security, behavioral, and medical sciences, for any community. They proposed the use of a framework built on deep learning by using a transfer learning technique to recognize facial expressions. They utilized existing ImageNet-trained models, VGG116 models, and modified them with additional layers. They subsequently tested the model on existing facial datasets and found around 94% accuracy and superior performance over existing facial recognition techniques. They also reported the ease of processing the technique by using Google Colab-GPU and the versatility of VGG16 and transfer learning techniques. To identify and categorize weld faults in ultrasonic pictures, Lee et al. [24] employed a deep CNN. They attained a classification accuracy of 97.5%, proving the use of deep learning algorithms in ultrasonic testing for weld fault identification and classification. Based on graph theory and CNNs, Zhao et al. [25] suggested a new approach for automated weld flaw classification. They obtained 98.1% classification accuracy, highlighting the promise of graph theory-based approaches for enhancing weld flaw classification accuracy. To diagnose weld faults in radiographic pictures, Oh et al. [26] employed a deep learning model that incorporated a CNN and a support vector machine (SVM). They demonstrated that their model could achieve excellent classification accuracy even with minimal training data, implying that deep learning approaches might be useful for weld fault recognition when data is scarce. Joshi et al. [27] explored the usage of deep learning methods in identifying Covid-19-affected patients from X-ray radiographs. They

analyzed datasets involving chest radiographs of affected patients to generate classification data. They also proposed three approaches for the problem namely, a CNN model with a customized architecture and two approaches based on transfer learning using DenseNet121 and MobileNetV2 architectures respectively. They observed improved performance of the three models in the experimental investigations involving available datasets. In comparison to the existing detection methods, they noted improved performance in determining the mean absolute error.

Thus, from analyzing existing research, it has been identified that artificial neural networks and deep learning techniques have largely been used in categorizing and predicting defects and issues. Studies utilizing deep learning, CNN, and transfer learning techniques have exhibited unique advantages based on variant training methods. From the literature, it was inferred that the use of transfer learning in problems requiring categorizing and detection of material problems and defects presents an interesting take on defect identification and the opportunities they present in computation analysis and preventive solutions. The objective of the present research is to identify weld faults utilizing an image-based identification method based on the VGG16 neural network and transfer learning using X-ray radiograph datasets from welded samples retrieved from ImageNet. The base models used for the experimental investigation were ResNet50 and VGG16. The innovative aspect of this study is the use of adaptive approaches to fine-tune the VGG16 to achieve greater average accuracy levels.

2 Materials and methods

2.1 Selection of neural network model

In the present work, Keras, an open-source deep learning library, was utilized for the development of models. The open-source library was used because of its capability to be fine-tuned by transfer learning to better serve the objectives of the present work. The library contained an array of pre-trained deep learning classifiers including VGG16, DenseNet121, InceptionV3, and ResNet50. Each classifier model had different depth values, indicating the need for more resource-intensive classifiers for larger training requirements. In the present work, the availability of data is relatively low in comparison to existing works, and thus the use of a shallow depth model was considered optimal. This was based on the consideration that the subsequent model would exhibit optimal convergence while eliminating the overfitting of the model. Therefore, in the present work, both VGG16 and ResNet50 were the preferred models. The Regions of Interest (ROI) were cropped from the images in the GDXray dataset, and the cropped defect regions were used to train
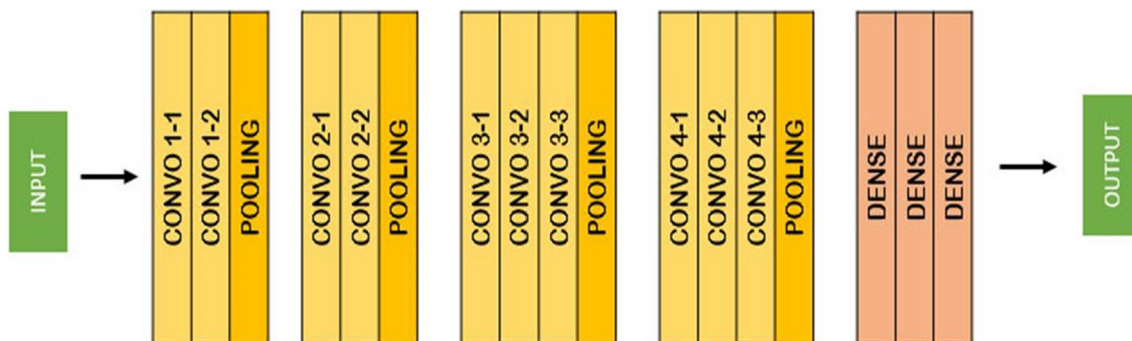
Fig. 1 VGG Net Architecture

Table 1 Features of VGG Architecture

Input	VGG accepts a 224 × 224 pixel RGB picture as input. To keep the input picture size equal for the ImageNet event, the authors reshaped the middle 224 × 224 patch in each image
CNN layers	The receptive field of VGG's convolutional layers is 3 × 3, which is fairly small. Before running the input via a ReLU unit, there are extra 1 × 1 convolution filters that conduct a linear change. To maintain spatial resolution after convolution, the convolution stride is set to 1 pixel
Fully-connected layer	VGG is made up of three entirely linked layers: 4096 channels making up the first and second, with the third layer having 1000 channels
Hidden layers	ReLU is used by hidden layers of VGG. Local Response Normalization (LRN) is not used by VGG since it increases memory usage and training time without improving accuracy

the models and test them. The GDXray dataset used in the present work is hosted in the following repository: http://dmery.ing.puc.cl/index.php/material/gdxray/.

2.2 VGG architecture

AlexNet had utilized ImageNet large scale for the visual recognition challenge when it was released, demonstrating it was one of the most capable in classifying images. Its major features include the use of ReLU, multi-GPU optimization, and overlapping pooling. ImageNet, a resource with over 15 million pictures annotated with over 22,000 classes, served as a general benchmark for image recognition in the present work. Even though AlexNet had all the well-developed features, the implementation of VGG Architecture was quite confusing in the beginning stage. However, the advanced features and time-consumption techniques of VGG architecture made it preferable for research. In Fig. 1, the clear architecture of the VGG network has been explained.

While earlier AlexNet derivatives concentrated on lower feature maps and leaps in the first convolution kernel, VGG tackles a crucial element which is the depth of CNN [15]. The advanced features of VGG Architecture in comparison to existing systems are listed in Table 1.

2.3 Transfer learning

Machine learning and deep learning are broad topics with many applications, including image recognition, speech recognition, recommendation and association systems, and so on. To create any model from the ground up, large storage and computational power are required that won't always be available. We may also encounter instances where we know strategies for improving current models, but the difficulties of retraining the models from scratch prevent us from doing so. The transfer learning technique is applied to address such problems. Instead of constructing models from the ground up, this strategy is used to pre-train models as the starting point. This enabled us to meet the challenge of developing deep learning models, which require a considerable amount of computing and storage resources. Equations (1) and (2) represent the component domains and marginal distribution, respectively. The formal description of transfer learning and its application is:

Two elements tuple with domain D, X_i as features space and $P(X)$ as a marginal probability, and sample data points as X,

For two component domains

$$D = \{X_i, P(X)\}. \tag{1}$$
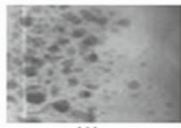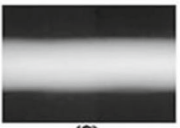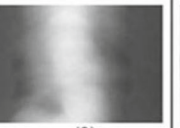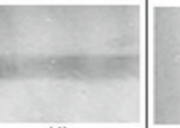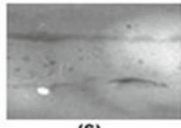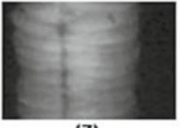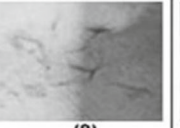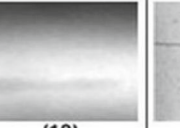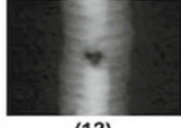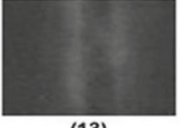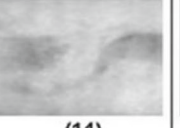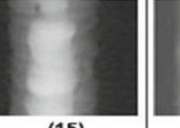
Sample Image					
	(1)	(2)	(3)	(4)	(5)
Defect Type	Cluster Porosity	No Defect	Cap Undercut	Lack of Root Penetration	Longitudinal Crack
No of Sample	64	116	96	80	167
Sample Image					
	(6)	(7)	(8)	(10)	(11)
Defect Type	Parallel Slag lines	Root Undercut	Silica Inclusions	Slag line	Transverse Crack
No of Sample	65	60	41	41	145
Sample Image					
	(12)	(13)	(14)	(15)	(16)
Defect Type	Burn through	Concave root	Concave root intermittent	Excessive root penetration	Lack of root fusion
No of Sample	39	36	39	31	36

Fig. 2 Defect type, number of images in each class, and a sample image of each class

For marginal distribution

$$P(X), X=\{x_1\ldots\ldots x_n\}, x_i \in X_i, \tag{2}$$

where x_i denotes a specific vector.

Task T denotes the twin element tuple with a label space γ and the objective function η. From a probabilistic standpoint, the objective function can be written as $P(\gamma|X)$.

Using domain D, and by defining the task by two components, Eqs. (3) and (4) are defined to be (adapted from [28]).,

$$T = \{\gamma, P(Y|X)\} = \{\gamma, \eta\}, \tag{3}$$

$$Y = \{y_1, \ldots y_n\}, y_i \epsilon \gamma. \tag{4}$$

where γ is the label space, η is the predictive function, $x_i \in X, y_i \in Y$ are the feature vectors for the two-component pairs, and $\eta(x_i) = y_i$ is the prediction of the correspondence label pair.

Transfer learning enabled the determination of targeted conditional probability distribution $P(YT \mid XT)$ with data obtained from DS, TS with DS, and TS corresponding to DT and TT respectively. As seen in the document categorization example, the source and destination situations differ on the following means,

- $X_{iS} \neq X_{iT}$. The source and destination domains have separate feature spaces.
- $P(X_S) \neq P(X_T)$. The marginal outcome and its probability from the source and target domain distributions diverge.
- $\gamma_S \neq \gamma_T$. The two tasks possess varying label spaces.
- $P(Y_S|X_S) \neq P(Y_T|X_T)$. The source task and target task retain varying conditional probability distributions.

2.4 Dataset description

The GRIMA X-ray database was utilized in creating the weld X-ray database used in this investigation. The X-ray radiographs were acquired based on the techniques specified by the ISO 17636–1 standard for radiographic examination of metallic fusion weld joints. A Lumisys LS85 SDR scanner was used to digitize the radiography films. With a linear LUT proportional to optical film density, 8-bit data were rescaled from the original 12-bit data. The radiographs have a pixel size of 630 DPI in TIFF format. The square regions of interest that contained the defects were manually separated from full-scale radiographs during image annotation. The experimental weld defect sample images are illustrated in Fig. 2. The VGG16 and ResNet50 are pre-trained models with 224 × 224 pixels image sizes. The resulting images were subjected to cropping at a 1:1 ratio to avoid distortions and were
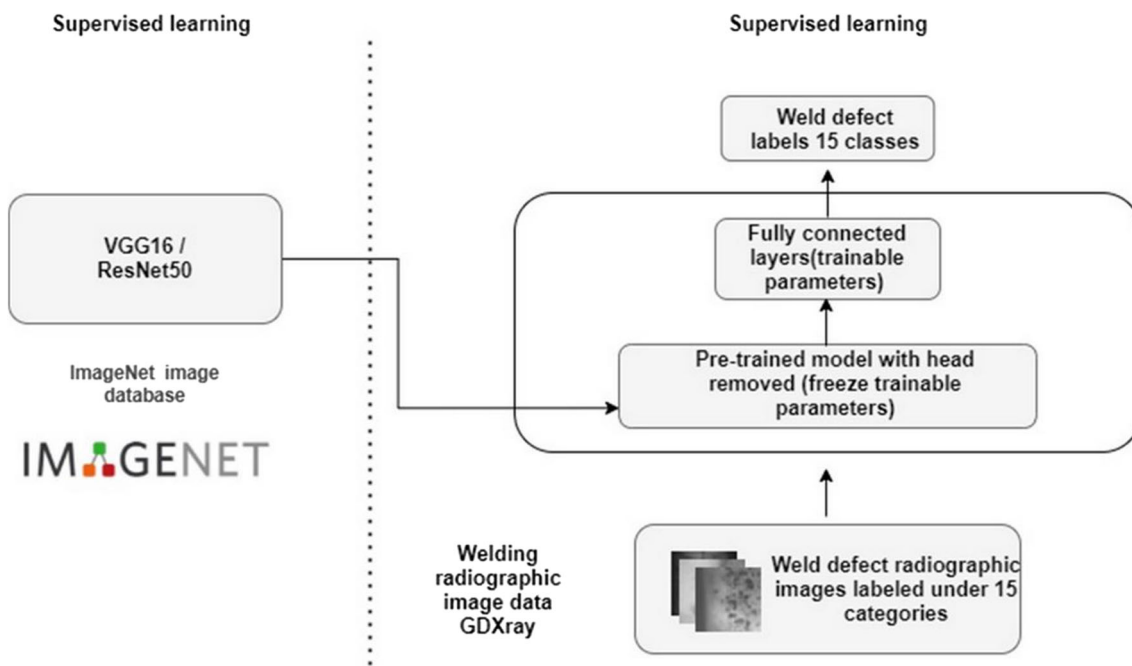
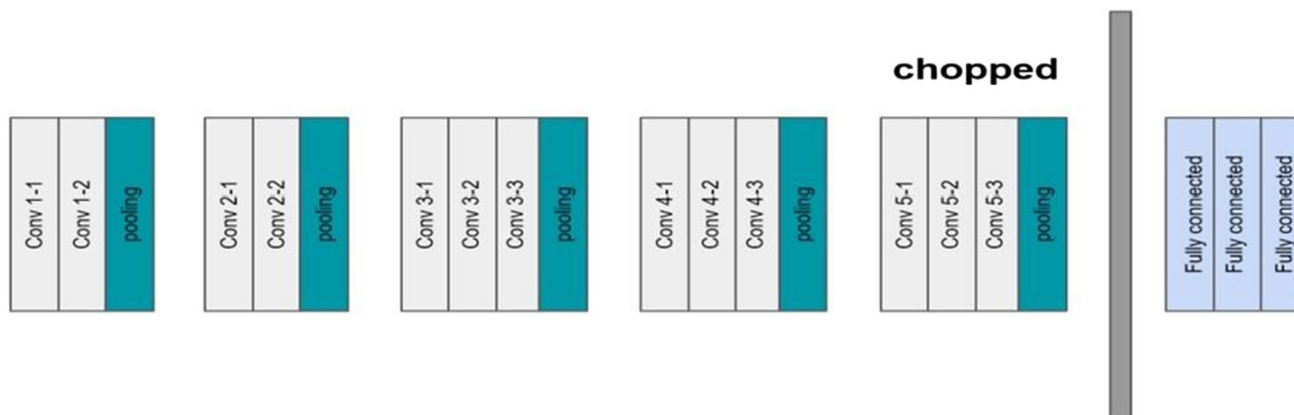**Fig. 3** Workflow of weld defect classifier training

Fig. 4 The head of the pretrained VGG16 model got chopped off and replaced with a randomly initialized head

categorized into 14 weld defect classes. The number of image samples for different classes revealed significant skewing in the distribution. The performance of the classifier is subjected to being affected by this skewing when trained with this data. The performance of the classifier is also affected by minority classes.

2.5 Training and compiling

As an initial step to prepare the image for our convolutional neural network, the data was loaded and preprocessed by converting it to RGB channel ordering and resizing it to 224*224 pixels. The pixel intensities were scaled to the range of 0 to 1. Data augmentation was done by generating new data from the available training data by applying random geometrical

transforms such as zoom, shift, flip, and more in different combinations to expand the data available for training with new plausible samples. It ensured better generalization of the present model and avoided overfitting. The labels are categorical features that cannot be applied to the machine learning models as it is, so they were transformed using one hot encoder (Fig. 3). The data has been split into 80% for training and 20% for testing.

In our previous work [29], a trained convolutional neural network was utilized as a feature extractor for transfer learning and it was used to propagate the image dataset and extract the activations at each layer, and subsequently, the features were saved. A standard machine learning classifier system was trained alongside CNN. In the present work, a fine-tuning technique was implemented where a pre-trained
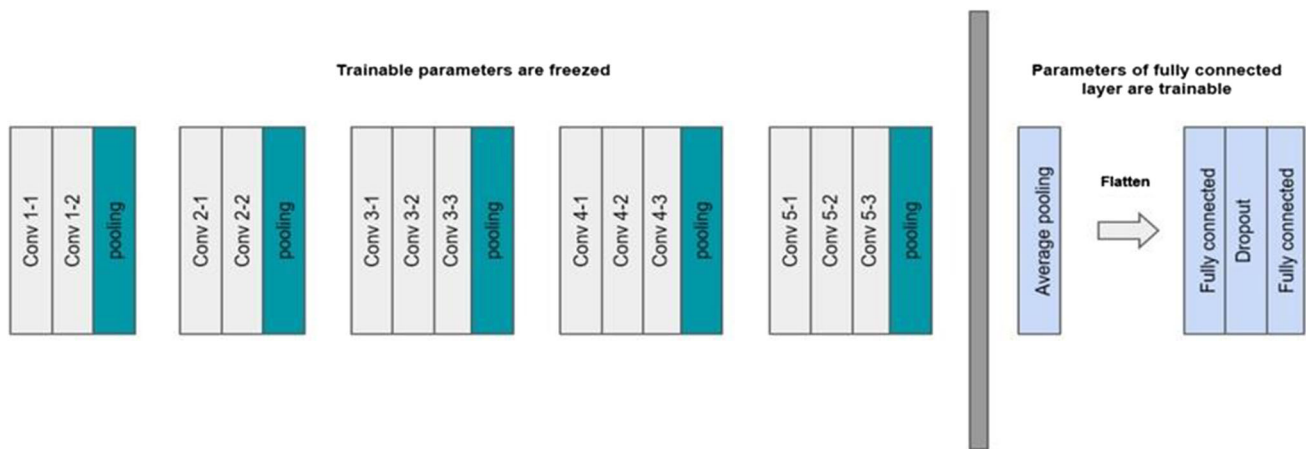
Fig. 5 The new fully connected layer was appended on top of the VGG16

CNN model (VGG16/ResNet50) was used without the final connected layers. The 'head' of the network, where the class label predictions are returned to, is replaced by a new fully connected layer set possessing randomly initialized trainable parameters (Fig. 4). The construction of the head includes an average pooling layer of pooling size (4, 4), a completely connected 64-node layer and relu activation, a final complete layer with Softmax activation, and a dropout layer whose number of nodes is the same as several classes need to be classified (Fig. 5).

The proposed models were trained in the Google Colab coding environment using available free resources, namely the Nvidia K80s, P4s, T4s, and P100s GPUs. To develop the inference and training scripts, the Python SDK was used. To implement the deep learning models, the open-source Keras library was used, as it is a high-level framework for deep learning with TensorFlow and Theano as the base. The pretrained weights for the present VGG16 and ResNet50 were imported from the Keras library. For training and inference, open-source libraries such as Numpy, Seaborn, and Scikit learn were used. The tensors were handled by Numpy and Scikit learn, and the character plotting was handled by Seaborn. The layers beneath the head were frozen, preventing adjustments in weight. The network was trained so that the newer, fully connected layer sets were taught the previous CONV pattern levels formerly used in the network. The training was continued by making the rest of the network unfrozen and using recognition of classes in pretrained networks that were not initially trained by fine-tuning. The final trained models were categorized into pickle files. The training was initiated with the following non-trainable parameters.

- Dropout layer is used for the newly built head of VGG16 for enhanced model generalization. The corresponding fraction of input units to drop is set to 0.5.
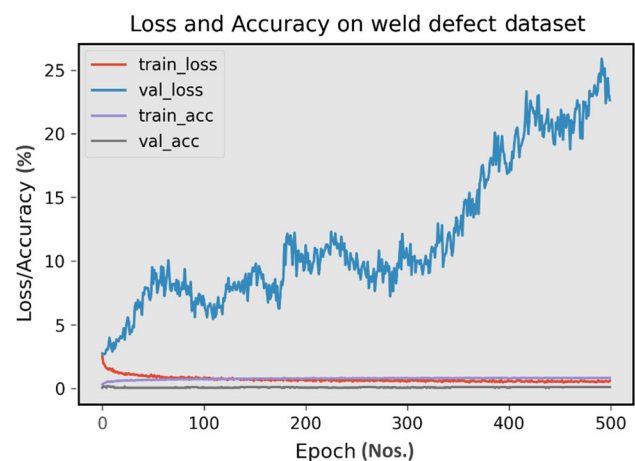
Fig. 6 Train learning curve and validation learning curve of ResNet50 model fine-tuning

- Two fully connected layers are present with either 64 or 15 nodes.
- Training of the network is done by Adam optimizer with a pre-scheduled learning rate. The initial learning rate is 1×10^{-3} with the decay of the initial learning rate/number of epochs. The initial high learning rate followed by slow decay is used to eliminate early overfitting.
- The batch size of 16 is used. A categorical cross-entropy loss function was used.

3 Results and discussions

3.1 Experimentation and result

Google Colab web IDE was utilized for the experiments. This IDE provides a free GPU instance. The open source

(a) Loss and Accuracy on weld defect dataset

(b) Loss and Accuracy on weld defect dataset

(c) Loss and Accuracy on weld defect dataset

Fig. 7 Train learning and validation curve of VGG16 model fine tuning **a** VGG16 model fine-tuning up to 100 epochs; **b** VGG16 model fine-tuning up to 400 epochs **c** VGG16 model fine-tuning up to 700 epochs
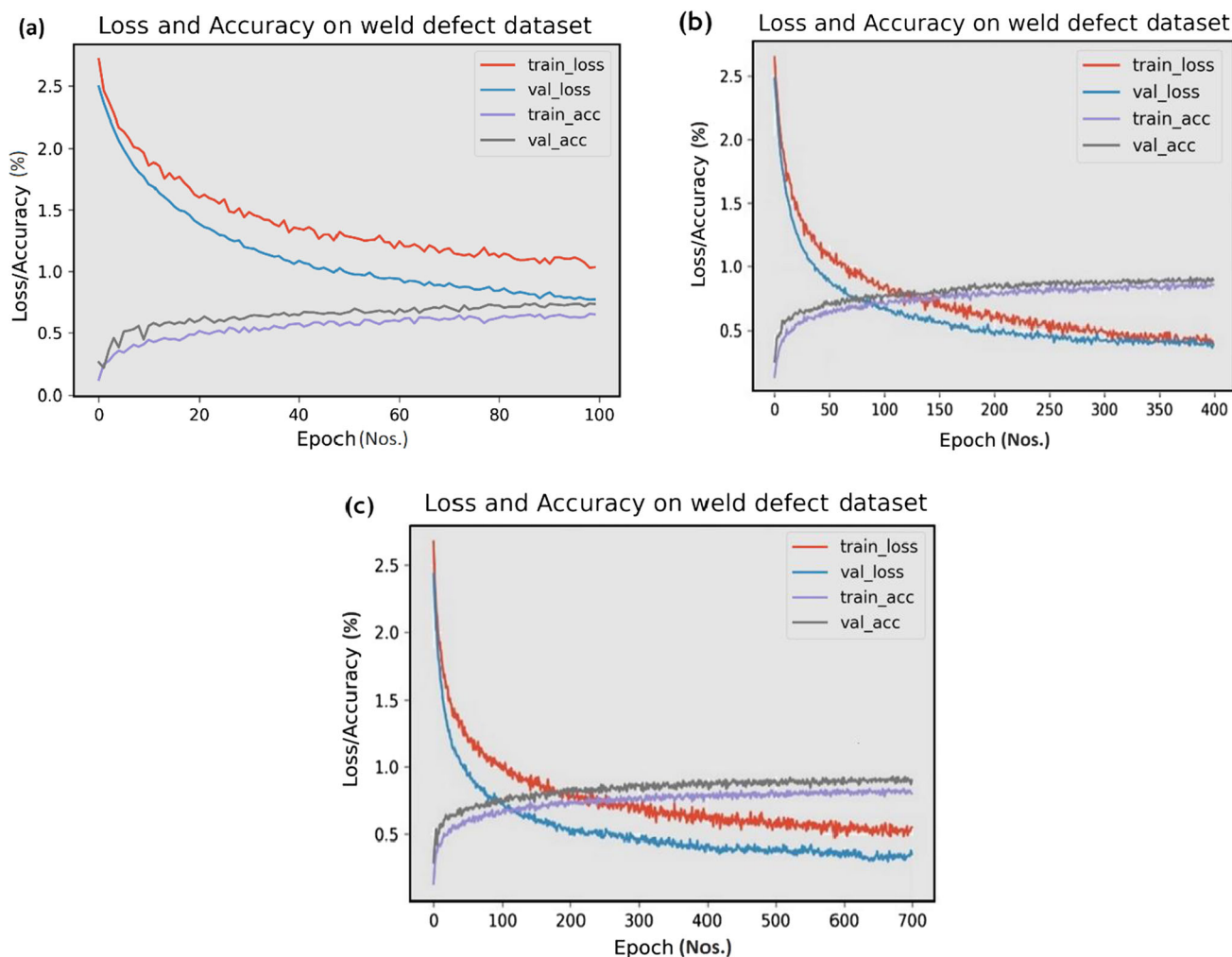
library, Keras, running on the Tensorflow backend, was used for the implementation. The pretrained models and their weights were imported from Keras itself. The experimentation was done using two different base models VGG16 (138 million trainable parameters) and ResNet50 (23 million trainable parameters). During the training process, the model's learning performance was monitored by using the learning curves. During training, the learning curves were used to identify overfitting or underfitting problems in the model learning so that the generalization behavior could be diagnosed. In the learning curves, the training curve showed a favorable model learning rate, and the generalization of the model was illustrated by the validation curve. The learning curves are given below. The learning curve of the model using ResNet50 as the base showed that the model did not get generalized. While training the model for the pre-determined epochs, the validation of the model was not generalized, even though the training learning curve indicated good learning

progress for the model. Increases in the validation loss and corresponding drops in the validation accuracy were indications of poor generalization. However, the training learning curve appeared satisfactory with increasing training accuracy (performance learning curve) and decreasing training loss (optimization learning curve), as shown in Fig. 6.

The learning curves of the model using VGG16 as the base showed relatively better learning and generalization characteristics. While looking at the performance learning curves, the validation accuracy was higher than the training accuracy, which indicated the model was generalized and could perform well using unseen data. The characteristics illustrated the model's improvement with an increasing number of epochs (Fig. 7).

From the figure, it is inferred that balanced learning happened along the network with an increasing number of epochs. For better inference, the validation of the model confusion matrix is plotted for the epochs 100, 400, and 700 in
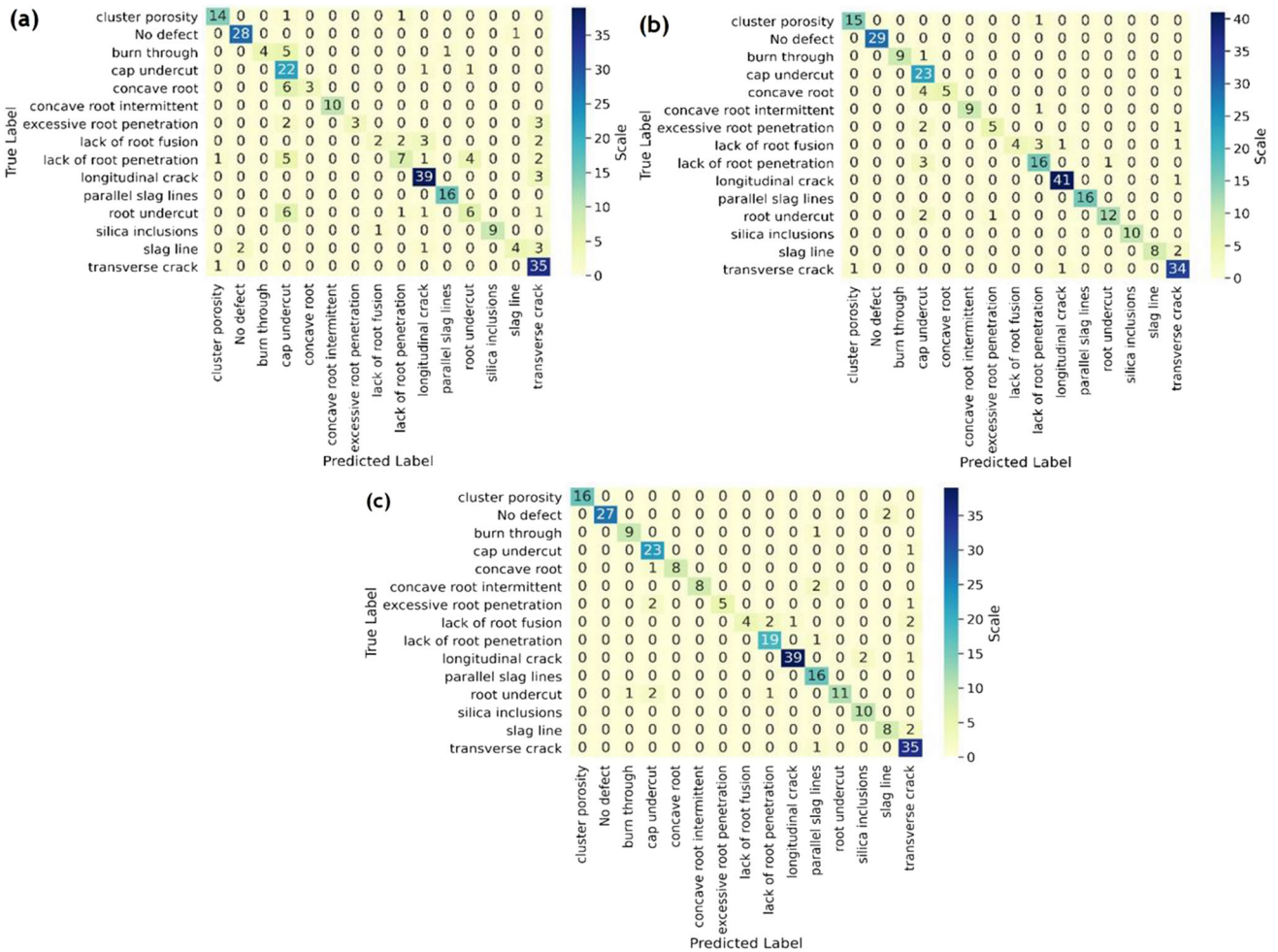
Fig. 8 Confusion matrix of VGG16 model fine-tuned on weld defect data for epochs: counts of classified images along diagonal. **a** VGG16 model confusion matrix fine-tuned for weld defect data for 100 epochs; **b** VGG16 model confusion matrix fine-tuned for weld defect data for 400 epochs; **c** confusion matrix of VGG16 model fine-tuned on weld defect data for 700 epochs

Fig. 8. From Fig. 7, it is inferred that there is an increase in the prediction of some classes in each epoch, which enhances the performance of the model with better identification of models without any misclassified predictions. As per the experimentation, it was indicated that although the welding dataset taken for this experiment was smaller having only 15 classes, the fine-tuned transfer learning approach provided a better model performance. The performance metrics are listed in Table 2.

In the present work, the defect regions were manually cropped, trained, and tested. The deep learning models that were used had input sizes of 224 × 224x3 (for both VGG16 and ResNet50) with a 3-channel image of 224 × 224 pixel size. The performance metrics of the proposed model in the present work are shown in Table 2. The recall, F1 score, precision, and accuracy were the performance metrics utilized to evaluate the classifier models. These obtained performance metrics were predicted by the model using the validated data,

and the performance of the classifier model was indicated. The proposed model mentioned in the support column of Table 2 was tested on a set of 264 images (out of a total of 1320).

3.2 Comparative study

The obtained results from the present work were compared with the existing work in this section. A VGG net-based model was trained on a small welding X-ray image dataset with three classes. The existing work used data augmentation techniques to expand their training data and attained average accuracy of 95%. A dataset of 3 classes, including 57 lack of penetration images, 44 pinhole images, and 115 porosity images was used for training [18]. Image smoothing, segmentation, and feature extractions were done; subsequently, the feature vectors were used to train a multiclass support vector machine (MSVM), capable of classifying 6 different

Table 2 Evaluation reports of classifiers fine-tuned on weld defect data of 15 classes, using VGG16 as the base model

Epochs	200			400			700			
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score	Support
Cluster porosity	0.79	0.94	0.86	0.94	0.94	0.94	1.00	1.00	1.00	16
No defect	0.94	1.00	0.97	1.00	1.00	1.00	1.00	0.93	0.96	29
burn through	1.00	0.60	0.75	1.00	0.90	0.95	0.90	0.90	0.90	10
Cap undercut	0.61	0.96	0.74	0.66	0.96	0.78	0.82	0.96	0.88	24
Concave root	0.83	0.56	0.67	1.00	0.56	0.71	1.00	0.89	0.94	9
Concave root intermittent	1.00	0.62	0.77	1.00	0.90	0.95	1.00	0.80	0.89	10
Excessive root penetration	1.00	0.33	0.50	0.83	0.62	0.71	1.00	0.62	0.77	8
Lack of root fusion	1.00	0.33	0.50	1.00	0.44	0.62	1.00	0.44	0.62	9
Lack of root penetration	0.74	.070	0.72	0.76	0.80	0.78	0.86	0.95	0.90	20
Longitudinal crack	0.89	0.98	0.93	0.95	0.98	0.96	0.97	0.93	0.95	42
Parallel slag lines	0.89	1.00	0.94	1.00	1.00	1.00	0.76	1.00	0.86	16
Root undercut	0.88	0.47	0.61	0.92	0.80	0.86	1.00	0.73	0.85	15
Silica inclusions	1.00	1.00	1.00	1.00	1.00	1.00	0.83	1.00	0.91	10
Slag line	1.00	0.60	0.75	1.00	0.80	0.89	0.80	0.80	0.80	10
Transverse crack	0.87	0.94	0.91	0.85	0.94	0.89	0.83	0.97	0.90	36
Accuracy			0.85			0.89			0.90	264
Macro average	0.90	0.78	0.81	0.93	0.84	0.87	0.92	0.86	0.88	264
Weighted average	0.87	0.85	0.84	0.91	0.89	0.89	0.91	0.90	0.90	264

Table 3 Comparative study

Author	Identifying capability	Accuracy	Method
Ajmi et al. [32]	3 defect types	95%	VGG-based CNN architecture with data augmentation
Malarvel and Singh [30]	6 defect types	96%	Multi-class support vector machine (MSVM)
Liu et al. [33]	3 defect types	97.60%	VGG16 transfer learning
Hou et al. [18]	4 defect types	97.20%	DCNN with 4 convolution layers
Yang and Jiang [34]	5 defect types	91.36%	DNN pretrained by SAE
Kumaresan et al. [29]	14 defect types	98%	SVM trained on features extracted using pretrained ResNet50
Present work	14 defect types	90%	Fine-tuning pretrained VGG16 (transfer learning)

types of weld defect [30]. The dataset consisted of 4 classes which included porosity, crack, lack of penetration, and slag inclusion classes with 3503 cropped patches as a whole. The deep feature extractors were compared with traditional feature extractors. As a result, the classifier performed better on deep features and exhibited an accuracy of 97.2% [31].

The deep feature extractor consisted of 4 convolution layers with a single fully connected layer. A multilevel feature for classifying weld defects by a united deep neural network was investigated [32, 33]. There were 220 samples in total, divided into five classes, with 50 samples for slag inclusion, a lack of penetration and porosity, 35 samples for cracking, and 35 samples for fusion. The proposed model reported an

accuracy of 91.36%. In the previous work [29], pre-trained models like ResNet50 and VGG16 with ImageNet weight were used for feature extraction, and machine learning methods such as logistic regression, random forest, and SVM were trained on the extracted features. The SVM model trained by the featured ResNet50 exhibited good results. However, the inference process required two models: a pretrained model for feature extraction and a machine-learning model for classification. A more detailed review of the comparison against the existing and the present work is discussed in Table 3. The existing works listed in Table 3 used similar weld defect radiograph images for training and testing the classifier models. In particular, [18, 30] utilized the same GDXray image dataset in their investigation. The average macro accuracy of the proposed model was compared to the other existing works; the versatility of the proposed model is indicated by the number of weld defects classified by the proposed model in the capable identification column in Table 3. The present work is an improvement in terms of enabling the VGG16 deep neural network in terms of detecting up to 14 types of weld defects which is higher than the capability reported by Ajmi et al. [27] and Liu et al. [33] in their study. The present work utilized ImageNet as the benchmark for achieving 90% accuracy by using the transfer learning technique. In comparison to the deep learning models developed by Hou et al. [18] and Yang and Jiang [34], the present work's Fine-tuned pretrained VGG16 is a more modern and larger network with a proportionally longer parametric training time.

In consideration of the above-mentioned works on weld defect classification, the results obtained in the present study appear promising since it does not require complicated feature extraction processes. The generalization of the model exhibits good results, even with smaller datasets. The performance of the model is well within acceptable limits, even with up to 15 weld defect classes. This indicates the promising capability of this approach, irrespective of data distribution imbalances and limited sample numbers. Furthermore, when the need for model optimization or application scaling up resource management and related processes arises, it will be relatively simple due to the usage of a single model for feature extraction and classification in the present work.

4 Conclusion

In the present work, a deep learning model for weld defect classification was developed. A pre-trained VGG16, with ImageNet weights, was used as a base model, and through transfer learning, the model was fine-tuned so that it could identify and classify weld defects. Even though the training data size was small, the data augmentation technique served the requirements of the process and provided better model generalization.

- The novel VGG16 model automatically extracted deep features from the input image data, eliminating the need for complicated feature extraction processes and providing rich features to the successive layer of the model for classification.
- The head of VGG16 was replaced, and the parameters of the head were trained alone while the remaining layers were frozen, resulting in a reduced number of parameters that needed to be trained and thus lower computational requirements.
- The number of classes was raised to 15, with fewer samples in each class, and the learning curves obtained were satisfactory. The validation curves indicated good model generalization.
- From the results obtained, it was affirmed that the present approach is capable of being adapted for other industrial computer vision classification applications with low training data availability.
- The future scope of the present work can be varied such that it could include: multi-class defect identification, real-time defect identification, automated defect recognition, and integration with other technologies.

References

1. Petcher, P.A., Dixon, S.: Weld defect detection using PPM EMAT generated shear horizontal ultrasound. NDT. E. Int. **74**, 58–65 (2015). https://doi.org/10.1016/j.ndteint.2015.05.005
2. Zou, Y., Du, D., Chang, B., Ji, L., Pan, J.: Automatic weld defect detection method based on Kalman filtering for real-time radiographic inspection of spiral pipe. NDT E. Int. **72**(1), 1–9 (2015). https://doi.org/10.1016/j.ndteint.2015.01.002
3. Dziczkowski, L.: Elimination of Coil Liftoff from eddy current measurements of conductivity. IEEE. Trans. Instrum. Meas. **62**(12), 3301–3307 (2013). https://doi.org/10.1109/tim.2013.2272842
4. Valavanis, I., Kosmopoulos, D.: Multiclass defect detection and classification in weld radiographic images using geometric and texture features. Expert. Syst. Appl. **37**(12), 7606–7614 (2010). https://doi.org/10.1016/j.eswa.2010.04.082
5. Yin, L., Ye, B., Zhang, Z., Tao, Y., Xu, H., Avila, J.R.S., Yin, W.: A novel feature extraction method of eddy current testing for defect detection based on machine learning. NDT E. Int. **107**(1), 102108 (2019). https://doi.org/10.1016/j.ndteint.2019.04.005
6. Boaretto, N., Centeno, T.M.: Automated detection of welding defects in pipelines from radiographic images DWDI. NDT E. Int. **86**(1), 7–13 (2017). https://doi.org/10.1016/j.ndteint.2016.11.003
7. Pérez, J.Z., Rafael, H., Ramón, M.: Performance evaluation of an automatic inspection system of weld defects in radiographic images based on neuro-classifiers. Expert Syst. Appl. **38**(7), 8812–8824 (2011). https://doi.org/10.1016/j.eswa.2011.01.092

8. Zahran, O., Kasban, H., El-Kordy, M., El-Samie, F.E.A.: Automatic weld defect identification from radiographic images. NDT E. Int. **57**(1), 26–35 (2013). https://doi.org/10.1016/j.ndteint.2012.11.005

9. El-Abbasy, M.S., Senouci, A., Zayed, T., Mirahadi, F., Parvizsedghy, L.: Artificial neural network models for predicting condition of offshore oil and gas pipelines. Autom. Constr. **45**(1), 50–65 (2014). https://doi.org/10.1016/j.autcon.2014.05.003

10. Jiang, H.Q., Zhao, Y., Gao, J., Zhao, W.: Weld defect classification based on texture features and principal component analysis. Insight: Non-Destr. Test. Cond. Monit. **58**(4), 194–200 (2016). https://doi.org/10.1784/insi.2016.58.4.194

11. Mu, W., Gao, J., Jiang, H., Zhao, W., Chen, F., Dang, C.: Automatic classification approach to weld defects based on PCA and SVM. Insight: Non-Destr. Test. Cond. Monit. **55**(10), 535–539 (2013). https://doi.org/10.1784/insi.2012.55.10.535

12. Carvalho, A.A., Rebello, J.M.A., Sagrilo, L.V.S., Camerini, C.S., Miranda, I.V.J.: MFL signals and artificial neural networks applied to detection and classification of pipe weld defects. NDT E. Int. **39**(8), 661–667 (2006). https://doi.org/10.1016/j.ndteint.2006.04.003

13. Shen, Q., Gao, J., Li, C.: Automatic classification of weld defects in radiographic images. Insight: Non-destr. Test. Cond. Monit. **52**(3), 134–139 (2010). https://doi.org/10.1784/insi.2010.52.3.134

14. He, D., Chen, J., Chen, Y., Miao, J., Deng, J., Chen, J., Liu, J.: Exploration of the correlation between welding flaw geometrical parameters and floating potential based on an ultrasonic duplex finite element model. Meas. **167**(1), 108269 (2021). https://doi.org/10.1016/j.measurement.2020.108269

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

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

17. Pattnaik, G., Shrivastava, V.K., Parvathi, K.: Transfer learning-based framework for classification of pest in tomato plants. Appl. Artif. Intel. **34**(13), 981–993 (2020). https://doi.org/10.1080/08839514.2020.1792034

18. Hou, W., Wei, Y., Guo, J., Jin, Y., Zhu, C.: Automatic detection of welding defects using deep neural network. J. Phys. Conf. Ser. **2018: 933**, 012006 (2017). https://doi.org/10.1088/1742-6596/933/1/012006

19. Park, S.B., Kim, T.H., Yoon, Y.J.: A transfer learning approach for weld defect classification using convolutional neural networks. Metals. **11**(10), 1555 (2021). https://doi.org/10.3390/met11101555

20. Du, W., Shen, H., Fu, J., Zhang, G., He, Q.: Approaches for improvement of the X-ray image defect detection of automobile casting aluminum parts based on deep learning. NDT E. Int. **107**(1), 102144 (2019). https://doi.org/10.1016/j.ndteint.2019.102144

21. Hirschberg, J., Manning, C.D.: Advances in natural language. Processing. Sci. **349**(6245), 261–266 (2015). https://doi.org/10.1126/science.aaa8685

22. Park, S.C., Jang, J.H., Yoon, Y.J.: Automated classification of weld defects using ensemble learning based on deep convolutional neural networks. J. Mech. Sci. Technol. **35**(8), 3893–3903 (2021). https://doi.org/10.1007/s12206-021-0756-9

23. Dubey, A.K., Jain, V.: Automatic facial recognition using VGG16 based transfer learning model. J. Inf. Opt. Sci. **41**(7), 1589–1596 (2020). https://doi.org/10.1080/02522667.2020.1809126

24. Lee, D., Lee, J.H., Kim, T.H., Yoon, Y.J.: Deep learning-based automated detection and classification of weld defects using ultrasonic testing. Metals. **11**(8), 1369 (2021). https://doi.org/10.3390/met11081369

25. Zhao, Y., Li, J., Li, Y.: Automated weld defect classification based on convolutional neural networks and graph theory. Mater. Letter. **307**, 131257 (2022). https://doi.org/10.1016/j.matlet.2021.131257

26. Oh, Y., Shin, D., Choi, H., Kim, M.: Weld defect classification using deep learning with limited training data. Metals. Mater. Int. **27**(6), 3038–3047 (2021). https://doi.org/10.1007/s12540-021-00975-9

27. Joshi, B., Sharma, A.K., Yadav, N.S., Tiwari, S.: DNN based approach to classify Covid'19 using convolutional neural network and transfer learning. Int. J. Comput. Appl. **44**(10), 907–919 (2021). https://doi.org/10.1080/1206212X.2021.1983289

28. Ruder, S.: Transfer learning - machine learning's next frontier. Ruder IO. http://ruder.io/transfer-learning/ (2017). Accessed June 3 (2022)

29. Kumaresan, S., Aultrin, K.S.J., Kumar, S.S., Anand, M.D.: Transfer learning with CNN for classification of weld defect. IEEE. Access. **9**(1), 95097–95108 (2021). https://doi.org/10.1109/access.2021.3093487

30. Malarvel, M., Singh, H.: An autonomous technique for weld defects detection and classification using multi-class support vector machine in X-radiography image. Optik **231**(1), 166342 (2021). https://doi.org/10.1016/j.ijleo.2021.166342

31. Hou, W.H., Ye, W., Yi, J., Zhu, C.A.: Deep features based on a DCNN model for classifying imbalanced weld flaw types. Meas. **131**(1), 482–489 (2019). https://doi.org/10.1016/j.measurement.2018.09.011

32. Ajmi, C., Zapata, J., Martínez-Álvarez, J.J., Ruiz, R.: Using deep learning for defect classification on a small weld X-ray image dataset. J. Nondestruct. Eval. **39**(1), 68 (2020). https://doi.org/10.1007/s10921-020-00719-9

33. Liu, X., Wang, C., Hu, Y., Zeng, Z., Bai, J., Liao, G.: Transfer learning with convolutional neural network for early gastric cancer classification on magnifiying narrow-band imaging images. In: 25th IEEE Int. Conf. Image Proc. Pp. 1388–1392 (2018). https://doi.org/10.1109/ICIP.2018.8451067

34. Yang, L., Jiang, H.: Weld defect classification in radiographic images using unified deep neural network with multi-level features. J. Intell. Manuf. **32**, 459–469 (2021). https://doi.org/10.1007/s10845-020-01581-2