A COMPARISON OF CNN-BASED IMAGE FEATURE EXTRACTORS FOR WELD DEFECTS CLASSIFICATION

Tito Wahyu Purnomo[1], Harun Al Rasyid Ramadhany[2], Hapsara Hadi Carita Jati[3], Djati Handoko*[1]

[1] Department of Physics, Universitas Indonesia, Kampus UI, Depok, Indonesia
[2] Research Center for Radiation Process Technology, National Research and Innovation Agency, Jl. Lebak Bulus Raya, Jakarta, Indonesia
[3] Polytechnic Institute of Nuclear Technology, National Research and Innovation Agency, Jl. Babarsari, Yogyakarta, Indonesia
* djati.handoko@ui.ac.id

ABSTRACT

Classification of the types of weld defects is one of the stages of evaluating radiographic images, which is essential in controlling the quality of welded joints in materials. By automating the weld defect classification based on deep learning and the CNN architecture, it is possible to overcome the limitations of visually or manually evaluating radiographic images. Good accuracy in classification models for weld defects requires the availability of sufficient datasets. In reality, however, the radiographic image dataset accessible to the public is limited and imbalanced between classes. Consequently, simple image cropping and augmentation techniques are implemented during the data preparation. To construct a weld defect classification model, we proposed to utilize the transfer learning method by employing a pre-trained CNN architecture as a feature extractor, including DenseNet201, InceptionV3, MobileNetV2, NASNetMobile, ResNet50V2, VGG16, VGG19, and Xception, which are linked to a simple classification model based on multilayer perceptron. The test results indicate that the three best classification models were obtained using the DenseNet201 feature extractor with a test accuracy value of 100%, followed by ResNet50V2 and InceptionV3 with an accuracy of 99.17%. These outcomes are better than state-of-the-art classification models with a maximum of six classes of defects. The research findings assist radiography experts in evaluating radiographic images more accurately and efficiently.

Keywords: CNN; Transfer Learning; Feature Extraction; Radiographic Image; Weld Defects

INTRODUCTION

Inspecting welded joints on metal or other materials is essential for ensuring materials' performance, reliability, and safety, particularly in industrial manufacturing applications. Several factors, including labor, equipment, and environmental conditions, can cause defects in welded joints to occur during the welding process. Nondestructive testing (NDT) techniques for detecting weld defects are typically categorized as radiographic, dye penetrant, ultrasonic, eddy current, magnetic particle, etc. Radiographic tests are widely used to inspect welded joints in industry. The inspected material is exposed to ionizing radiation, typically X-rays or gamma rays, so that the detector or film captures the radiation emitted by the material, and a radiographic image representing the internal condition of the

welded joint is created. Radiography experts interpret and evaluate these images as part of the inspection process. At this stage, it is crucial to classify the defects found in the welded joints. The disadvantages of manual interpretation and evaluation of radiographic images include the time-consuming, complex processes, subjective results, inaccuracy, sometimes biased results for similar defects, and the difficulty in classifying the type of defect with a small size. Implementing automatic classification of weld defects using a computer vision system and artificial intelligence algorithms can overcome the insufficiency of manual interpretation.

Recent research developments have focused on automating weld defect classification in radiographic images using CNN-based deep learning algorithms. The availability of radiographic image datasets is a prerequisite for constructing a weld defect classification model. The GDXray database served as the radiographic image database source, consisting of 88 high-quality digitized radiographic images categorized into three series and stored as uncompressed TIFF files with a 40.3 m pixel size[1]. The WDXI datasets contain 13,766 radiographic images of welding defects, including seven significant types of weld defects, but they are not freely available[2]. The RIAWELC dataset contains 24,407 8-bit radiographic images in PNG format with 224 x 224-pixel sizes, which is a further application in the development of the GDXray dataset and includes four classes of weld defect types[3].

Various CNN architectures are employed to develop weld defect classification models. As a method for automatically detecting weld defects, Faster R-CNN is proposed to integrate feature extraction and classification in a single algorithm to automate the detection process[4] fully. A framework for object detection with the AF-RCNN attention mechanism is proposed because image defects comprise numerous minor defects whose feature information is more probable to be lost throughout convolution[5]. In a series of stages for detecting and classifying weld defects, the FgSegNet architecture is used to segment weld areas and defects. In contrast, the EfficientNet architecture is used to classify weld areas and defects[6]. Transfer learning is a deep learning approach that applies pre-trained models to new problems.

As feature extractors, several CNN architectures that have been pre-trained using ImageNet datasets are utilized to employ weld defect classification models, including VGG19[7], AlexNet[8], MobileNet[9], VGG16[10], VGG16 & ResNet50[11], and Xception[12]. In addition to research on the automation of detection and classification of weld defects, the researchers compared the performance of several CNN-based pre-trained models as feature extractors in a variety of problems, such as rice plant disease classification[13], daily object image classification[14], knee image classification[15], chest pneumonia classification[16], and eye cancer classification[17].

This study employs the deep convolutional neural network (CNN) architecture to extract features from a radiographic image with defects. The extracted features are input to a simple classifier. We achieve a short computation time with high accuracy during training, validation, and testing. Then, we evaluate the performance of each CNN architecture as a feature extractor without modifying the input dataset or the configuration of the classification model. Finally, we compared the effectiveness of our work to state-of-the-art methods. This work contributes to selecting the most efficient CNN architecture for classifying specific types of weld defects and provides rapid, straightforward computations. A comprehensive step is required to construct a weld defect classification model. In the second section of this paper, the stages of preparing radiographic image datasets are

described, followed by an explanation of the CNN architecture used as a feature extractor, the implementation of multilayer perceptron (MLP) as a classifier, and performance evaluation indicators. The experimental results have been described, including the obtained dataset and model performance testing and comparison. The paper ends with conclusions and suggestions for future work.

METHOD

Preparing datasets derived from radiographic images is the first stage in this study. The obtained images are augmented to ensure sufficient for the model to be built after being cropped to remove everything except one class of weld defects. Each set is fed into the pre-trained CNN architectures for feature extraction. The dataset is split into training, validation, and testing sets. The output of the feature extraction stage is referred to as the bottleneck features, which are input for the classifier or Fully Connected (FC) layer in the form of a multilayer perceptron using the Leaky ReLU activation function and the Softmax activation function in the final layer for categorical classification.

Data Preparation

We utilized the publicly accessible GDXray database for our work. The original radiographic films were scanned in high-density mode using the Lumisys LS85 SDR scanner. After visually adjusting the image content, the original 12-bit data depth was rescaled to 8-bit using a linear LUT proportional to the optical film density. This ensures that the 8-bit images retain all required defect information. Although only approximately 68 digitized radiographic images are available, they have a high resolution and represent multiple classes of weld defects. Radiographic images may contain multiple types of defects. To obtain an image with only one type of defect, it is therefore necessary to crop the whole image into small patches. We manually crop images to 224 x 224 pixels because this is the default input image size required by most CNN architectures we implement. Figure 1 shows an image patch divided into six classes of defects.
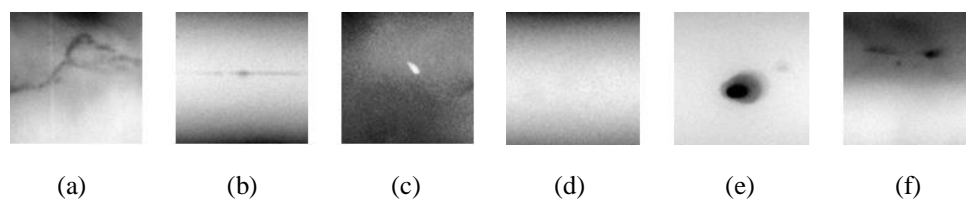

| (a) | (b) | (c) | (d) | (e) | (f) |

Figure 1. Image patches contain only one defect class. a) crack, b) lack of penetration, c) metallic inclusion, d) non-defect, e) porosity, f) slag inclusion

Due to the limited number of image patches, we use image augmentation techniques to artificially increase the size of the training set, thereby providing the model with more images for training, enhancing model accuracy, and reducing overfitting[18]. Several image augmentation methods based on generative adversarial networks (GAN), including Wasserstein GAN[19], contrast enhancement conditional GAN[12], and attention self-supervised learning-auxiliary classifier GAN[20], have been considered to be significantly effective in improving the model's performance. Instead of these three methods, we employ simple image augmentation techniques, including vertical-horizontal flipping and brightness changes, ensuring that the amount of data stays constant to the level where it limits the computational work. Figure 2 shows the results of image augmentation on an image patch containing crack-type defects.
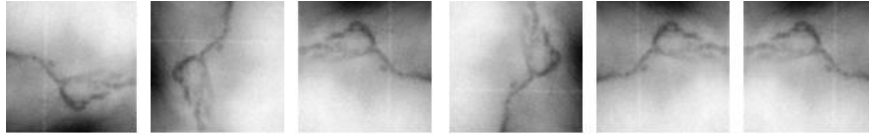
Figure 2. Example of image augmentation results on "crack"

The total number of generated image patches is split into training sets, validation sets, and testing sets with a ratio of 7:2:1. Thus, 70% of the training sets and 20% of the validation sets are selected randomly from the image patch collection. In comparison, the other 10% is utilized for testing. It should be noticed that all images in the testing data are entirely distinct from those in the training and validation data.

CNN Architectures

As feature extractors, various types of modern CNN models are implemented in this work. Several CNN models with pre-trained weights were chosen. The Keras Library provides access to these pre-trained models. In addition to serving as feature extraction methods, these pre-trained models can be utilized for predicting and fine-tuning. The CNN architecture is as outlined below:

1) DenseNet architecture. This architecture is comprised of numerous dense blocks with differing amounts of filters. The dimensions of the block are unique. For batch normalization, transition layers are positioned between blocks. Using downsampling to complement the dimensions of the subsequent layer[21]. The Keras Library includes the DenseNet architectures DenseNet 121, DenseNet 169, and DenseNet 201.

2) Inception architecture. This architecture initially debuted as GoogLeNet, or Inception V1, and was subsequently enhanced as Inception V2 and Inception V3. Inception is a contextual convolutional feature extractor that can learn additional representations with fewer parameters[22]. This architecture was improved to become Inception V4 and combined with ResNet to construct Inception-ResNet. In other forms, it becomes Inception-ResNetV2[23].

3) MobileNet architecture. This architecture is founded on a simpler algorithm that utilizes depthwise separable convolutions. It performs a minor computational task with rapidity and precision[24]. MobileNetV2 employs lightweight depthwise convolutions to filter intermediate expansion layer features instead of expanding output representations[25].

4) NASNet architecture. This architecture was discovered through a neural architecture search (NAS). NASNet is a scalable CNN architecture comprising fundamental building blocks that have been optimized[26].

5) ResNet architecture. The fundamental concept of residual networks (ResNet) is to use convolutional layer blocks by forming residual blocks from shortcut connections to create networks that can update the layer's weight to a lower depth[27]. These stacked residual blocks enhance training efficiency and address existing problems with deep tissue degradation.

6) VGG architecture. The Visual Geometry Group's (VGG) network with 16 layers and 19 layers, VGG16 and VGG19, respectively, served as the foundation for the ImageNet Challenge 2014 submission. VGG16 consists of thirteen convolutional layers, five pooling layers, three fully connected layers, and a softmax layer, frequently used in classification problems because of its high accuracy. Like the VGG16, the VGG19 comprises multiple layers, resulting in a more intensive training process[28].

7) Xception architecture. This architecture comprises a linear stack of depthwise separable convolution layers with residual connections to decrease time and space complexity[29]. Xception is an acronym for 'Extreme Inception' because this hypothesis is a more robust type of the foundational hypothesis of the Inception architecture.

Feature Extraction using CNN-based Pre-trained Models

Transfer learning is based on utilizing pre-trained CNN models to solve our problems, which may be distinct. We use a pre-trained model that has been trained without retraining it, or we freeze the weight of this pre-trained model to implement transfer learning. In this instance, the frozen model is utilized as a feature extractor. The pre-trained CNN model provides an arbitrarily selective feature extractor during feature extraction. The input image can transmit forward through a pre-trained model, break at a pre-specified layer, and extract features from the input image using the layer's output. This feature is known as a bottleneck feature. The extracted features can be quickly processed as input for the classifier and stored as a numerical array. In the proposed approach, we randomly utilized CNN architectures from Keras libraries, including DenseNet201, InceptionV3, MobileNetV2, NASNetMobile, ResNet50V2, VGG16, VGG19, and Xception, as a pre-trained base model that has completed the task of object detection on a data set from ImageNet, which contains public data and up to 1.28 million images from 1000 classes.

Classifier Architecture

The bottleneck features generated by the feature extraction stage are in a two-dimensional array; therefore, they must be flattened into one-dimensional single-vector data. The FC-100 layer receives data in a single dimension and then transmits it to the FC-50 and FC-6 layers using the Leaky ReLU activation function, which has a negative slope coefficient of 0.3. Leaky ReLU is a ReLU-based activation function with a slight slope for negative values instead of a flat slope. Before training, the coefficient value is decided; it is not acquired through training. The Softmax activation function is employed to classify the category of weld defects in the final layer.

Performance Indicator

A set of evaluation indicators is provided to verify the model's performance. The accuracy value is the most widely employed model performance metric. In image classification, a confusion matrix is utilized to accurately describe the model's performance by comparing the predicted and actual labels. The confusion matrix, on the other hand, reveals true and false classifications and error categories at the class level. TN, which stands for "true negative," represents the number of cases accurately identified as harmful.

Similarly, TP stands for "true positive" and denotes the number of positively classified instances that were accurately identified. "False negative" (FN) refers to the number of actual positive cases misclassified as unfavorable. "False positive" (FP) refers to the number of actual negative examples misclassified as positive. Figure 3 represents the confusion matrix cells.

Predicted Labels

Figure 3. Confusion matrix

Using the confusion matrix, the six types of weld defect distribution can be identified directly, with the accuracy determined by equation (1).

$$A_{CC} = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

In contrast to accuracy, misclassification refers to the proportion of incorrectly predicted data relative to the total prediction.

RESULTS AND DISCUSSION

After performing image preprocessing, which included image cropping and simple data augmentation, we obtained 1178 images with a size of 224 x 224 pixels. These images were divided into six classes of weld defects, including crack (CR), lack of penetration (LP), metallic inclusion (MI), non-defect (ND), porosity (PO), and slag inclusion (SI). The types of welding defects we have selected are the types that commonly occur during the welding process. Table 1 shows the distribution of the number of images used for training, validation, and testing in each class.

Table 1. Datasets Distribution

Class	Train set	Val set	Test set	Total
Crack (CR)	196	56	29	281
Lack of Penetration (LP)	105	30	15	150
Metallic Inclusion (MI)	37	10	6	53
Non-defect (ND)	175	50	25	250
Porosity (PO)	176	50	26	252
Slag Inclusion (SI)	134	38	20	192
Total	**823**	**234**	**121**	**1178**

In comparison to other classes of weld defects, the number of image datasets generated for the metallic inclusion class is disproportionately low. This is because the metallic inclusion class has fewer original images than the other classes. It is intriguing for us to determine if these issues affect the performance of the classification model that we have developed. Each data set is fed to the DenseNet201, InceptionV3, MobileNetV2, NASNetMobile, ResNet50V2, VGG16, VGG19, and Xception architectures to extract features. The feature extraction output is a two-dimensional array of data called bottleneck features. As a result of comparing eight feature extractor architectures, we obtain eight bottleneck features for training, validation, and testing, named *bottleneck_features_training*, *bottleneck_features_validation*, and *bottleneck_features_testing*, respectively. The

bottleneck_features_training and *bottleneck_features_validation* are used to train and validate a weld defect classification model.

We use an epoch of up to 100 because the training accuracy value for the eight used feature extractor architectures, including DenseNet201, InceptionV3, MobileNetV2, NASNetMobile, ResNet50V2, VGG16, VGG19, and Xception, was close to 1 at that epoch. Figure 4 depicts the rising trend of training accuracy, whereas Figure 5 shows training loss, Figure 6 shows validation accuracy, and Figure 7 shows validation loss.
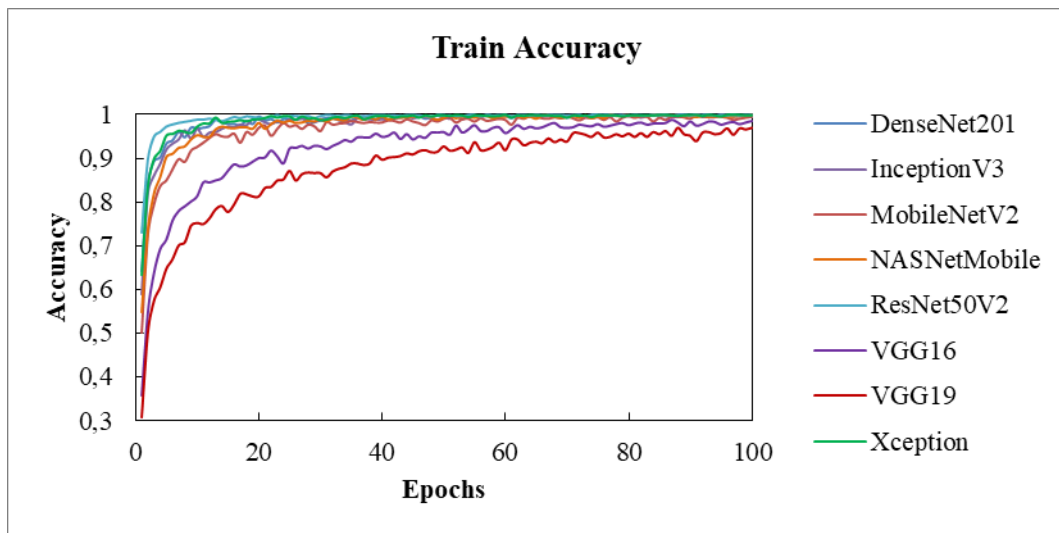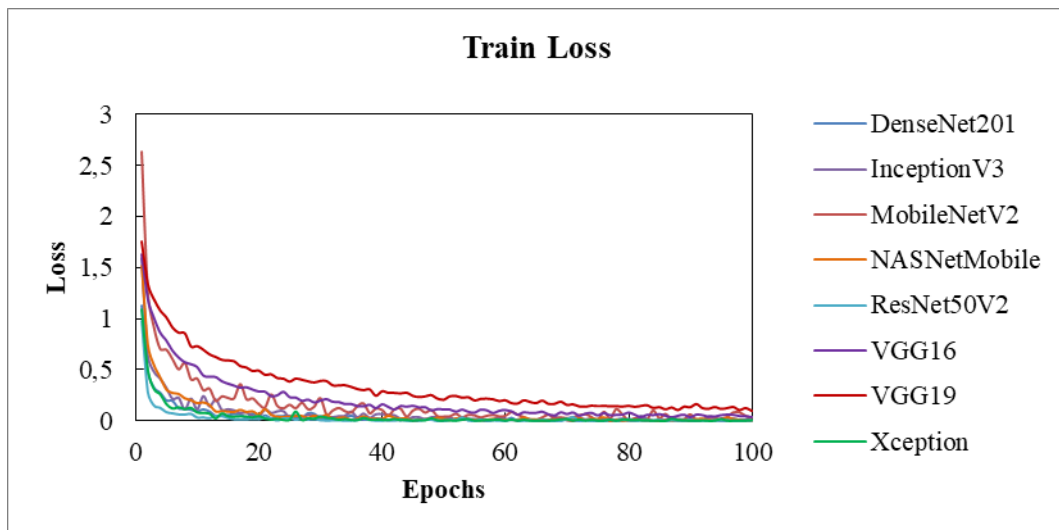

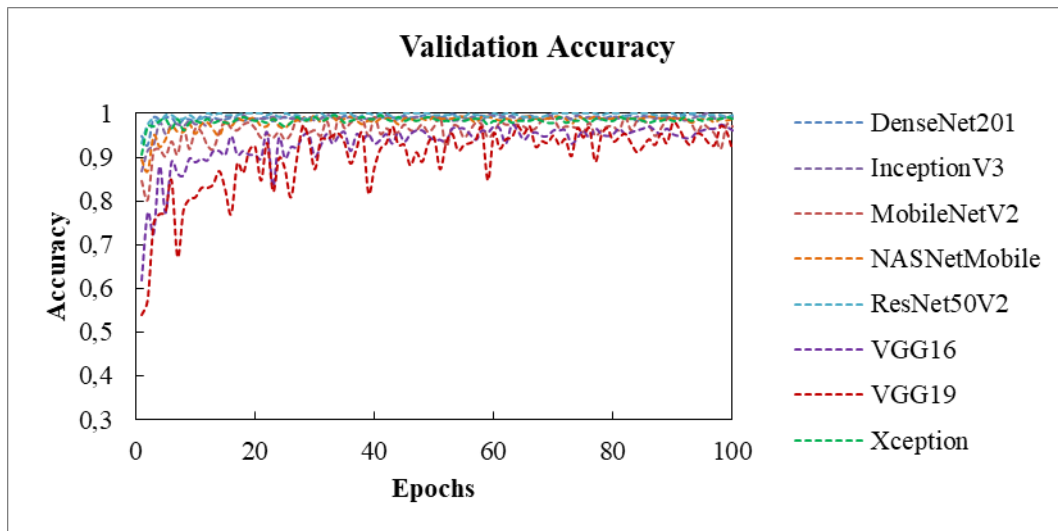**Figure 4**. Train accuracy

Figure 5. Train loss

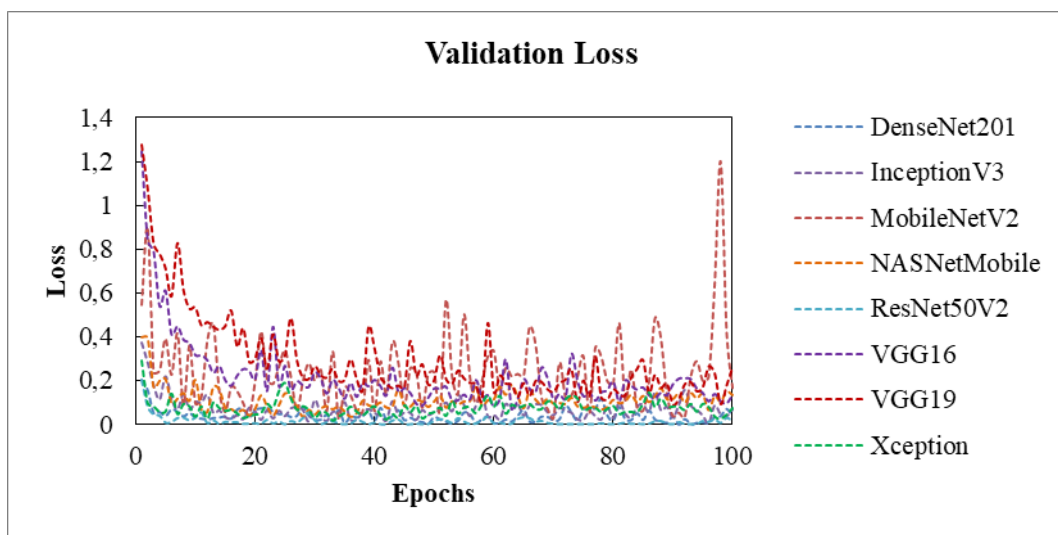**Figure 6**. Validation accuracy

Figure 7. Validation loss

Figure 4 illustrates that at minor epochs below 30, the distinction in training accuracy values is quite pronounced, and the classification model with feature extractors VGG16 and VGG19 has yet to reach 90% accuracy. Consequently, these two architectures require more training epochs than the others. Meanwhile, it can be observed in the classification model with other feature extractors that can achieve training accuracy above 90% prior to reaching epoch 10: ResNet50V2, Xception, DenseNet201, InceptionV3, NASNetMobile, and MobileNetV2 are, in order of speed, the algorithms with the highest accuracy. These six architectures do not necessitate an excessively long training epoch to achieve high accuracy. Particularly on ResNet50V2, Xception, and DenseNet201, the training accuracy value appears stable and close to 100%. The eight models were tested using unseen images taken from the testing set after we obtained eight weld defect classification models based on distinct feature extractor architectures. Therefore, the accuracy and misclassification of the classification model derived from the confusion matrix are tested using *bottleneck_features_testing*. Figure 8 illustrates each model's testing accuracy.
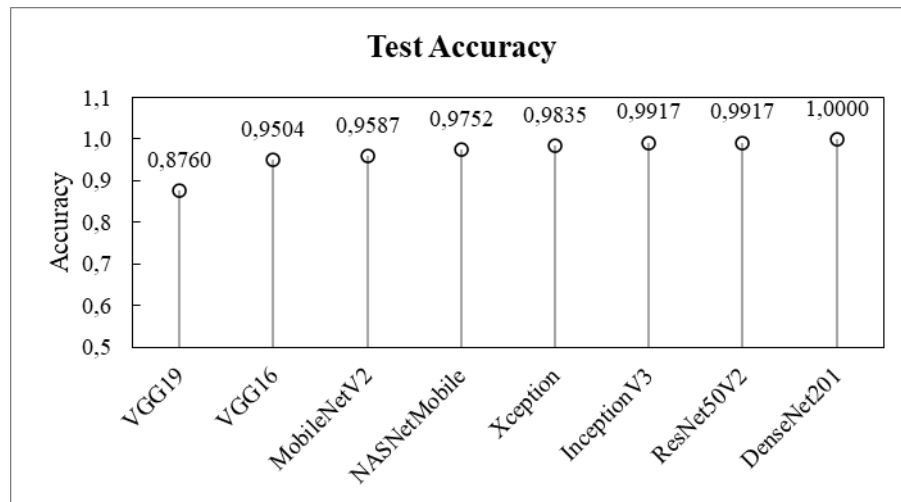
Figure 8. Comparison of testing accuracy results

Using a single random image, we further evaluated the eight classification models. By providing one image for each class as input, the classification model can predict the type of defect in the image. The test results are shown in Table 2.

Table 2. Comparison of prediction result using single random image test

			True Labels				
		CR	LP	MI	ND	PO	SI
Prediction Results	**DenseNet201**	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)
	InceptionV3	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)
	MobileNetV2	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)
	NASNetMobile	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)
	ResNet50V2	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)
	VGG16	CR (100%)	LP (100%)	MI (99.98%) ND (0.01%) PO (0.01%)	ND (100%)	PO (100%)	SI (100%)
	VGG19	CR (99.98%) LP (0.02%)	LP (93.57%) PO (6.43%)	MI (96.68%) PO (3.28%) SI (0.03%)	LP (0.19%) ND (99.78%) PO (0.03%)	PO (100%)	MI (0.01%) PO (32.63%) SI (67.36%)
	Xception	CR (100%)	LP (100%)	MI (100%)	ND (100%)	PO (100%)	SI (100%)

Based on testing with a single random image, most weld defect classification models can recognize the type of defect in each class with a percentage of 100% or close to it. In contrast, the classification model with a feature extractor based on VGG19 experiences conflicts in almost every defect class, excluding porosity. The model exhibits minor misclassification errors in other defect classes, including crack, lack of penetration, metallic inclusion, and non-defect. In the slag inclusion class, the misclassification rate exceeds 30%. Therefore, this model needs to be revised for determining the types of slag inclusion defects. This may be due to the similarity between the slag inclusion classes and the porosity. This model should be trained on a more significant number of data sets to overcome this issue. We compared the performance of our three best weld defect classification models, based on the DenseNet201, ResNet50V2, and InceptionV3 architectures, to that of a state-of-the-art model with a maximum of six weld defect classes that also uses the feature extractor architecture described in the CNN Architectures Section. Table 3 provides a performance comparison of the models.

Table 3. Performance comparison of different weld defect classification models

Reference	Number of Classes	Architecture	Evaluation Mode	Result
Thakkallapally [7]	3	VGG19	Train Acc	93.17%
			Val Acc	91.14%
			Test Acc	91%
Zhang *et al.* [19]	4	Inception & MobileNet	Acc (normal)	100%
			Acc (burn through)	94.77%
			Acc (crack)	99.75%
			Acc (porosity)	99.67%
Pan *et al.* [9]	5	MobileNet	Mean Acc	97.69%
Nazarov *et al.* [10]	5	VGG16	Acc	86%
Guo *et al.* [12]	5	Xception	Acc	92.5%
<u>Ours</u>	6	DenseNet201	Acc (DenseNet201)	100%
		ResNet50V2	Acc (ResNet50V2)	99.17%
		InceptionV3	Acc (InceptionV3)	99.17%

Our classification model outperforms the state-of-the-art model regarding accuracy and number of defect classes. The model proposed by Zhang *et al.*[19] has good accuracy for each class of defects, but compared to our model, it uses a much larger dataset, over 7000 images augmented with the WGAN technique. Guo *et al*.'s CECGAN augmentation technique generates an even more significant dataset [12]. In addition to a large number of datasets, Pan *et al.*[9] Moreover, Thakkallapally[7] utilized 6208 and 3000 images, respectively. Larger data sets necessitate more complex computing systems and longer processes. Meanwhile, our augmentation method is a simple, primary method that does not require time-consuming computational processes. The total number of datasets we employ is comparable to Nazarov *et al.*[10] who employ 1270 images divided into five classes of weld defects. However, our model is more accurate. Increasing the number of datasets will improve the classification model's accuracy. However, the number of this dataset should refer to the original image dataset before applying the augmentation technique. According to the obtained results, the weld defect classification model that we developed, particularly those based on the DenseNet201, ResNet50V2, and InceptionV3 feature extractors, could recognize and classify weld defects with very satisfactory performance and has the potential to detect weld defects in real time, particularly in the manufacturing industry.

CONCLUSION

A large dataset is required to create a weld defect classification model with high accuracy. However, the availability of public radiographic image datasets is limited, with imbalanced classes of weld defects. Developing a weld defect classification model using transfer learning methods and a pre-trained model as a feature extractor should be described as a method of overcoming this issue. This paper proposes selecting the optimal CNN-based pre-trained model architecture as a feature extractor for a simple classification model. Our study indicates that the eight pre-trained model architectures are capable of serving as feature extractors to classify six classes of weld defects, with results indicating that the three best classification models were obtained by using the DenseNet201 feature extractor with a test accuracy of 100%, followed by ResNet50V2 and InceptionV3 with an accuracy of 99.17%. In the future, we will develop models with more than six classes of weld defects, particularly models for effectively detecting classes of weld defects that occur relatively infrequently during welding. The preparation of datasets is also a crucial aspect of model development.

Consequently, we will provide this dataset as radiographic images of elliptical welding or double-wall techniques. Even though this dataset is public, it still needs to be explored. In the future, radiographic image datasets will be more widely available so that they can be utilized for future work, and weld defect detection, as well as the classification model, can be directly applied to tests based on real-time radiography.

ACKNOWLEDGMENTS

REFERENCES

1 Mery, D., Riffo, V., Zscherpel, U., Mondragón, G., Lillo, I., Zuccar, I., Lobel, H., & Carrasco, M. 2015. GDXray: the database of X-ray images for nondestructive testing. *J Nondestruct Eval*, *34*(42), 1-12.

2 Guo, W., Qu, H., & Liang, L. 2018. WDXI: The dataset of x-ray images for weld defects. *2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, *2018*, 1051-1055.

3 Perri, S., Spagnolo, F., Frustaci, F., & Corsonello, P. 2023. Welding defects classification through a Convolutional Neural Network. *Manufacturing Letters*, *35*, 29-32.

4 Oh, S-j., Jung, M-j., Lim, C., & Shin, S-c. 2020. Automatic detection of welding defects using Faster R-CNN. *Applied Sciences*, *10*(23), 8629.

5 Liu, W., Shan, S., Chen, H., Wang, R., Sun, J., & Zhou, Z. 2022. X-ray weld defect detection based on AF-RCNN. *Weld World*, *66*(2022), 1165–1177.

6 Golodov, V. A., & Maltseva, A. A. 2022. Approach to weld segmentation and defect classification in radiographic images of pipe welds. *NDT & E International*, *127*, 102597.

7 Thakkallapally, B. C. 2019. Defect classification from weld radiographic images using VGG-19-based convolutional neural network. *NDE*, *18*, 2019.

8 Ajmi, C., Zapata, J., Elferchichi, S., Zaafouri, A., & Laabidi, K. 2020. Deep learning technology for weld defects classification based on transfer learning and activation features. *Advances in Materials Science and Engineering*, *2020*.

9 Pan, H., Pang, Z., Wang, Y., Wang, Y., & Chen, L. 2020. A new image recognition and classification method combines a transfer learning algorithm and a MobileNet model for welding defects. *IEEE Access*, *8*(2020), 119951-119960.

10 Nazarov, R. M., Gizatullin, Z. M., & Konstantinov, E. S. 2021. Classification of defects in welds using a convolution neural network. *2021 ElConRus*, *2021*, 1641-1644.

11 Kumaresan, S., Aultrin, K. S. J., Kumar, S. S., & Anand, M. D. (2021). Transfer learning with CNN for classification of weld defect. *IEEE Access*, *9*, 95097-95108.

12 Guo, R., Liu, H., Xie, G., & Zhang, Y. 2021. Weld defect detection from imbalanced radiographic images based on contrast enhancement conditional generative adversarial network and transfer learning. *IEEE Sensors Journal*, *21*(9), 10844-10853.

13 Kongara, R. K. V., Somasila, V. S. C., Revanth, N., & Polagani, R. D. 2022. Classification and comparison study of rice plant diseases using pre-trained CNN models. *2022 International Conference on Inventive Computation Technologies (ICICT)*, *2022*, 140-145.

14 Mascarenhas, S., & Agarwal, M. 2021. A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for image classification. *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, *2021*, 96-99.

15 Rahouma, K., & Salama, A. 2021. Knee Images Classification using Transfer Learning. *Procedia Computer Science*, *194*, 9-21.

16 Ikechukwu, A. V., Murali, S., Deepu, R., & Shivamurthy, R. C. 2021. ResNet-50 vs VGG-19 vs training from scratch: A comparative analysis of the segmentation and classification of pneumonia from chest x-ray images. *Global Transitions Proceedings*, *2*(2), 375-381.

17 Santos-Bustos, D. F., Nguyen, B. M., & Espitia, H. E. 2022. Towards automated eye cancer classification via VGG and ResNet networks using transfer learning. *Engineering Science and Technology, an International Journal*, *35*, 101214.

18 Takahashi, R., Matsubara, T., and Uehara, K. 2020. Data augmentation using random image cropping and patching for deep CNNs. *IEEE Transactions on Circuits and Systems for Video Technology*, *30*(9), 2917-2931.

19 Zhang, H., Chen, Z., Zhang, C., Xi, J., & Le, X. 2019. Weld defect detection based on deep learning method. *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, *2019*, 1574-1579.

20 Yang, W., Xiao, Y., Shen, H., & Wang, Z. 2023. An effective data enhancement method of deep learning for small weld data defect identification. *Measurement*, *206*, 112245.

21 Biradar, S., Virupakshappa, Bagewadi, S., Veerashetty, S., & Ambika. 2022. Improved Densenet model for automatic categorization of brain tumors. *2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT)*, *2022*, 1-6.

22 Thanomsingh, J., & Supratid, S. 2021. A comparison study of using linear and nonlinear classifiers on object recognition based on inception convolutional neural networks with different numbers of inception-block. *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, *2021*, 35-38.

23 Lee, S-W. 2021. Novel classification method of plastic wastes with optimal hyperparameter tuning of Inception_ResnetV2. *2021 4th International Conference on Information and Communications Technology (ICOIACT)*, *2021*, 274-279.

24 Ran, X., Yan, T., & Cai, T. 2021. MobileNet for differential constellation trace figure. *2021 13th International Conference on Communication Software and Networks (ICCSN)*, *2021*, 168-172.

25 Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. 2019. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *arXiv preprint ArXiv*, 1801.04381.

26 YILMAZ, F., & DEMİR, A. 2020. Cutting effect on classification using Nasnet architecture. *2020 Medical Technologies Congress (TIPTEKNO)*, *2020*, 1-3.

27 Budhiman, A., Suyanto, S., & Arifianto, A. 2019. Melanoma cancer classification using resnet with data augmentation. *2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, *2019*, 17-20.

28 Liu, J. 2022. VGG, MobileNet and AlexNet on recognizing skin cancer symptoms. *2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI)*, *2022*, 525-528.

29 Wu, X., Liu, R., Yang, H., & Chen, Z. 2020. An Xception based convolutional neural network for scene image classification with transfer learning. *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, *2020*, 262-267.