Defect Classification from Weld Radiography Images Using VGG-19 based Convolutional Neural Network

Bharath Chandra Thakkallapally
JNTUH College of Engineering Hyderabad, Hyderabad, India
t.bharathchandra@gmail.com

Abstract. Weld radiography images are traditionally analyzed by the experts. The accuracy of this inspection process is more dependent on various external factors and is also time consuming. Due to these reasons, there is a need to perform automatic weld defect detection by analyzing the images obtained directly from the digital radiographic systems. A VGG-19 based Convolutional Neural Network is trained by using transfer learning technique over a sample of 3000 weld radiography images of size 128X128 pixels belonging to three different classes. This network is trained on Tesla K80 GPU for better results and it has got a training accuracy of 93.17%, a validation accuracy of 91.14% and a test accuracy of 91%. These results corroborate that the model is universal and ready for real world usage.

Keywords: Welding defects, Convolutional Neural Network, Transfer learning, Weld radiography.

1. Introduction

From decades, non-destructive testing (NDT) methods are implemented to check the quality of finished product in the industries. Radiography is one those methods which is mostly used to check the quality of the weld beads and casted components. Conventional radiography (CR) is done by passing X-rays through the component to be inspected and the results are captured by X-ray sensitive film placed behind the component. The radiographic images of components under inspection are traditionally analyzed by the experts with years of experience. The reports of this process can be influenced by various external factors and also a time taking process.

Welding is the most widely used method to fasten two components permanently. Defects in weld will greatly impact the strength of the joint. Cracks, solid inclusion, porosity, lack of fusion and undercut are some of the extremely common defects in welded joints.

From the past three decades, many researchers had suggested different methodologies to automate the task of defect detection and classification. Lashkia [1] had done great job in detecting defects by applying an adaptive threshold on output of membership function of a neuro-fuzzy system which take the spatial variance and spatial contrast at a particular pixel into consideration. Lashkia's system processes the image and converts it into a binary format which make task of inspectors easier, but this system did not fully automated the task of inspection. Alaknanda [2-3] had proposed an approach by first applying canny [4] edge detector over radiographic image, with an appropriate threshold and then by applying morphological closing (eroding after dilation of image) or multistage watershed segmentation technique to further improve the system. Silva [5-6], Wang [7], Lim [8] had extracted features of potential defects with the help of shape descriptors like position, aspect ratio, e/A ratio, roundness and many more. The values of these shape descriptors along with their defect label are fed into an artificial neural network (ANN) or support vector machines (SVM). But all these approaches need some feature extraction before feeding the data to a classifier. In the process of feature extraction, it may lose some potential defect data. With the usage of Convolutional Neural Network (CNN) can eliminate such limitations and this was a main motivation towards this work.

This work presents a unique approach to solve this problem by using VGG-19 [9] based CNN as classifier. Section 2 provides an overview of CNN. Section 3 has detailed the architecture of VGG-19 and transfer learning technique. Section 4 has described the methodology used in training the CNN, followed by results and conclusions.

2. Overview of CNN

Convolutional neural network is the type of neural network which uses a set of convolution layers before the dense layer in network architecture. Convolution is a mathematical operation that is performed between the kernel and the input volume. Feature maps are generated after convolving the filters over the input image. The number of filters in each layer of the network is completely the choice of the developer. The filters in the deep layers of CNN are capable of detecting complex features present in the dataset. Matt Zeiler and Rob Fergus [10] had made an exceptional study on visualizing the filters in the deep layers of CNN. After each convolution layer, it is a convention to add a ReLU layer for non-linearity and pooling layers are added in between a set of convolution layers to reduce the dimensions of input volume and control overfitting. Dropout layers are added between the layers of the network to reduce the chances of overfitting. At the end of these convolution and pooling layers, the output is flattened and fed it to conventional dense layers. Softmax function or sigmoid function is added at the end of the dense layers depending whether it is a multi-class classification or a binary classification respectively.

The filters which detect the required edges and shapes in CNN are adjusted by themselves through a training process called backpropagation. Backpropagation algorithm is implemented in four stages:

Forward pass: It is just the process of obtaining the output of an input image after feeding it to the network

Loss function: This is a function which helps in calculating the loss between the output obtained after forward pass and the label of the input image. The loss function generally used is mean squared error function (MSE).

Backward pass: Determining the contribution of each weight for the loss.

Weights update: An optimizer like Stochastic Gradient Descent (SGD) or Adam will help in reducing the loss by updating the weights of the filters and weights in the dense layers in network.

Fig. 1. Architecture of VGG-19

3. Architecture of VGG-19

There are six different configurations of VGG Net out of which VGG-16 and VGG-19 had produced better results comparatively to the other configurations. To address the complex nature of radiographic images, VGG-19 is used in current work. VGG-19 has 19 weighted layer and over 144 million parameters to be tuned while training. Fig. 1 shows the architecture of VGG-19 along with the number of filters in each convolution layer. VGG19 has only 3X3 filters with stride and pad of 1, along with 2X2 maxpooling layers with stride of 2. Down the network the number of filters keeps on multiplying. ReLU layer is applied after every convolution layer.

Transfer learning is a technique of using the weights or parameters of a pre-trained model to build a model on our dataset. Generally, transfer learning is used when a really huge dataset is not available to train a model from the scratch. Training CNN with small datasets will cause overfitting. Transfer learning can be applied in two different formats:

As Feature Extractor: By freezing all the weights of convolution layer and removing the fully-connected layers, the network can be used as feature extractor and the extracted data can be fed into an ANN or a SVM. This technique cannot be used in present work because our dataset is extremely different from ImageNet dataset (The dataset over which VGG Net is trained in ILSVRC 2014).

Fine-tuning: A new model can be created by freezing the weights of first few layers of VGG-19 and customizing the fully-connected layers to our requirement. This model can be fine tuned over our dataset. This technique can be used in present work by freezing the first five layers of pre-trained VGG-19 Net and training it over the current dataset of radiographic images.

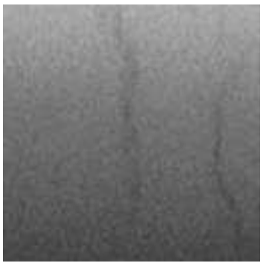
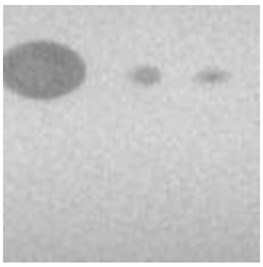
4. Methodology

4.1 Dataset Processing

GRIMA database of X-ray images (GDXray) [11] is used in this work for preparing the dataset. The GDXray welding subset contains of 78 X-ray images of different weld specimens. Length of each image in the dataset is nearly equal to 5000 pixels. It is decided to have only three different classes in our dataset: Porosity and Solid Inclusions (PO), Cracks (CR) and Good weld or no defects (GW). The reason behind grouping solid inclusions and porosity defects into a single class is because of the lack of large database of radiographic images. Every image from the GDXray dataset is chopped down into several 128X128 pixel images by sliding window method and then dropped them into any one of the above classes according to the defect present in it manually. Table 1 shows a sample image for each class. A final dataset of 3000 images is created, where each class contains 1000 images. This dataset is divided into training, validation and test subsets by transferring 60% of images into training set, 20% of images into validation set and remaining 20% to test set with random selection.

The data augmentation techniques like rescaling, horizontal flipping, zooming, width shifting, height shifting and rotating are used before feeding the data to the model.

Table 1 Sample image of each class

Cracks (CR)	
Good Weld (GW)	
Porosity or Solid Inclusions (PO)	

4.2 Model Architecture and Training

As mentioned earlier in section 2, a new network is created by freezing the first five layers of VGG-19 and replacing the fully-connected layers of VGG-19 by two dense layers and a dropout layer in between them. The final architecture of model along with the shape of output at each layer is given in the Table 2.

Softmax function is chosen as activation function at the last layer because it is a multiclass classification problem. The reason for placing a dropout layer in between the dense layers is to decrease the chance of overfitting and to generalize the model. 30% dropout means 30% of neurons present in the dense layer will show their activation as zero during the forward pass.

Training of the model was started by placing categorical cross-entropy as the loss function and SGD as an optimizer, with 0.0001 as learning rate and 0.9 as momentum. A batch of size 2 is used on training and validation sets during training. The model is trained on Tesla K80 GPU for better results.

Table 2 Architecture of model along with the output shape

Layer	Output shape
Input layer	128X128X1
Convolution	128X128X64
Convolution	128X128X64
Maxpooling	64X64X64
Convolution	64X64X128
Convolution	64X64X128
Maxpooling	32X32X128
Convolution	32X32X256
Convolution	32X32X256
Convolution	32X32X256
Convolution	32X32X256
Maxpooling	16X16X256
Convolution	16X16X512
Convolution	16X16X512
Convolution	16X16X512
Convolution	16X16X512
Maxpooling	8X8X512
Convolution	8X8X512
Convolution	8X8X512
Convolution	8X8X512
Convolution	8X8X512
Maxpooling	4X4X512
Flatten	8192
Dense	1024
Dropout (30%)	1024
Dense	1024
Dense(Softmax)	3

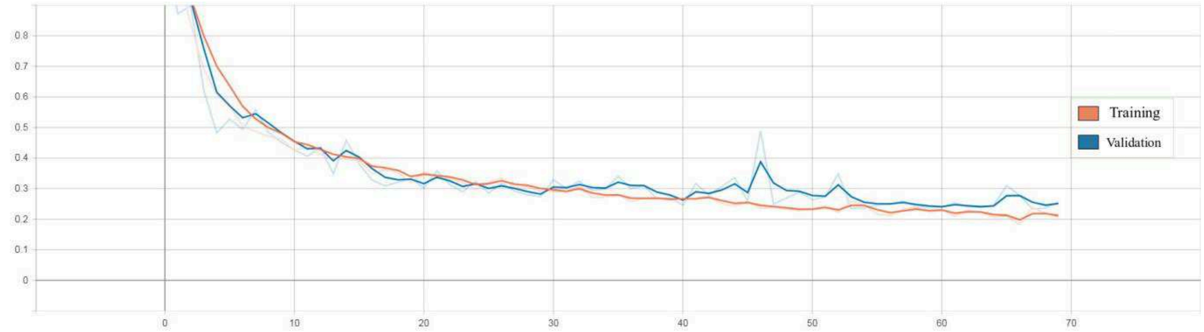


Fig. 2. Plot of training and validation loss while training

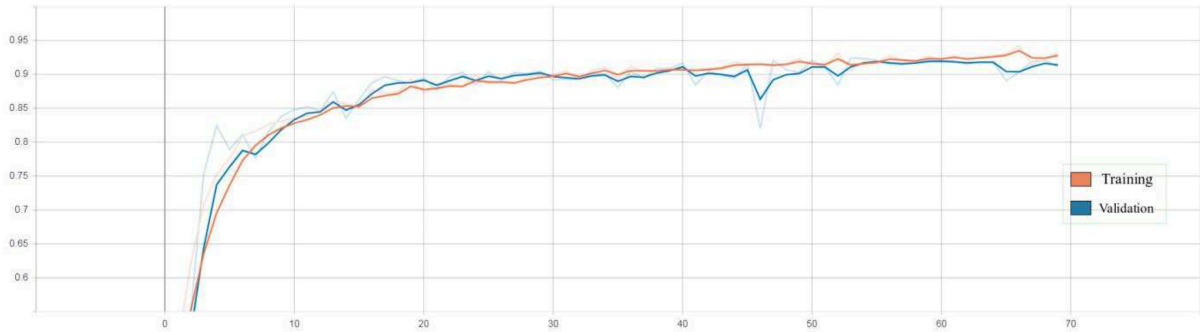


Fig. 3. Plot of training and validation accuracy while training

	precision	recall	f1-score	support
CR	0.93	0.90	0.92	200
GW	0.86	0.96	0.91	200
PO	0.94	0.87	0.90	200
accuracy			0.91	600
macro avg	0.91	0.91	0.91	600
weighted avg	0.91	0.91	0.91	600

Fig. 4. Classification report of predictions on test set

5. Results and discussion

After 70 epochs, the model had converged to a training accuracy of 93.17% and a validation accuracy of 91.14%. Figure 2 is the plot of training accuracy and validation accuracy collected while training the model. Figure 3 is the plot of training loss and validation loss collected while training the model. Both the plots are generated by Tensorboard and they have a smoothing factor of 50%. The lighter curves behind in plots are the original plots of accuracy and loss. The training and validation curves are pretty close to each other and they followed the expected trajectory almost precisely. From the above results show that the health of the model is good but there is still a chance that the model is not well generalized.

As mentioned in section 4.1, there is another set known as test set which will be used to check whether the model is well generalized or not. A classification report is generated after comparing the predictions on the test set by the model and their actual labels. Figure 4 shows the classification report which clearly specify the average precision and average recall are 91% and the overall accuracy of the model over the test set is also 91%, which is a pretty good result and a good sign that the model is well generalized.

6. Conclusion

A very unique approach to detect weld defects in X-ray images is presented in this paper. The classification of defects in X-ray images is done by a model which is trained by fine-tuning VGG-19 network. While using this model, a larger image should be chopped down into several images of target size (128X128 pixels) and get the predictions of these images by the current trained model. From these prediction, the frequency of each defect can be known and classify the original large image accordingly. This approach has got two main advantages:

1. It will not lose any potential defect data while feature extraction where other approaches does.
2. It can automate the process of inspection and also a less time taking process.

References

[1] Lashkia V (2001) Defect detection in X-ray images using fuzzy reasoning. Image and Vision Computing 19:261-269. doi: 10.1016/s0262-8856(00)00075-5

[2] Alaknanda, Anand R, Kumar P (2006) Flaw detection in radiographic weld images using morphological approach. NDT & E International 39:29-33. doi: 10.1016/j.ndteint.2005.05.005

[3] Alaknanda, Anand R, Kumar P (2009) Flaw detection in radiographic weldment images using morphological watershed segmentation technique. NDT & E International 42:2-8. doi: 10.1016/j.ndteint.2008.06.005

[4] CANNY J (1987) A Computational Approach to Edge Detection. Readings in Computer Vision 184-203. doi: 10.1016/b978-0-08-051581-6.50024-6

[5] da Silva R, Calôba L, Siqueira M, Rebello J (2004) Pattern recognition of weld defects detected by radiographic test. NDT & E International 37:461-470. doi: 10.1016/j.ndteint.2003.12.004

[6] da Silva R, Siqueira M, de Souza M et al. (2005) Estimated accuracy of classification of defects detected in welded joints by radiographic tests. NDT & E International 38:335-343. doi: 10.1016/j.ndteint.2004.10.007

[7] Wang Y, Guo H (2014) Weld Defect Detection of X-ray Images Based on Support Vector Machine. IETE Technical Review 31:137-142. doi: 10.1080/02564602.2014.892739

[8] Lim T, Ratnam M, Khalid M (2007) Automatic classification of weld defects using simulated data and an MLP neural network. Insight - Non-Destructive Testing and Condition Monitoring 49:154-159. doi: 10.1784/insi.2007.49.3.154

[9] Simonyan K, Zisserman A (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR abs/1409.1556:

[10] Zeiler M, Fergus R (2014) Visualizing and Understanding Convolutional Networks. Computer Vision – ECCV 2014 818-833. doi: 10.1007/978-3-319-10590-1_53

[11] Mery D, Riffo V, Zscherpel U et al. (2015) GDXray: The Database of X-ray Images for Nondestructive Testing. Journal of Nondestructive Evaluation. doi: 10.1007/s10921-015-0315-7