

# Deep-Learning Based Segmentation Algorithm for Defect Detection in Magnetic Particle Testing Images \*

Akira Ueda, Huimin Lu, Tohru Kamiya<sup>†</sup>

*Department of Mechanical and Control Engineering, Kyushu Institute of Technology, 1-1 Sensui-Cho, Tobata-ku, Kitakyushu-shi, Fukuoka, 804-8550, Japan<sup>‡</sup>*

## Abstract

Magnetic Particle Testing (MPT), also referred to as magnetic particle inspection, is a nondestructive examination (NDE) technique used to detect surface and slightly subsurface flaws in most ferromagnetic materials such as iron, nickel, and cobalt, and some of their alloys. In a bad environment, the procedure is complicated, and automation of MPT is strongly desired. To find defects in the formed magnetic powder pattern, it is required to be highly skilled and automation has been considered difficult. In recent years, many defect detection methods based on deep learning have been proposed, and the effectiveness of deep learning has been shown in the task of automatically detecting various types of defects having different shapes and sizes. In this paper, we describe the development of deep learning based segmentation algorithm for defect detection in MPT images. We have achieved a F2 score of 84.04% by using U-Net as the segmentation model and by utilizing a strong backbone network and an optimal loss function.

*Keywords:* Magnetic Particle Testing, Nondestructive Examination, Defect Detection, Segmentation.

## 1. Introduction

Magnetic Particle Testing (MPT) is used the leakage flux generated in cracks, holes, and internal defects when magnetizing a ferromagnetic material. By scattering the fluorescent magnetic powder while the test target is magnetized, the magnetic powder is attracted to the leakage magnetic flux generated in the defect. In that way, defects are highlighted to help visual confirmation.

Defects in MPT have various shapes such as circular, linear, etc. In addition, magnetic particles are also adsorbed on the irregularities other than defects, the residual magnetism, and the boundaries of the metal structure, and they have the property of forming "pseudo patterns". MPT provides very good defect resolution and is used extensively on: welded fabrications in magnetic material, castings, locating fatigue cracks in items subject to cyclical stress. However, cleaning and demagnetization are required when performing inspection, and development of automation technology is required to reduce work load and improve efficiency.

Regarding the identification of fatigue cracks of the target item, it is easy to check whether it is normal or abnormal, and it is possible to inspect it on the line. Therefore, automation has been realized by high-resolution cameras and image processing technologies such as binarization and thinning. However, in the field of welding processing of magnetic materials such as buildings and facilities, the surface condition is poor, and therefore many pseudo patterns occur. Moreover, since the shooting environment cannot be completely fixed, the difference in images is large. For the above reasons, image analysis has been difficult and automation has not been realized.

In recent years, research on anomaly detection areas by deep learning has made remarkable progress. There are many approaches such as binary classification of abnormal and normal, classification of abnormal data,

GAN[1] and VAE[2] that learns only normal data and detects abnormal data, detection of abnormal areas by object detection method and segmentation. They are highly accurate and have already been put to practical use in many fields. As described above, anomaly detection by deep learning is highly reliable. We believe that the effective use of deep learning can realize the automation of MPT, which has been considered difficult to detect defects.

In this paper, we discuss a deep learning based segmentation algorithm for the MPT automation. Based on U-Net, we apply an effective backbone network and loss function, and describe its effectiveness.

## 2. Datasets

### 2.1. Dataset

In this article, we use images taken in the field inspection of the welding fabrications of magnetic materials. The image size is 4608 x 3456 pixels (138 images) and 4000 x 2256 pixels (35 images). However, the defect area is very small for the image, which is too large to apply the deep learning model. Therefore, the area including the defect is cropped, and the training and testing are executed. The procedure for creating a data set is as follows.

1. The original image is cropped at the size of 512 x 512 pixels including defects (total 173 images). An example of the generated images is shown in Fig.1 (a). When cropping, we consider not to have all the defects in the same position.
2. In the cropped image, the label images were created by annotating the defect area in detail using labeling technique. An example of annotation of the defect images is shown in Fig.1.

### 2.2. Type of defects

The dataset includes defects of various sizes and shapes, from 5 mm to 150 mm with circular, linear, and so on. Circular and near-circular defects with the size of 5 mm to 10 mm are the most common, followed by linear defects with the size of 20 mm to 50 mm. It also contains a small number of linear defects of about 150 mm and defects that do not fit any of them. Fig.2 shows an example of defects.

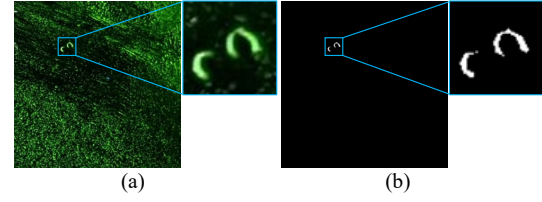


Fig. 1. Examples of annotation of defect image. (a) An image including defects and (b) a label image are shown.

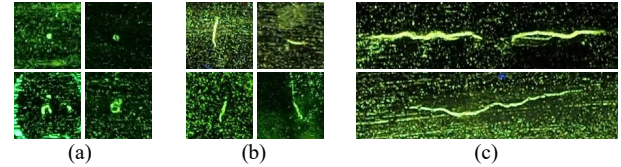


Fig. 2. Examples of defect types. (a) Circular and near circular defects, (b) linear defects, and (c) large linear defects.

## 3. Methods

We adopted U-Net[3] as the segmentation model. The detection score is improved by changing the encoder and applying the loss function considering the balance between classes. This section gives an overview of our approach.

### 3.1. Models

U-Net is a state-of-the-art semantic segmentation method with an encoder/decoder architecture. These encoder/decoder architectures use skip connections to preserve the location information lost due to convolutions, allowing for more precise output. Also, instead of using the original U-Net encoder, we use EfficientNet[4], which claims to be balanced between network depth, width, and resolution.

EfficientNet is available with different versions, starts from B0 at 5.3 million parameters to B7 at 66 million. Subsequent results show that a deeper encoder is not needed, so the subsequent experiments will be performed with a smaller encoder (EfficientNet B1).

We speed up the training process with pre-trained weights. Although it was trained on ImageNet, which is a database of natural images, the pre-trained weights do improve the training and local validation scores.

### 3.2. The loss function

Since the defects in MPT images are very small, training is greatly affected by the balance between the

Table. 1. Quantitative result of using different backbone network for U-Net encoder. In addition to the original encoder and EfficientNet B1 to B4, these show the results of experiments with ResNeXt50[5].

Models	Precision	Recall	F1	F2
U-Net	0.7147	0.7390	0.6980	0.7156
U-Net + EfficientNetB1	<b>0.8166</b>	<b>0.7929</b>	<b>0.7933</b>	<b>0.7904</b>
U-Net + EfficientNetB2	0.8085	0.7718	0.7777	0.7716
U-Net + EfficientNetB3	0.7886	0.7898	0.7712	0.7784
U-Net + EfficientNetB4	0.8143	0.7817	0.7865	0.7811
U-Net + ResNeXt50	0.7490	0.7500	0.7227	0.7362

background and defect classes. Therefore, it is desirable to use a loss function suitable for the target.

In the task of segmentation, Dice score coefficient (DSC) is commonly used. The 2-class DSC variant for class  $c$  is expressed in Equation 1, where  $g_{ic} \in \{0,1\}$  and  $p_{ic} \in \{0,1\}$  represent the ground truth label and the predicted label, respectively. The total number of pixels in an image is denoted by  $N$ . The  $\epsilon$  provides numerical stability to prevent division by zero.

$$DSC_c = \frac{\sum_{i=1}^N p_{ic} g_{ic} + \epsilon}{\sum_{i=1}^N p_{ic} + g_{ic} + \epsilon} \quad (1)$$

The linear Dice loss (DL) is therefore defined as a minimization of the overlap between the prediction and ground truth:

$$DL_c = \sum_c 1 - DSC_c \quad (2)$$

One of the limitations of the Dice loss function is that it equally weighs false positive (FP) and false negative (FN) detections. In practice, this results in segmentation maps with high precision but low recall. With highly imbalanced data and small ROIs such as small defects, FN detections need to be weighted higher than FPs to improve recall rate. The Tversky similarity index is a generalization of the Dice score which allows for flexibility in balancing FP and FNs:

$$TI_c = \frac{\sum_{i=1}^N p_{ic} g_{ic} + \epsilon}{\sum_{i=1}^N p_{ic} g_{ic} + \alpha \sum_{i=1}^N p_{ic} g_{i\bar{c}} + \beta \sum_{i=1}^N p_{i\bar{c}} g_{ic} + \epsilon} \quad (3)$$

where,  $p_{ic}$  is the probability that pixel  $i$  is of the lesion class  $c$  and  $p_{i\bar{c}}$  is the probability pixel  $i$  is of the non-lesion class,  $c^-$ . The same is true for  $g_{ic}$  and  $g_{i\bar{c}}$ , respectively. Hyperparameters  $\alpha$  and  $\beta$  can be tuned to shift the emphasis to improve recall in the case of large class imbalance. The Tversky index is adapted to a loss function (TL) in [6] by minimizing  $\sum_c 1 - TI_c$ .

#### 4. Experiments and results

To detect a defect region on a sequential image, we mainly conducted two experiments.

1. Experiment to verify the effectiveness when using EfficientNet as U-Net encoder and to determine the optimal EfficientNet version.
2. Experiment to verify the effectiveness of Tversky loss compared to Dice loss. For Tversky loss, we search for optimal  $\alpha$  and  $\beta$  values.

We used 80% of the dataset for training and 20% for testing. When partitioning the dataset, we took care to ensure that the defect types (circular, linear, large defects, etc.) are even. We trained for 500 epochs with batch size of 2 and evaluated using the weight that gave the highest score. Also, we used Adam as an optimization algorithm and trained with an initial learning rate of 0.001. We used precision and recall, F1 score, which is the harmonic mean of them, and F2 score, which emphasizes recall, as evaluation functions.

The results of each experiment are shown in Table.1 and Table.2. From the Table.1, by using EfficientNet as the encoder of U-Net, both the precision and recall improved. In addition, among the models used, EfficientNet B1 achieved the highest score in both precision and recall, and we did not see any improvement by increasing the

Table. 2. Quantitative results of using different loss functions (Dice loss, Tversky loss). We use EfficientNet B1 for Unet encoder (Efficient U-Net)

Models	Parameters	Precision	Recall	F1	F2
Efficient U-Net + DL	$\alpha = 0.5, \beta = 0.5$	<b>0.8166</b>	0.7929	<b>0.7933</b>	0.7904
Efficient U-Net + TL	$\alpha = 0.6, \beta = 0.4$	0.7099	0.8760	0.7725	0.8281
Efficient U-Net + TL	$\alpha = 0.7, \beta = 0.3$	0.7112	0.8922	0.7812	<b>0.8408</b>
Efficient U-Net + TL	$\alpha = 0.8, \beta = 0.2$	0.7086	0.8765	0.7720	0.8276
Efficient U-Net + TL	$\alpha = 0.9, \beta = 0.1$	0.6112	<b>0.9258</b>	0.7275	0.8305

version. It shows that deeper encoders are not needed in the segmentation of MPT defects. We also found that using EfficientNet as the backbone is more effective than ResNeXt50.

From Table.2, it was found that using Tversky loss instead of Dice loss improves Recall and F2 score. It is considered that the weighting of FN detection higher than that of FP led to improvement in recall. The highest F2 score was obtained when  $\alpha = 0.7$  and  $\beta = 0.3$ . Fig.2 shows a comparison of output images when using Dice loss and Tversky loss. We obtained better output results when using Tversky loss.

## 5. Discussion and conclusions

We examined the defect segmentation method using the MPT defect dataset. It was verified that the EfficientNet used as the backbone of U-Net is effective and does not require a large model holding a large number of parameters. It was also shown that Tversky loss is effective in advancing the training of MPT defect dataset.

In this paper, we focused only on the areas containing defects, but we need to consider an approach to normal areas. Therefore, we created normal area images dataset from the original images described in Section 2. We prepared about 3000 normal images of 512 x 512 pixels. For those normal images, we tested using the segmentation model built in Section 3. Images that could be recognized as normal were not over-detected, and there was no false detection of pseudo-patterns or other noise that did not resemble defects. However, there were several cases in which the pseudo patterns similar to defects were erroneously detected. We need to improve them.

As the countermeasure, it is possible to include normal images in the training dataset. However, it is considered that the inclusion of normal images that has not been erroneously detected causes a decrease in detection

accuracy. Therefore, we are mainly considering the following approaches.

- i) Include normal images with remarkable pseudo patterns in the training dataset.
- ii) Build a model to classify normal and abnormal and ensemble with our segmentation model. The approach to the normal area is important, and we would like to work on it in the future.

We developed a deep learning based segmentation algorithm for MPT defects. We achieved an F2 score of 84.08% by utilizing the backbone network and the loss function that facilitates optimization. In the future, we would like to consider the approach to the normal area.

## References

1. Houssam Zenati et al., "Efficient GAN-Based Anomaly Detection", arXiv preprint arXiv: 1802.06222, 2018.
2. Yuchen Lu et al., "Anomaly Detection for Skin Disease Images Using Variational Autoencoder", arXiv preprint arXiv: 1807.01349, 2018.
3. Ronneberger et al., "U-Net: Convolutional networks for biomedical image segmentation", International Conference on Medical Image Computing and Computer-assisted Intervention, pp.234-241, 2015.
4. Mingxing Tan et al., "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks", arXiv preprint arXiv: 1905.11946, 2019.
5. Saining Xie et al., "Residual Transformations for Deep Neural Networks", arXiv preprint arXiv: 1611.05431, 2016.
6. Seyed Sadegh Mohseri Salehi et al., "Tversky loss function for image segmentation using 3D fully convolutional deep networks", arXiv preprint arXiv: 1706.05721, 2017.
7. Nabila Abraham et al., "A Novel Focal Tversky loss function with improved Attention U-Net for lesion segmentation", arXiv preprint arXiv: 1810.07842, 2018.