ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

Industrial Defect Classification Algorithm based on Meta Learning

Yifan Luo, Mingquan Zhang*

School of Computer Science and Technology, North China Electric Power University, Hebei 071003, China

*Corresponding author Email: zmqcherish@163.com

Abstract

Aiming at the problems existing in the industrial defect classification scene, in order to improve the accuracy and robustness of classification, a defect classification algorithm combining meta learning and self adversarial training is proposed. When the data set is not ideal, through the optimized meta learning algorithm, the internal and external double-layer loop neural network mode is used to replace the original single-layer loop mode, so as to improve the generalization of the model and make the model show better classification accuracy in the case of few samples. Through self adversarial training, the sample breadth is widened, the feature learning ability of the model is improved, and the robustness of classification results is enhanced. Experiments show that the algorithm has better performance than the traditional method in the scene of industrial defect classification.

Keywords

Classification of Industrial Defects; Meta Learning; Self-adversarial Training; Deep Learning; Few Shot Learning.

1. Introduction

In industrial production, quality control is a very important link. Evaluate the product quality or the results of the production process. According to the method of identifying the defects on the product surface, the quality control strategy can be divided into destructive and non-destructive. Among them, nondestructive testing (NDT) belongs to the non-destructive category. Its purpose is to monitor the quality of a component without extracting or damaging samples. Nondestructive testing methods mainly include: visual method, dye penetration method, radiographic method, ultrasonic testing method, eddy current method, thermal imaging method, etc.

Among them, vision based defect detection method is one of the more common defect detection methods in industry. However, traditional artificial vision inspection is a difficult process to measure, and its results are variable and subjective Due to the complexity and uniqueness of this problem, researchers have developed a new and demanding automatic defect detection system. This system monitors the surface characteristics of materials and environmental conditions, and automatically monitors them through artificial intelligence algorithm.

However, there are many problems in the specific implementation, which can be summarized into two aspects: data set and neural network.

The dataset aspect contains the following questions:

(1) The data are inseparable. The standard of data differentiation is difficult to define, the gap between positive and negative samples is small, and it is difficult to have a consistent label to separate positive and negative samples. Even if standards are set, there are difficulties in implementation. No matter

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

using any means to describe defects, they cannot be clearly separable. For example, the histogram is drawn according to the area and gray value. There will always be a certain amount of samples in the middle transition area, which is in the gray area and ambiguous.

(2) Poor diversity. This shows that there are too many differences within the class. If it is the same scratch, it has different forms. Some scratches are white and some are black, and the shooting angle is also different. Therefore, it is difficult to collect all forms of defect samples, so it is difficult to have a good performance in the test set. There are obvious deviations between the training set and the test set that affect the performance. The deviation here is not caused by the annotation, but by the data itself.

(3) Sample imbalance. The sample level is unbalanced. A large number of samples are normal samples, and the proportion of defective samples is relatively small. The level of defect instance is unbalanced, and the defect accounts for a small proportion of the whole, 2500px × On the 2000px scale image, the defect scale may be only 10px × 10px level. If the defect is too small, the image cannot be resized, which leads to time-consuming test and difficult to control false detection.

(4) The data is dirty. Dirty data means that the label category is mistaken when labeling. Dirty data will have an adverse impact on network training, and forced training will have the risk of over fitting. Because the network extracts general features, if it cannot fit defects, it will fit other noises.

The problems of neural network are caused by the particularity of data set. Neural networks include the following problems:

(1) The feature extraction ability is poor and easy to be disturbed. Due to the difficulties in the collection of industrial defect data sets, the amount of data is usually small. If the deep neural network is selected, the training results are not fitted due to the insufficient amount of data. If the shallow neural network is selected, the neural network is difficult to learn the correct fault characteristics and the monitoring results are inaccurate due to the environmental differences in the data set, such as illumination, weather, smoke and dust.

(2) The generalization ability is poor, and it is easy to over fit on a single fault. When introducing new fault types, it is found that the neural network has been over fitted on the original fault, which makes it difficult to learn new features and the generalization of the model is very poor. At this time, it is often necessary to re train as a whole, which is time-consuming and laborious.

To solve the problems in landing, we should start from two aspects: data set and neural network. In terms of data set, some data enhancement methods can be used to strengthen the training results, such as mosaic data enhancement [1], or the robustness of the results can be enhanced by self adaptive training [1]. In the aspect of neural network, due to the rapid development of small sample learning in recent years Small sample learning attempts to realize the classification or fitting task under the condition of limited samples. The meta learning based on optimization method aims to learn a group of meta classifiers and fine tune the new task to achieve better performance, which is very suitable for the scene of industrial defect classification. Literature [2] proposed a model agnostic meta learning (MAML) method. The algorithm can achieve better performance only through a few steps of iterative updating when facing new tasks.

Therefore, this paper proposes a new industrial defect classification algorithm by using the improved meta learning algorithm combined with self adversarial training to enhance the data. Using the meta learning training method to search the optimal initialization parameters to obtain a better initialization model, and then quickly learn the data characteristics of the current task through a small number of specific industrial defect samples on the basis of the initialization model, and combined with the data enhancement algorithm to obtain a neural network that can quickly adapt to the detection and classification of specific industrial defects, It effectively improves the accuracy and robustness of defect classification, and can still obtain better results on unbalanced or small sample data sets.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

2. Related Technology

2.1 Meta Learning

In 1998, Thrun et al. detailed meta-learning in the modern deep neural network era in the literature [3], stating that for a given task, an algorithm is considered to be able to learn if it performs on that task "if it improves over experience". At the same time, for multiple tasks to be solved, an algorithm "if its performance on each task improves as experience and the number of tasks grows", it is considered that the algorithm is capable of learning how to learn, and refers to the latter as a meta-learning algorithm. It does not learn how to solve a particular problem, but can successfully learn how to solve multiple tasks. Whenever it learns to solve a new task, the more it is capable of solving other new tasks: it learns how to learn, and the article gives a clear definition of modern meta-learning algorithms, while also marking the beginning of modern meta-learning research.

Meta-learning is a concept that has a number of different implementation ideas, the more common of which is optimization-based meta-learning. The optimization-based approach is roughly based on the idea of starting one parameter $\theta$and generating another neural network that solves the task through the neural network. In the past, neural networks can be seen as training data, looking for a mapping between input $x$data and output data$y$, that is$y = f(x)$, meta-learning based on optimization is to train a series of tasks $T$to obtain a $f$function, and the function is used to solve the task that needs to be completed, its output is a function$f$, and then the input data through the function f x and get the output y, ie $f = F(T)y = f(x)$, that is.

In traditional deep learning, the training unit is a piece of data that optimizes the model; the data can be divided into training sets, test sets, and validation sets. In meta-learning, the training unit is hierarchical, the first layer of training units is the task, that is to say, in the meta-learning to prepare many tasks to learn, the second layer of training units is the corresponding data of each task.

The purpose of both is to find a function, but the functions of the two functions are different, and the things to be done are different. Functions in traditional deep learning act directly on features and labels to find associations between features and labels, while functions in meta-learning are used to find new $f$ones, and new $f$ones are applied to specific tasks.

But like traditional deep learning, optimization-based meta-learning also requires the backpropagation of derivatives to optimize functions F so that they can produce more ideal functions$f$.

The MAML algorithm proposed by Finn et al. in the literature [2] is a typical meta-learning algorithm based on optimization ideas, and this paper will also improve on this algorithm, MAML The algorithm is divided into two levels, internal and external, which perform gradient descent updates through training data and reflect changes externally. More complex alternatives also learn step lengths [4,5] or train recursive networks to come from gradients [6-8]. Predict the step size. The meta-optimization training process by gradient requires the calculation of second derivatives and potentially thousands of internal optimization steps, so it is also a challenge to computing power. For this reason, it is often used in small sample learning, because only a small number of cycle steps are required, so it is very suitable for industrial defect classification scenarios.

2.2 Self-adversarial Training

Adversarial training is an important way to enhance the robustness of neural networks. By mixing some perturbations in the sample, these perturbations are small but are likely to cause misclassification, and then adapt the neural network to this change, making it robust to the adversarial sample.

Adversarial training can be summarized as follows:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{Z},y)\sim\mathcal{D}} \left[\max_{\|\boldsymbol{\delta}\|\leq\epsilon} L(f_{\boldsymbol{\theta}}(\boldsymbol{X}+\boldsymbol{\delta}), y) \right]$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

The inner layer is a maximization, where$X$ the input representation of the sample represents the δ perturbation superimposed on the input, is a neural network function, is the label of the sample, then represents the superimposed on the sample of a perturbation $f_\theta()yL(f_\theta(X+\delta),y)X\delta$, and then through the neural network function, compared with the label to obtain the loss. is the optimization goal, i.e. to find the perturbation that maximizes the loss function.

The outer layer is the minimization formula for optimizing the neural network, that is, when the perturbation is fixed, we train the neural network model to minimize the loss on the training data, that is, to make the model have a certain robustness to adapt to this perturbation.

A sample-self-adversarial training method is proposed in the literature [1], which consists of two stages: in the first stage, the neural network changes the original image; in the second stage, the neural network is trained to perform tasks on the modified image in the normal way. Experiments show that the proposed method can effectively enhance the robustness of the network.

3. Algorithm Design

The algorithm designed in this paper consists of two inner and outer loops, which are measured in tasks. The general structure of the inner and outer circulation is as follows:

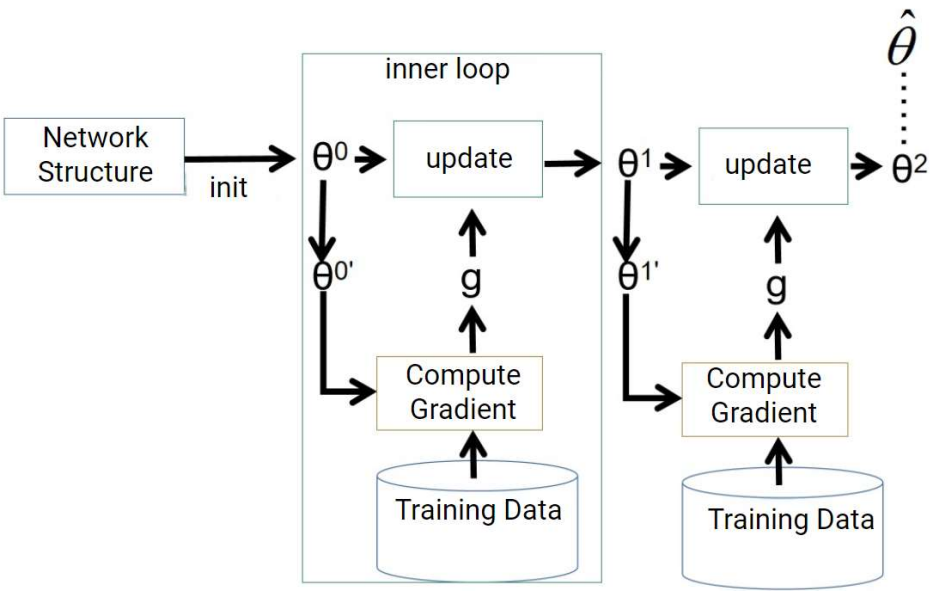

Fig. 1 The approximate structure of the algorithm

3.1 Task Settings

There are two modes of tasks in this article: 5way-1shot and 5way-5shot, taking 5way-5shot as an example: Set 80% of the total sample as meta training samples $\mathcal{D}_{\text{meta-train}}$ and 20% as meta test samples$\mathcal{D}_{\text{meta-test}}$. During the meta-train phase, each task randomly takes five categories from the total category, and each category is then taken from $\mathcal{D}_{\text{meta-train}}$ the middle 20 labeled samples are randomly taken to form one$task\ T$, of which 5 labeled samples are called$T$ support sets and the other 15 samples are called Tquery sets (query set). Take this $task\ T$as a piece of training data in ordinary machine learning, repeat the sample in the training sample 10 times, combine them into a batch, and train them through the initial neural network Do stochastic gradient descent SGD algorithm. During the fine-tune phase, the above operation is repeated, but the sample is taken from $\mathcal{D}_{\text{meta-test}}$.

3.2 Algorithmic Flow

In the task setup, the samples have been divided. The network is initialized randomly, the model is noted M_{meta} as, and the model initialization parameter isϕ. Take a task support set to trainM_{meta} ,

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

perform the first gradient descent, if it is the first task, the parameters are updated to$\hat{\theta}^1 \Leftarrow \phi - \eta \cdot \partial l(\phi)/\partial\phi$. When performing the second task, there is $\hat{\theta}^2 \Leftarrow \phi - \eta \cdot \partial l(\phi)/\partial\phi$. Then after performing the first batch size task, there is$\hat{\theta}^{bz} \Leftarrow \phi - \eta \cdot \partial l(\phi)/\partial\phi$. Each task is an inner loop in Fig. 1.

After the above training, take the query set in the previous task to test the $\hat{\theta}^i$model effect M_{meta} of the obtained parameters, and obtain the total loss function$L(\phi) = \sum_{i=1}^{bs} l^i(\hat{\theta}^i)$, which is for each task The sum of the $\hat{\theta}^i$losses in the M_{meta} query set in $l^i(\hat{\theta}^i)$the respective parameters.

After obtaining the total loss function, the gradient descent of the second type is to update the initialization parametersϕ, that is, to $\phi \Leftarrow \phi - \eta \cdot \partial L(\phi)/\partial\phi$update the initialization parameters. Repeat this step over and over again to get the model's better initialization parameters on the dataset. After that, $\mathcal{D}_{\text{meta-test}}$ the model is fine-tuned with the support set, and the query set used after fine-tuning $\mathcal{D}_{\text{meta-test}}$ evaluates the model.

Table 1. The algorithm pseudocode in this article

Process of industrial defect classification algorithm based on meta-learning

Input: task dataset distribution$p(\mathcal{T})$, step size hyperparameter α, β
Random parametersθ
While not done do
Take the task$\mathcal{T}_i \sim p(\mathcal{T})$
For all do$\boldsymbol{\mathcal{T}_i}$
Evaluate the results $\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$and update the parameters$\theta_i' = \theta - \alpha\nabla_\theta \mathcal{L}_{\mathcal{T}_i}(f_\theta)$
End for
update$\theta \leftarrow \theta - \beta\nabla_\theta \sum_{\mid T_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\theta_i'}\right)$
End while

3.3 Self-adversarial Training Design

Through the original sample, the calculation generates a perturbation sample, which is added to the original sample during the training process to enhance the robustness of the training result.

Let the original image be x, and the increased disturbance noise be $\epsilon \times \text{sign}(\nabla_x J(\boldsymbol{\theta}, \boldsymbol{x}, y))$, then the resulting interference image is $x + \epsilon\text{sign}(\nabla_x J(\theta, x, y))$. where ϵnoise multiples, $\epsilon \in [0,0.09]$for $\boldsymbol{\theta}$ added perturbations, y for labels. Each sample is generated by a noise algorithm to generate two interference images, which are mixed into the original sample.

4. Experiments and Analysis

4.1 Experimental Environment

The experimental environment of this paper uses the online GPU computing power service provided by Baidu Intelligent Cloud Platform. The memory is 80GB and the GPU model is the NVIDIA Tesla T4.

4.2 Experimental Dataset

The experiment used the Vision-based SIS for steel dataset. The dataset, published by Northeastern University, contains three types of data: 1, neu surface defect database. The dataset collected 6 defects of inclusion, scratching, pressing scale, crack, pockmark and plaque, 300 photos of each defect, and the image size was 200×200. 2, Micro surface defect database. Miniature strip defect data, image size 640×480, defects only about 6×6 pixels in size, Is a typical unbalanced sample. 3, Oil pollution defect database. The surface defect data set of silicon steel with oil pollution interference has a small amount of data, which is an extremely undesirable small sample scenario.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

This article divides the dataset into training sets and test sets in an 8:2 ratio, with 20 samples in each category subset, of which 5 are support sets, in addition 15 sheets are query sets. The task is divided into 1 category and 5 categories, and 5way-1shot and 5way-5shot verification are performed respectively.

Fig. 2 NEU surface defect database

Fig. 3 Micro surface defect database

Fig. 4 Oil pollution defect database

4.3 Experimental Results and Analysis

This paper uses a convolutional neural network structure to build the basic network structure, the convolutional kernel size is 3*3, the output channel is 32, and the gradient update is performed by using the SGD algorithm. For each 100-dimensional training task, the $\mathcal{D}_{\text{meta-test}}$ query set verification results are recorded and recorded as a batch.

The control network selected for this article is AlexNet[9], which is a very classic neural network for image classification, which is in The structure of the network is deepened on the basis of LeNet-5[10], which enables the learning of richer and higher dimensional image features, and the rejection of overfitting by dropout and pooling. Each 1000 samples trained during the training process is counted as a batch, and after each batch of training is completed, samples are taken from the test set to evaluate the network. In the experiment, due to the limitation of the data set, the original AlexNet model is prone to underfitting, so it is correct The AlexNet network structure has been modified to reduce the number of convolutional layers in the middle.

(1) Comparison of model performance on neu surface defect database

Fig. 5 and 6 show the accuracy curves of the algorithm and AlexNet's classification verification on the NEU surface defect database dataset.

Fig. 5 shows the accuracy ratio of the algorithm and AlexNet in the 5way-1shot case. It can be observed from Fig. 5 that the accuracy rate of the algorithm in this paper is higher than that of AlexNet in the entire training batch, and the highest accuracy rate is 70.5%, which is the highest accuracy rate than AlexNet 63.4% higher than 7.1%.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

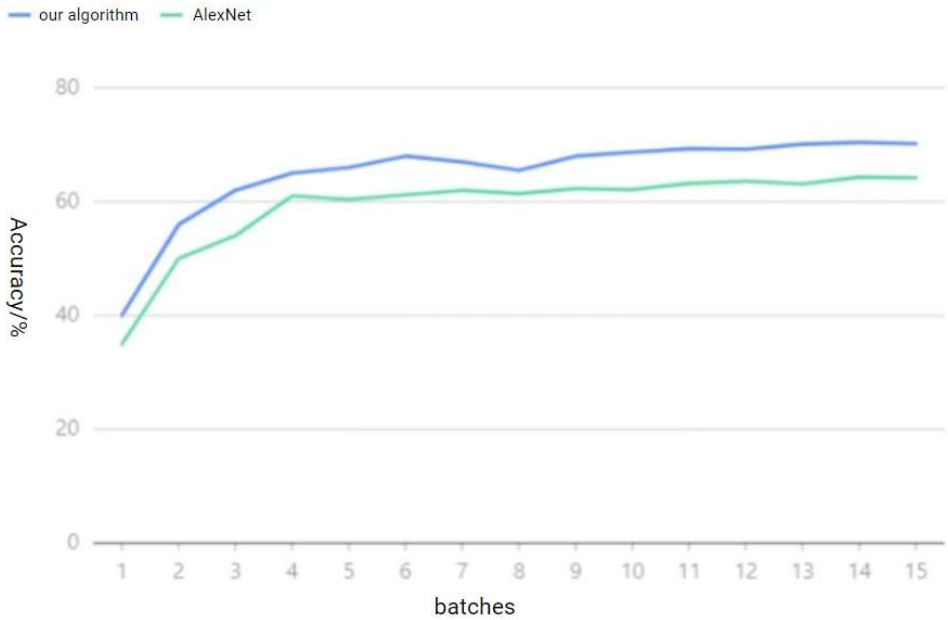

Fig. 5 Neu surface defect database 5way-1shot accuracy comparison

Fig. 6 shows the accuracy ratio of the algorithm and AlexNet in the 5way-5shot case. The highest accuracy rate of the algorithm in this paper is 79.9% in the entire training batch, which is 9.8% higher than the maximum accuracy rate of AlexNet of 70.1%. Experimental results show that the accuracy of AlexNet classification is significantly improved compared with that of 5way-1shot and 5way-5shot.

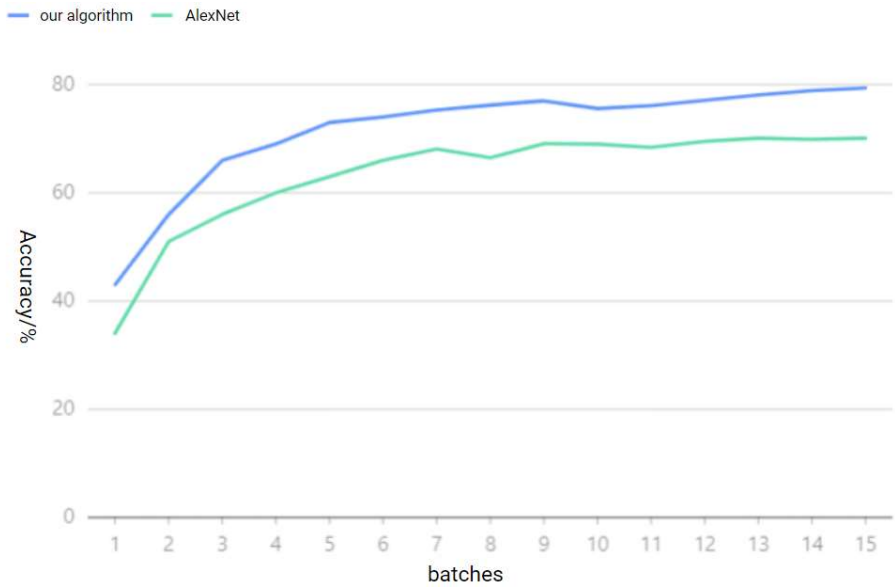

Fig. 6 Neu surface defect database 5way-5shot accuracy comparison

At the same time, it can be observed from the figure that both models have good robustness on this data set.

(2) Model performance comparison on the Micro surface defect database

Fig. 7 shows the accuracy comparison of the algorithm and AlexNet in the 5way-5shot case. It can be observed that the arithmetic of this paper shows a higher accuracy rate of 56.7%, which is higher than AlexNet's 45.5%, and the algorithm of this paper shows it Better robustness.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

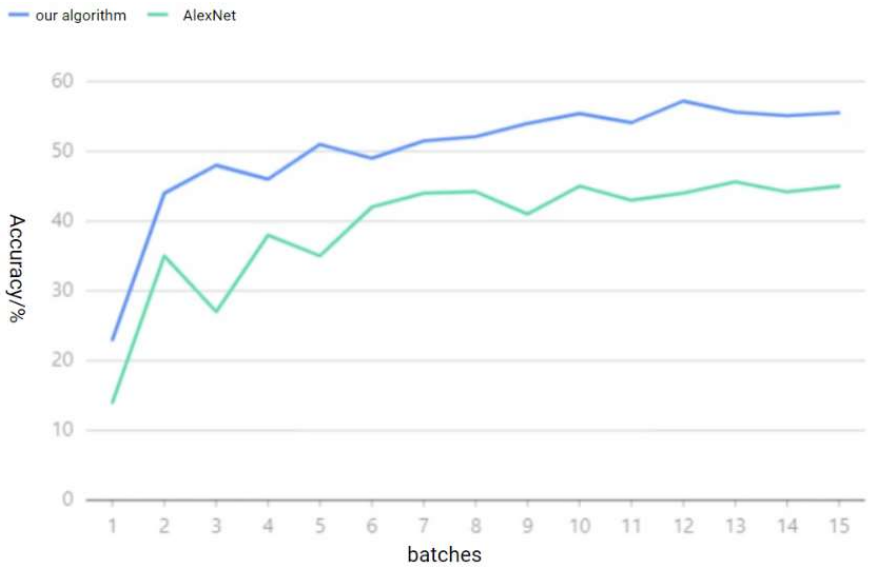

Fig. 7 Micro surface defect database 5way-5shot accuracy comparison

(3) Model performance comparison on the Oil pollution defect database

Fig. 8 shows the accuracy ratio of the algorithm and AlexNet in the 5way-5shot case. The accuracy of the algorithm in this paper is 45.2% higher than that of AlexNet 34.8%, and it has better robustness.

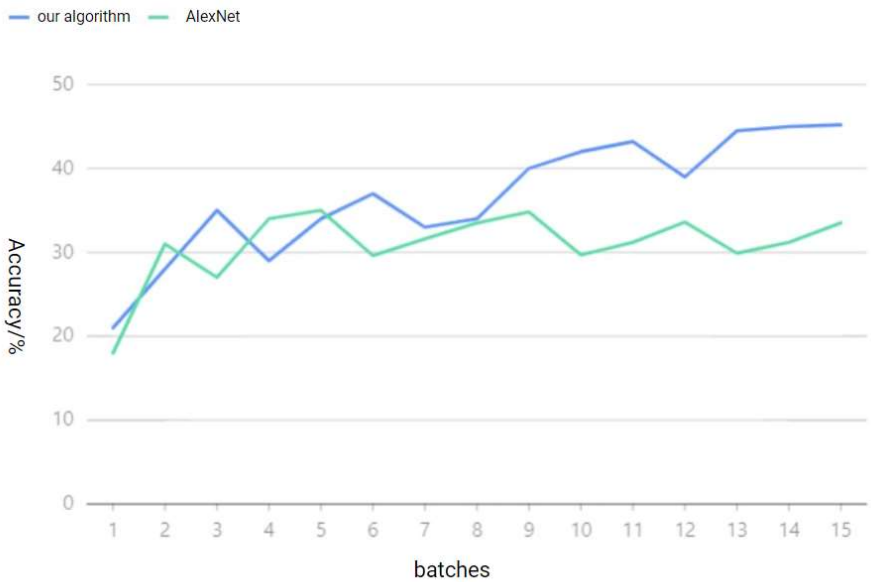

Fig. 8 Oil pollution defect database 5way-5shot accuracy comparison

The results are analyzed, because the data set in (1) is ideal, so the two models can learn the characteristics from the sample well, so they show good robustness, in terms of accuracy, the algorithm in this paper is significantly higher than AlexNet It is proved that the proposed algorithm has better generalization and is more suitable for use in industrial defect classification scenarios. In the two datasets of (2) and (3), due to the small amount of data, it is difficult for the model to learn the correct features, and the accuracy performance of the model is largely different from (1). (2) The dataset is a typical unbalanced data set, (3) the data set is a data set with extremely small data volume, and it can be found through comparison that the proposed algorithm can show a relatively traditional model in the case of extremely unsatisfactory data set With better robustness and accuracy,

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0076

experiments show that the proposed algorithm model is more suitable for industrial defect classification scenarios than the traditional model.

5. Conclusion

In this paper, a solution for self-adversarial training combined meta-learning is proposed for the problem of image classification of industrial defects. First of all, the algorithm increases the breadth of data through self-adversarial training, improves the robustness of the results, and increases the model's ability to perceive features. Secondly, through the meta-learning algorithm based on optimization, the model can show better generalization, and under the premise of extremely undesirable data sets, it can show relatives Better accuracy and robustness than traditional models. Experimental results verify the superiority of the proposed algorithm, and in the special scenario of industrial defect detection, this paper proposes a better solution.

Acknowledgements

This paper was financially supported by “the Fundamental Research Funds for the Central Universities (2020MS122)”.

References

[1] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[2] Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks[C]// International conference on machine learning. PMLR, 2017: 1126-1135.

[3] THRUN S, PRATT L. Learning to learn: Introduction and overview [M]. Learning to learn. Springer. 1998: 3-17.

[4] ANTONIOU A, EDWARDS H, STORKEY A. How to train your MAML [J]. arXiv preprint arXiv: 1810 09502, 2018.

[5] LI Z, ZHOU F, CHEN F, et al. Meta-sgd: Learning to learn quickly for few-shot learning [J]. arXiv preprint arXiv:170709835, 2017.

[6] ANDRYCHOWICZ M, DENIL M, GOMEZ S, et al. Learning to learn by gradient descent by gradient descent; proceedings of the Advances in neural information processing systems, F, 2016 [C].

[7] LI K, MALIK J. Learning to optimize [J]. arXiv preprint arXiv:160601885, 2016.

[8] RAVI S, LAROCHELLE H. Optimization as a model for few-shot learning [J]. 2016.

[9] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25.

[10] LeCun Y. LeNet-5, convolutional neural networks[J]. URL: http://yann. lecun. com/exdb/lenet, 2015, 20 (5): 14.