Optics and Lasers in Engineering xxx (xxxx) xxx

Contents lists available at ScienceDirect

Optics and Lasers in Engineering

journal homepage: www.elsevier.com/locate/optlaseng

Unified detection method of aluminium profile surface defects: Common and rare defect categories

Defu Zhang [a,b], Kechen Song [a,b,*], Jing Xu [a,b], Yu He [a,b], Yunhui Yan [a,b,*]

[a] School of Mechanical Engineering & Automation, Northeastern University, Shenyang, Liaoning, China
[b] Key Laboratory of Vibration and Control of Aero-Propulsion Systems Ministry of Education of China, Northeastern University, Shenyang, Liaoning, China

ARTICLE INFO

Keywords:
Aluminium profile surface defects
Common and rare defects
Attention mechanism
Proposal feature maps

ABSTRACT

It is difficult to achieve automatic visual detection of aluminium profile surface defects (APSD) owing to their various categories, irregular shapes, random distribution, and unbalanced samples. Utilising the attention mechanism, the unified detection method attempts to address these challenges for both common and rare defects. We formulate our method as a variant of few-shot learning to recognise the common and rare defect categories. First, a category representation network is applied to extract common category feature maps (CCMs). Second, an attention module is proposed to generate the proposal feature maps (PMs) of each rare category. Third, rare category feature maps (RCMs) are transformed from the CCMs under the guidance of the PMs. Finally, the scores of each category are obtained through the spatial pooling of both CCMs and RCMs. Experimental results on our constructed dataset show that our method is effective and outperforms the state-of-the-art methods.

1. Introduction

Inspecting the surface quality of aluminium profiles is extremely important in their production process. The quality and application scenarios of the aluminium profile depend to some extent on its appearance and texture. The manual detection, with workers' naked eyes, cannot satisfy the rapid and intelligent requirements of the modern aluminium production line. Vision-based automatic detection is therefore of great significance. Numerous excellent methods have been proposed to detect surface defects in other fields, such as steel [1, 3, 6, 18], fabrics [19, 24], and precision devices [28].

In recent years, some practices addressed the automatic detection of aluminium profile surface defects (APSD) [2, 10, 13]. These methods used traditional image processing algorithms, such as the variation of bright and dark view fields. Actually, they can only be applied to the aluminium plate, not to profiles with complex structures. Regarding this, some excellent solutions have been proposed in a competition held by Ali Cloud. Based on the deep convolution neural network (DCNN) technology, these methods have good performance regarding common defects, but not for both common and rare defects.

Two properties of both the common and rare defects are against vision-based automatic detection. One is the number of different categories. There are more than a dozen defects caused by physical and chemical actions, such as lacquer bubbles, bruises, and coating cracks. These have irregular shapes, random distribution, and similarity effects. The other property is the unbalanced probability of the category. Unbalanced samples are not conducive to learning methods. Fig. 1 shows examples of partial APSD. The first row shows the common defects, which can be collected in a short term, whereas the second row shows the rare defects, which exist but appear less frequently in production.

The lack of samples leads to the inability of the existing methods to detect rare defects. In general, a DCNN needs large samples to prevent over-fitting during training. Nevertheless, it is extremely difficult to collect hundreds of samples of the rare defects. How to detect the rare categories with only a few samples? To address this issue, researchers have proposed some methods that can recognise both the common categories with large samples and the rare categories with few samples.

Those methods focus on finding parameters that map the vectors with the category identification into the weight of the rare categories [4, 5, 9]. Although promising, it is possible that these solutions do not perform well in our task as it is a variant of few-shot learning with common and rare categories. All the rare categories are recognised simultaneously in our task. However, several rare categories, for example 5 or 20, are picked out randomly in traditional few-shot learning with common and rare categories. Besides, our task still confronts three difficulties. First, some defects are very similar and difficult to recognise, as shown in Fig. 2. The method needs a strong feature extractor to distinguish the differences among similar defects. Second, there is an extreme unbalance between the common and rare defect samples. How to integrate them reasonably? Third, the samples of some common defects are

* Corresponding authors.
E-mail addresses: unkechen@gmail.com (K. Song), yanyh@mail.neu.edu.cn (Y. Yan).

https://doi.org/10.1016/j.optlaseng.2019.105936
Received 26 August 2019; Received in revised form 31 October 2019; Accepted 1 November 2019

Please cite this article as: D. Zhang, K. Song and J. Xu et al., Unified detection method of aluminium profile surface defects: Common and rare defect categories, Optics and Lasers in Engineering, https://doi.org/10.1016/j.optlaseng.2019.105936

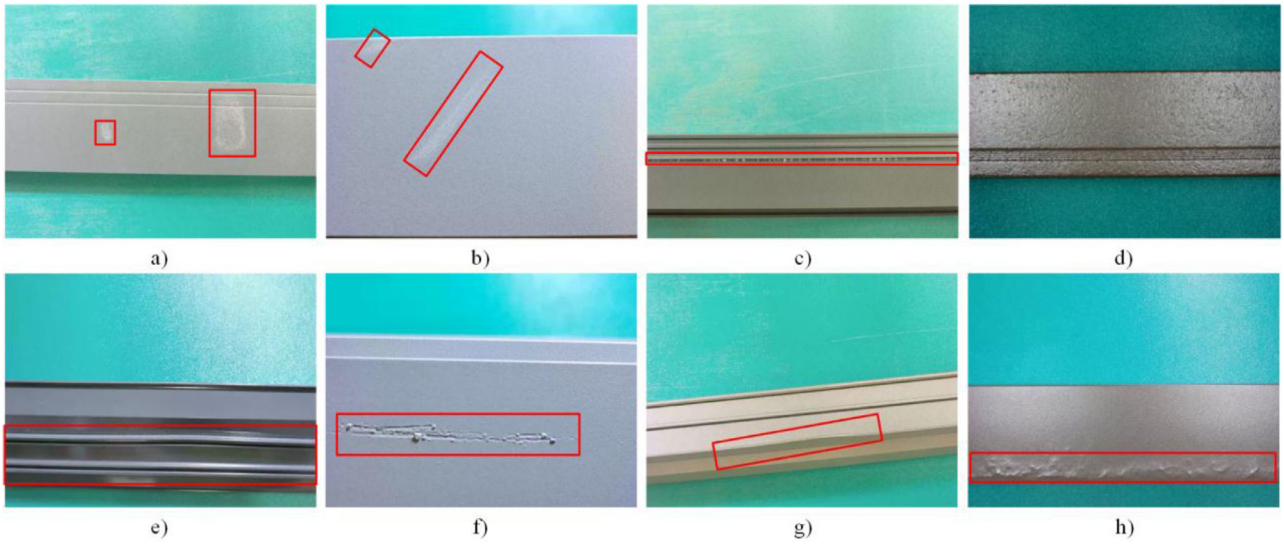

Fig. 1. Examples of partial APSD.

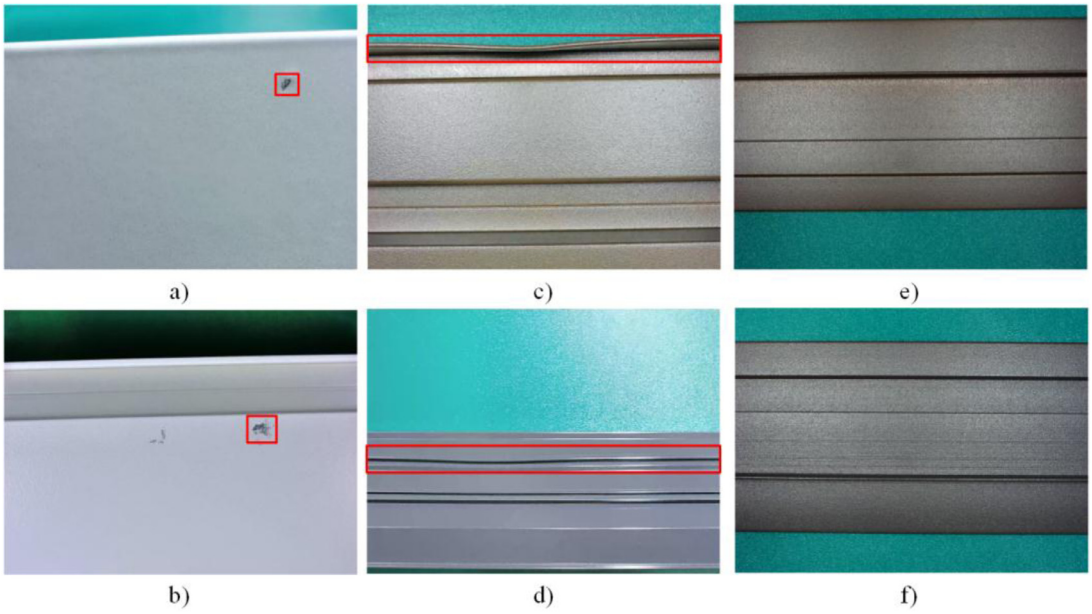

Fig. 2. Examples of similar defects.

insufficient, as there are only a few dozens. How to further enhance data is also a difficult point.

In this paper, we propose a unified method that is able to detect not only the common defects but also the rare defects of the APSD. Our method realises an unified detection of all types of defects at the same time. The method adopts 1) a strong category representation network based on ResNet-50 to extract the common category feature maps (CCMs), 2) an attention module to generate the proposal feature maps (PMs) that transform the CCMs into the rare category feature maps (RCMs), and 3) a combination approach of data enhancement with resizing, cropping, and random rotating. Our contributions can be summarised as follows:

- A unified method recognising both common and rare categories is proposed to detect the common and rare defects of the APSD at the same time.
- An attention module is developed with both channel and spatial attention branches to generate quality PMs, which can significantly improve the prediction accuracy of the rare defects of the APSD.
- An APSD dataset is constructed with 11 common defects and 6 rare defects, which evaluates comprehensively the performance of all methods.
- Experimental results show that our method achieves good performance on our constructed dataset, obtaining a good balance between common and rare defects.

The rest of the paper is structured as follows: in the next section, we present related studies on the few-shot method, attention mechanism, and pooling layer. Section 3 elaborates our proposed method in detail. Section 4 presents the experimental results and performance analysis. Finally, the conclusion is drawn in Section 5.

2. Related works

2.1. Few-shot with common and rare categories

Most existing few-shot methods only recognise the rare categories [7, 8, 29, 30]. In the cases where both common and rare categories need to

be recognised simultaneously, some models have also been proposed. Matthias [4] proposed a probabilistic model to establish the map between the weights of common and rare categories. Qiao [9] employed a set of parameters transforming the feature vectors into weights of categories. Gidaris [5] introduced the attention mechanism to select the weights of the feature vectors at the few-shot learning. These methods attempt to optimise the parameters that map the vectors with category identification into the weight of rare categories.

Inspired by this, our method is similar to that in [5] with the attention module. In contrast to previous models, there are two main differences. First, our attention module generates the fixed PMs of each rare category, not the weights of the feature vectors. The other difference is that our method replaces the full connection layer with the spatial pooling layer.

2.2. Attention mechanism

A comprehensive summary of the attention mechanism has been presented in [11, 12, 33]. Here, we discuss various methods based on self-attention mechanism to improve the performance of the DCNNs. The residual attention network (ResNet) [16] utilises an hourglass module to generate 3D attention maps for intermediate features. The pixel-wise contextual attention network (PiCANet) [21] and non-local block [22] employ the context of each pixel to construct an attended contextual feature or a filter to generate spatial attention maps. Similarly, the attention-based dropout layer (ADL) [14] is also a spatial attention module without additional trainable parameters and with only two hyper-parameters. Squeeze-and-excitation (SE) [15] applies attention to the channels by assigning different weights to each channel. The bottleneck attention module (BAM) [26], convolutional block attention module (CBAM) [23], and multi-source attention module (MAM) [17] combine the spatial attention and the channel attention in different ways (series connection, parallel connection, or mixed connection). Likewise, our attention module is composed of the above two parts, similar to the structure of the MAM. However, the outputs of our module are attention maps, while that of the MAM is an attention feature vector.

2.3. Pooling layer

It is necessary to leverage the pooling layer in deep learning networks as it not only integrates the high-level semantic features but also accelerates the calculation speed and prevents over-fitting. Maximum pooling and average pooling are applied to almost all deep learning networks. Based on maximum pooling, some variant pooling layers [20, 31, 25] are proposed, whose basic idea is still to select extreme values as representatives. Their difference is the way of selecting extreme values – some max or min values instead of one max value. Besides, the second-order pooling [27] is utilised in semantic segmentation networks. Some learnable pooling layers [32, 34, 35] are proposed based on singular value decomposition, eigenvalue decomposition, or global Gaussian distribution theorems. In our method, two types of pooling layers are used. One is a variant of maximum pooling. The other is an average pooling among channels. These details are introduced in Section 3.

3. The proposed method

3.1. Task statement

The engineering task we want to achieve is the surface defect detection in aluminium profiles with complex structure. We need to recognise 17 types of defects (11 common defects and 6 rare defects) in our constructive dataset correctly. Abstractly, a given dataset D consists of D_α and D_δ. In $D_\alpha = \{x_i^\alpha, y_i^\alpha | y_i^\alpha \in \{0, 1, \dots k_\alpha - 1\}\}_{i=1}^{k_\alpha}$, there are k_α categories with a large number of labelled samples called common categories. Similarly, in $D_\delta = \{x_i^\delta, y_i^\delta | y_i^\delta \in \{k_\alpha, k_\alpha + 1, \dots k_\alpha + k_\delta - 1\}\}_{i=1}^{k_\delta}$, there are k_δ categories called rare categories, and each category has one or few samples. Our task is to provide a method that can adequately recognise both the common and rare categories by learning from the training dataset.

3.2. Method overview

A unified classification method called UCR (Fig. 3) is proposed to detect both the common and rare defects of the surface of aluminium profiles. It consists of three sub-networks: one is the category representation network, which can extract the feature maps of each common category. The other is the rare category transfer network with the self-attention mechanism, which can transfer the CCMs to RCMs under the guidance of PMs, which are generated by the self-attention module. The remaining sub-network is the spatial pooling module, which extracts the category similarity from the representative regions of both CCMs and RCMs. In the following sections, the structure and the principle of each sub-network are introduced in detail.

3.3. Category representation network

The network can be denoted as $s_c = C(x)$, where x is input images. The output s_c is of k_α-channel maps, and each channel represents the feature map of a category. It consists of a) a feature extractor that extracts a d-channel feature maps (FMs) from an input image, b) a multi-map layer that transfers the FMs into the $k_m \times k_\alpha$-channel FMs, and c) a category-map pooling layer that combines the k_m maps of each category

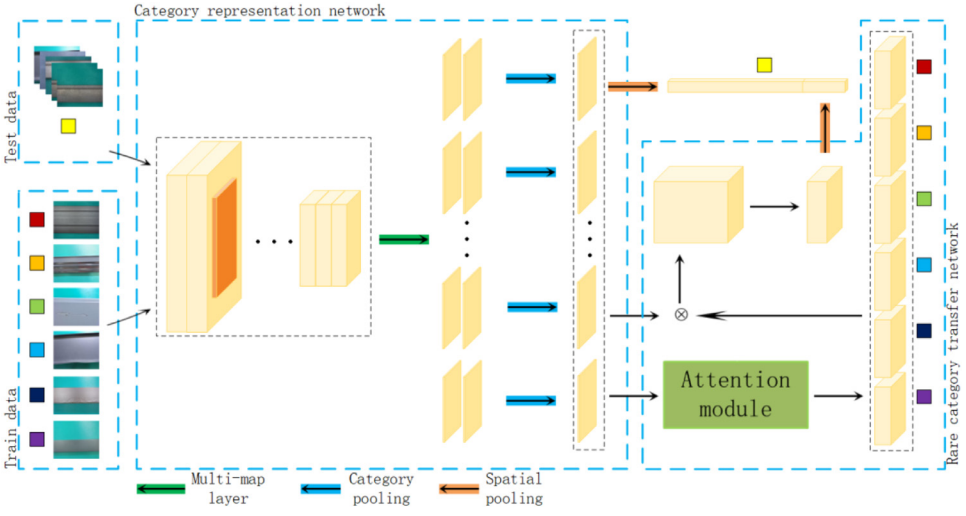

Fig. 3. Overview of our method.

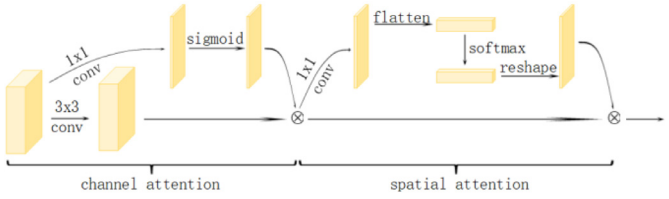

Fig. 4. Detail of the attention module architecture.

into a category map s^i_{base} by the average pooling. Obviously, there is no learnable parameter and only one super-parameter k_m, which aims at specialising in different category-specific features in the pooling layer.

Specifically, the feature extractor plays the role of the general filter by removing the background interference and refining the semantic information. It is very important because invalid feature representations cannot be classified correctly, regardless of how powerful the classifier is. We decide to select it from the publicly released successful baseline networks. The ResNet (removing the global average pooling and fully connected) is chosen as the backbone by comparing the parameters (0.85 M vs. 138 M) of the ResNet and VGG with similar depths (ResNet-50 vs. VGG16) [27]. It has been applied to image classification, object detection, and semantic segment. The pre-trained model is initialised on ImageNet dataset and fine-tuned on our relatively small defect dataset.

The multi-map layer transfers the feature maps learned by the last convolution layer of the feature extractor into $k_m \times k_\alpha$-channel feature maps encoding k_m-channel feature maps per category through a 1×1 convolution. This layer generates a combination for varied specific features, such as shape and texture. When k_m is set as 1, this layer degenerates into a category representation layer.

The category pooling layer combines the k_m maps of each category into a category feature map by average pooling among channels. It can be represented by the following formula: $s^i_c = \frac{1}{k_m}\sum_{j=1}^{k_m} m^i_j$, where m^i_j is the j^{th} feature map of the category i and $i = \{0, 1, \ldots, k_\alpha - 1\}$. Apparently, s_c can be expressed as $\{s^0_c, s^1_c, \ldots, s^{k_\alpha-1}_c\}$.

3.4. Rare category transfer network

After the category representation network, the rare category transfer network $s_r = R(s^{test}_c | s^{train}_c)$ is proposed to learn the RCMs of each rare category. And the output s_r is of k_δ-channel maps. The network transfers the CCMs s^{test}_c into the RCMs under the guidance of PMs. The PMs are generated by the attention module $A(s^{train}_c|.)$, where s^{train}_c denotes the average CCMs of all samples of each rare category.

Specifically, our attention module includes a channel attention branch and a spatial attention branch, as shown in Fig. 4. At the channel attention branch, we use a 1×1 convolution followed by a sigmoid function on the feature map to generate a coarse salient map M_s and 3×3 convolutions to obtain the proposed channel-weight maps M_q for each channel. Then, the channel attention maps M_c of each channel are obtained by a dot product between $M_s$and each channel of M_q. After that, the spatial attention branch utilises a 1×1 convolution following a normalised function softmax to generate the spatial proposed mask M_p. Similarly, the final PMs are obtained by dot products between M_p and each channel of M_c. In our method, the output of the attention module s_a consists of k_δ group of PMs, i.e., $s_a = \{s^{k_\alpha}_a, s^{k_\alpha+1}_a, \ldots, s^{k_\alpha+k_\delta-1}_a\}$. Each group of PMs is generated by the CCMs of the corresponding category.

Following the attention module, the rare category multi-feature maps s_{rm} are calculated by $s_c \cdot s^i_p$ $(i = k_\alpha, k_\alpha + 1, \ldots, k_\alpha + k_\delta - 1)$, where $\cdot$ is the dot products operator. s_{rm} is of $k_\alpha * k_\delta$-channel feature maps. Finally, a 1×1 convolution layer transfers s_{rm} into s_r with the k_δ-channel (a single map per category).

3.5. Spatial pooling module

There are no learnable parameters and only three hyper-parameters in the module P(.|.). It selects relevant regions within the maps to support predictions. It can be represented by the following formula as

$$\begin{aligned} s_s = \max \frac{1}{k_{top}} \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}s_{ij} + \gamma(\min \frac{1}{k_{low}} \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij}s_{ij}) \\ s.t. p_{ij} \in \{0,1\} q_{ij} \in \{0,1\} \\ \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} = k_{top}, \quad \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij} = k_{low} \end{aligned} \tag{1}$$

where s_{ij} denotes the value of feature map s at (i, j) with the size of m*n. k_{low}, k_{top} and γ are all hyper parameters. The maximum and minimum, as the most extreme value, are important for good results but carry different information. We consider that the effective number of both most extreme values should be the same, but the maximum is more useful for the classification. Therefore, set k_{low} and k_{top} as 2 and γ as 0.5. Eq. (1) can be put into the following form:

$$\begin{aligned} s_s = \frac{1}{4}\left(\max \sum_{i=1}^{n}\sum_{j=1}^{m} 2p_{ij}s_{ij} + \min \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij}s_{ij}\right) \\ s.t. p_{ij} \in \{0,1\} q_{ij} \in \{0,1\} \\ \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} = 2, \quad \sum_{i=1}^{n}\sum_{j=1}^{m} q_{ij} = 2 \end{aligned} \tag{2}$$

4. Training

In our method, our network mainly includes three sub-networks, whereas there are no learnable parameters in the spatial pooling module. Hence, only the category presentation network C(x) and the rare category transfer network $R(s^{test}_c | s^{train}_c)$ need to be trained. To learn the parameters well, we a) split the training process into two phases and train the two sub-networks separately, b) employ the common categories as negative samples at the sub-network *R* training phase, and c) select the cross entropy loss function at both phases.

At the sub-network C training phase, the network does not contain the sub-network R. Therefore, the training process can be seen as a regular classification task training. Only the common defect images (i.e. D_α) are utilised to train it. At the sub-network R training phase, the parameters of the sub-network C are fixed and only the parameters of the sub-network R are trained, as expressed in Algorithm 1. Compared with the regular classification task training, there are two key different points: one refers to the categories of each batch. Although only the sub-network R is trained, there are not only all the rare categories, but also some common categories as "negative categories" in each batch. The other point is the internal sequence of each batch. Different from the random shuffling of samples in regular classification training, they must be sorted according to the order of label values, i.e., $\{x^{k_\alpha}, x^{k_\alpha+1}, \ldots, x^{k_\alpha+k_\delta-1}, \ldots\}$.

5. Experiments

5.1. Dataset

The APSD dataset is constructed by us. The raw data come from a competition [1][1] hosted by Alibaba. In the raw data, there are 30 defect categories, of which 11 categories have more than 50 samples, 6 categories have between 50 and 11 samples, and 14 categories have less than 11 samples. In theory, our UCR method can detect all these defects. However, it is unable to carry out an effective evaluation for the categories with less than 11 samples. In this situation, these defect categories

[1] [1] https://tianchi.aliyun.com/competition/entrance/231682/introductionntrance/231682/introduction

Algorithm 1
training of the sub-network R.

Input: $D \leftarrow D_{train}(k_{com} \leftarrow k_\alpha,\ k_{rare} \leftarrow k_\delta)$ ➢ k_α is the number of the common categories
$D_t \leftarrow D_{train}(k_{com} \leftarrow 0,\ k_{rare} \leftarrow k_\delta)$ ➢ k_δ is the number of the rare categories
Output: trained R
for k_l in $[k_\alpha,\ k_\alpha + 1,\ \ldots,\ k_\alpha + k_\delta - 1]$ do ➢ randomly select N_l samples with the label k_l
$(x_t,\ y_t) \leftarrow D_t(k_l,\ N_l)$
$s_t^{k_l} \leftarrow C(x_t)$
$s_c^{k_l} \leftarrow \mathrm{mean}(s_t^{k_l})$ ➢ the mean of the feature maps $s_t^{k_l}$
end for
$s_c^t \leftarrow \{s_c^{k_\alpha}, s_c^{k_\alpha+1}, \ldots, s_c^{k_\alpha+k_\delta-1}\}$
for 1 to epoch do ➢ randomly select N samples from k_α common categories
$D \leftarrow D(k_{com} \leftarrow (k_\alpha,\ N),\ k_{rare} \leftarrow (k_\delta,\ M_\delta),\ K)$ ➢ M_δ samples per category from k_δ rare categories
for 1 to K do
$(x,\ y) \leftarrow \mathrm{next}(D)$
$s_c \leftarrow C(x)$
$s_r \leftarrow R(s_c|s_c^t)$
$s \leftarrow \mathrm{connect}(s_c,\ s_r)$ ➢ update loss
$s_s \leftarrow P(s)$ ➢ w_r denotes the parameters of sub-network R
$J \leftarrow \mathrm{loss}(s_s,\ y)$
$w_r \leftarrow \mathrm{updata}(\partial J/\partial w_r)$
end for
end for
return R

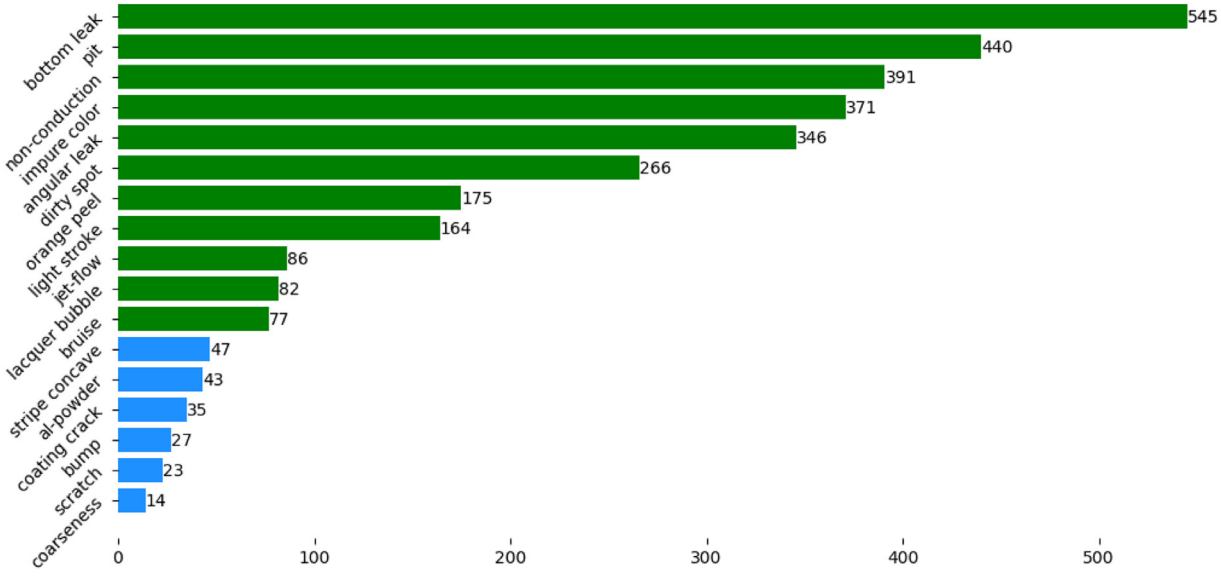

Fig. 5. Number of samples of all the defects.

are removed. Then, the defects with more than 50 samples are regarded as common defects and the rest are the rare defects. In other words, the dataset includes 17 defect categories, of which 11 categories are the common defects and 6 categories are the rare defects. Fig. 5 shows the detailed defect categories and the numbers in the dataset. The samples of each rare category are split with 5, 6, and the rest are samples for training, validation, and testing, respectively. Similarly, the samples of each common category are split into 10, 10, and the rest are samples for validation, testing, and training. Our constructive dataset is available at https://pan.baidu.com/s/1GdfN2a2Njni_ZpTpudK5Ag and key: 5211

5.2. Implementation detail

At the sub-network C training phase, we use the Adam optimiser with batch size 22, which are randomly chosen from the common defects, and learning rate 0.001, which is divided by 2 every 10 epochs. At the sub-network *R* training phase, each batch contains 6 rare defect images (one per category) and 3 common defect images. In this phase, we use the Adam optimiser and learning rate 0.0005, which is divided by 2 every 10 epochs. Each epoch contains 100 batches. All the training images are resized to 400 × 400.

All our algorithms are implemented in Python and executed on a PC with an Intel(R) Core (TM) i7-7700QM 3.6 GHz processor, 8 GB RAM, and a 12GB Nvidia TITAN Xp. We used PyTorch to implement all codes.

5.3. Results and comparison

To evaluate our UCR method, we carry out the experiments on the above dataset. The accuracy rate is used to evaluate the performance of the models, which is calculated as i_c/i_t (i_c denotes the number of correctly categorised images and i_t denotes the number of test images). Fig. 6 displays the accuracy rates of both the common and rare defects by our method. It is clearly shown that our method has a good effect on the common defects. For the rare defects, some of them have a high accuracy rate, such as scratch and al-powder, whereas other rare defects have low accuracy rate, such as concave stripe and bump. We analyse the main causes of the low accuracy rate. One is that these rare defects

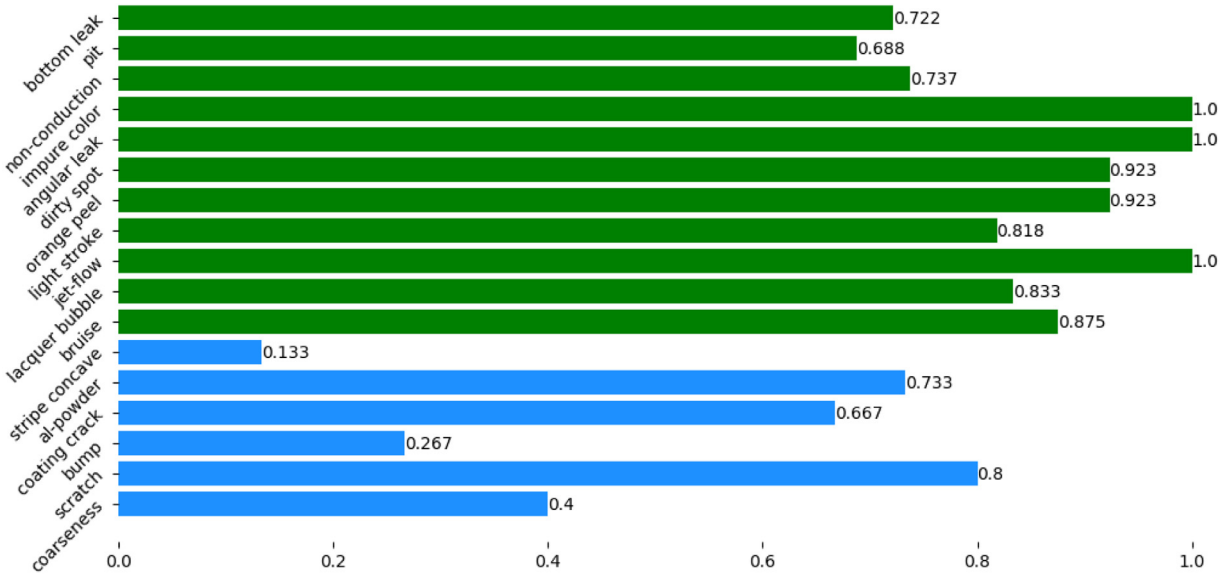

Fig. 6. Accuracy of our method on the above dataset.

Table 1
Accuracy rate of different methods on the above dataset.

method	ResNet	DFS	URC without att	UCR
common	0.9000	0.8717	0.8061	0.8606
Rare	0.1590	0.4267	0.3667	0.5000

are similar, such as concave stripe and bump. The other cause is that the training samples are not very representative, such as coarseness.

To evaluate the performance of our method, we compare our UCR method with ResNet [16], DFS [5], and UCR without the attention module on the above dataset. The results of accuracy rates are presented in Table 1. We can observe that for the common defects, ResNet has the best detection effect. However, it is unable to detect the rare defects, having an accuracy rate of only 15.92%. DFS has a slightly better detection effect for the common defects than our method, but only by approximately 1%. However, our method has obvious advantage in the detection of the rare defects. The performance of our UCR method is 7% higher than that of DFS in detecting the rare defects. Besides, to explain the effectiveness of the attention mechanism, we conduct the experiment with UCR without the attention module. From the results, we can observe that the attention module significantly improves the detection of both the common and rare defects.

Table 2
Accuracy rate of different attention modules on the above dataset.

the attention module	ResNet	PiCANet	non-local block	SE	BAM	CBAM	Ours
common	0.8545	0.1515	0.3152	0.8667	0.6970	0.8364	0.8606
Rare	0.4889	0.4111	0.3444	0.2222	0.5333	0.4778	0.5000

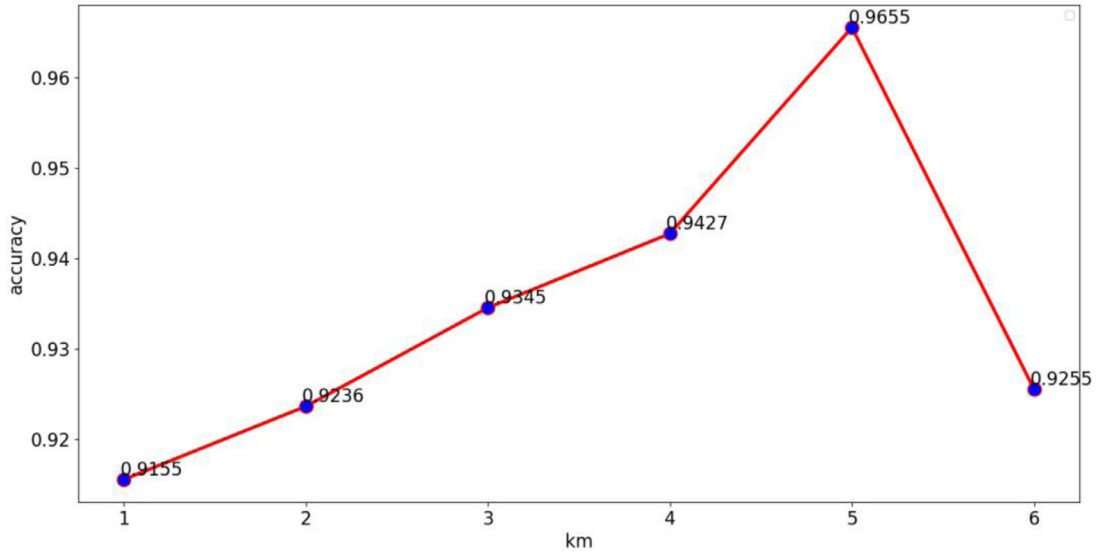

Fig. 7. Accuracy of different k_m on the above dataset.

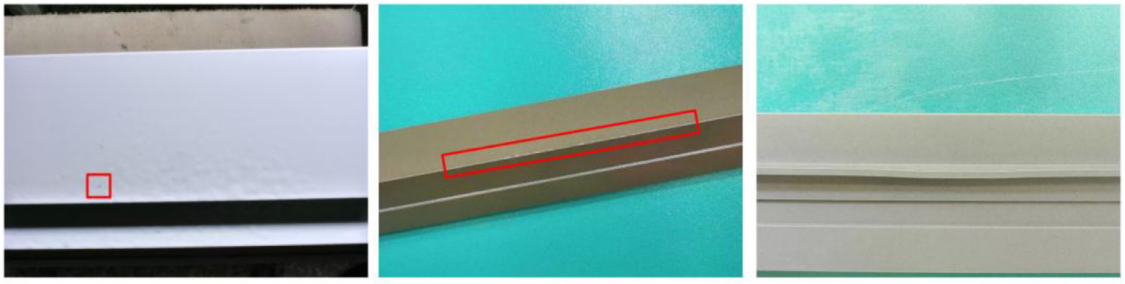

Fig. 8. Examples of failure cases.

Table 3
Accuracy of different numbers of negative samples per batch.

number	2	3	4	6
common	82.424	86.06	90.91	90.91
rare	40.00	50.00	44.44	43.33

6. Discussion

6.1. Rationale of designing the attention module

The design of the attention module for a specific view is based on the assumption that the features of the common and rare categories are correlated. Actually, the function of the attention module is to map the feature maps based on common categories into the feature maps based on rare categories.

To prove the effectiveness of our attention module, we compare it with Resnet [16], PiCANet [21], non-local block [22], SE [15], BAM [26], and CBAM [23] on the above dataset. The results of accuracy are presented in Table 2, and the top two results are shown in red and green colour, respectively. It can be observed that our attention module is the second-best regarding common and rare defects and is slightly worse than SE and BAM, respectively. However, SE is much worse than our module on rare defects. Similarly, BAM is much worse on common defects. Hence, in general, our attention module is the best and most appropriate for our method.

6.2. Parameter k_m of multi-map layer

The multi-map layer is utilised to generate a combination of category feature maps. An appropriate value of k_m contributes to the effective extraction of the category feature maps and the accuracy of the classification. We conduct experiments with k_m increasing in turn from 1 on the above dataset. In Fig. 7 we can observe that the accuracy of classification is improved with the increase in k_m value. When k_m is increased to 6, the classification accuracy decreases. Therefore, we believe that 5 is the most appropriate value of k_m, which can make the combination of category feature maps representational and simple.

6.3. Variant maximum pooling vs. maximum pooling

For the spatial layer, the variant maximum pooling (Eq. (2)) is used instead of the maximum pooling. The reason is that we attempt to find the most appropriate representative value. To illustrate the significance of the variation, we compare it to the normal maximum pooling and obtain the results of 0.9655 vs. 0.9337 for the variant vs. normal on common defects of the above dataset. Obviously, the variant improves the classification accuracy by approximately 3% with better performance.

6.4. Number of negative samples per batch

At the sub-network R training phase, we pick randomly some common defect samples as negative samples of the rare defects in each batch. The number of the common defect samples affects the result of classification accuracy. We hypothesise that less negative samples can lead to over-fitting of training. On the contrary, more negative samples can disturb the effective feature extraction of the rare defects. To find a suitable number of negative samples, some comparative experiments are carried out and their results are presented in Table 3. When the number of common defect samples is set as 4 and 6, the accuracy of rare defects is greatly reduced, although the accuracy of common defects is slightly improved. Hence, the number is not adopted. The method obtains the best result for all the defects when it is 3 in each batch.

6.5. Method complexity

In our network, the learnable layers consist of all the convolution layers in the Resnet-50 and other 4 convolution layers. The method has a relatively low computational complexity during training. It takes about 5 h to train the network on the above dataset, and our network can classify images at 196 fps on our computer. The fast online testing suggests that our approach could be used in routine defect detection in industrial pipelining.

6.6. Failure case analysis

Although our method achieves promising results for the common and rare defects of the surface of aluminium profiles, it has a poor performance for defect categories such as pit and concave stripe. We attempt to analyse the reasons of the unsatisfactory detection. In addition to the similarity among the defects and, as mentioned above, the lack of representativeness of the training samples, we observe that there are several other reasons leading to classification errors: 1) the defect is too small, 2) the defect is too slight, 3) the intensity of illumination changes considerably. Some examples of these failure cases are shown in Fig. 8.

7. Conclusion

In this paper, a unified method is proposed to detect both the common and rare defects on the surface of aluminium profiles. An attention module is developed to promote the accuracy of the common and rare defects by providing PMs. We also construct an APSD dataset with common and rare defects. Experimental results show that our method has a good performance on the dataset. In the future work, we will study how to obtain more robust PMs and how to extend our method for detection in other industrial products, such as rails.

Decleration of Competing Interest

None.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (51805078, 51374063), the National Key Research and Development Program of China (2017YFB0304200), the Fundamental Research Funds for the Central Universities (N170304014), and the China Scholarship Council (201806085007).

References

[1] He Y, Song KC, Meng QG, Yan YH. An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE Trans Instrum Meas 2019;99:1–12.
[2] Huang XQ, Luo XB. A real-time algorithm for aluminum surface defect extraction on non-uniform image from CCD camera. In: Proceedings of the 2014 International Conference on Machine Learning and Cybernetic(ICMLC); 2014. p. 556–61.
[3] He Y, Song KC, Dong HW, Yan YH. Semi-supervised defect classification of steel surface based on multi-training and generative adversarial network. Opt Lasers Eng 2019.
[4] M. Bauer, M. Rojas-Carulla, Świątkowski, J.Bartłomiej, et al. Discriminative k-shot learning using probabilistic models. 2017; https://arxiv.org/abs/1706.00326.
[5] S. Gidaris, N. Komodakis. Dynamic few-shot visual learning without forgetting. 2018; https://arxiv.org/abs/1804.09458.
[6] Di H, Ke X, Peng Z, et al. Surface defect classification of steels with a new semi-supervised learning method. Opt Lasers Eng 2019;117:40–8.
[7] Koch G, Zemel R, Salakhutdinov R. Siamese neural networks for one-shot image recognition. In: Proceedings of the 32nd International Conference on Machine Learning(ICML), 37; 2015.
[8] Ravi S, Larochelle H. Optimization as a model for few-shot learning. International Conference on Learning Representations(ICLR); 2017. p. 1–11.
[9] Qiao S, Yuille A. Few-shot image recognition by predicting parameters from activations. arXiv:1706.03466. 2017.
[10] Zhang HL, Qi XG, Li XT. Research on key technology of cold-rolled aluminum plate surface defect detection system. Appl Mech Mater 2013;433-435:915–18.
[11] Wang F, Tax DMJ. Survey on the attention based RNN model and its applications in computer vision. https://arxiv.org/abs/:1601.06823. 2016.
[12] Lee JB, Rossi RA, Kim S, et al. Attention models in graphs: a survey. https://arxiv.org/abs/:1807.07984. 2018.
[13] Huang XQ, Luo XB, Wang RZ. A real-time parallel combination segmentation method for aluminum surface defect images. In: Proceedings of the 2015 International Conference on Machine Learning and Cybernetic(ICMLC); 2015. p. 544–9.
[14] Choe J, Shim H. Attention-based dropout layer for weakly supervised object localization. Proc IEEE Comput Vis Pattern Recognit (CVPR) 2019.
[15] Hu J, Shen L, Sun G. Squeeze-and-Excitation networks. https://arxiv.org/abs/:1709.01507. 2017.
[16] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. arXiv:1512.03385. 2015.
[17] Zeng Y, Zhuge Y, Lu H, et al. Multi-source weak supervision for saliency detection Proc IEEE Comput Vis Pattern Recognit (CVPR); 2019.
[18] Fu GZ, Sun PZ, Zhu WB, Yang JX, Cao YL, Yang MY, et al. A deep-learning-based approach for fast and robust steel surface defects classification. Opt Lasers Eng 2019;121:397–405.
[19] Tsang CSC, Ngan HYT, Pang GKH. Fabric inspection based on the ELO rating method. Pattern Recognit 2016;51:378–94.
[20] Lin D, Shen X, Lu C, Jia J. Deep lac: deep localization, alignment and classification for fine-grained recognition. Proc IEEE Comput Vis Pattern Recognit (CVPR) 2015:1666–74.
[21] N. Liu, J. Han, M.H. Yang. PiCANet: learning pixel-wise contextual attention for saliency detection. 2017; https://arxiv.org/abs/:1708.06433.
[22] X. Wang, R. Girshick, A. Gupta, et al. Non-local neural networks. 2017; https://arxiv.org/abs/:1711.07971.
[23] S. Woo, J. Park, J.Y. Lee, et al. CBAM: convolutional block attention module. 2018; https://arxiv.org/abs/:1807.06521.
[24] Ngan HYT, Pang GKH, Yung NHC. Automated fabric defect detection-a review. Image Vis Computing 2011;29(7):442–58.
[25] Zhang J, Bargal SA, Lin Z, et al. Top-down neural attention by excitation backprop. In: Eur Conf Computer Vis (ECCV); 2016. p. 543–59.
[26] Park J, Woo S, Lee JY, et al. BAM: bottleneck attention module. https://arxiv.org/abs/:1807.06514. 2018.
[27] Carreira J, Caseiro R, Batista J, et al. Semantic segmentation with second-order pooling. In: Eur Conf Computer Vis (ECCV); 2012. p. 430–43.
[28] Liu Y, Yu F. Automatic inspection system of surface defects on optical IR-CUT filter based on machine vision. Opt Lasers Eng 2014;55:243–57.
[29] J. Snell, K. Swersky, R.S. Zemel. Prototypical networks for few-shot learning. 2017; https://arxiv.org/abs/1703.05175.
[30] Vinyals O, Blundell C, Lillicrap T, et al. Matching networks for one shot learning. In: Proc Neural Inf Process Syst (NIPS); 2016. p. 3630–8.
[31] Durand T, Thome N, Cord M. WELDON: weakly supervised learning of deep convolutional neural networks. In: Proc IEEE Comput Vis Pattern Recognit (CVPR); 2016. p. 4743–52.
[32] Wang Q, Li P, Zhang L. G2DeNet: global gaussian distribution embedding network and its application to visual recognition. In: Proc IEEE Comput Vis Pattern Recognit (CVPR); 2017. p. 6507–16.
[33] S. Chaudhari, G. Polatkan, R. Ramanath, et al. An attentive survey of attention models. 2019; https://arxiv.org/abs/1904.02874.
[34] T.Y. Lin, A. Roychowdhury, S. Maji. Bilinear CNNs for fine-grained visual recognition. 2015; https://arxiv.org/pdf/1504.07889v5.pdf.
[35] Ionescu C, Vantzos O, Sminchisescu C. Matrix backpropagation for deep networks with structured layers. In: Proc IEEE Int Conf Comput Vis (ICCV); 2015. p. 2965–73.