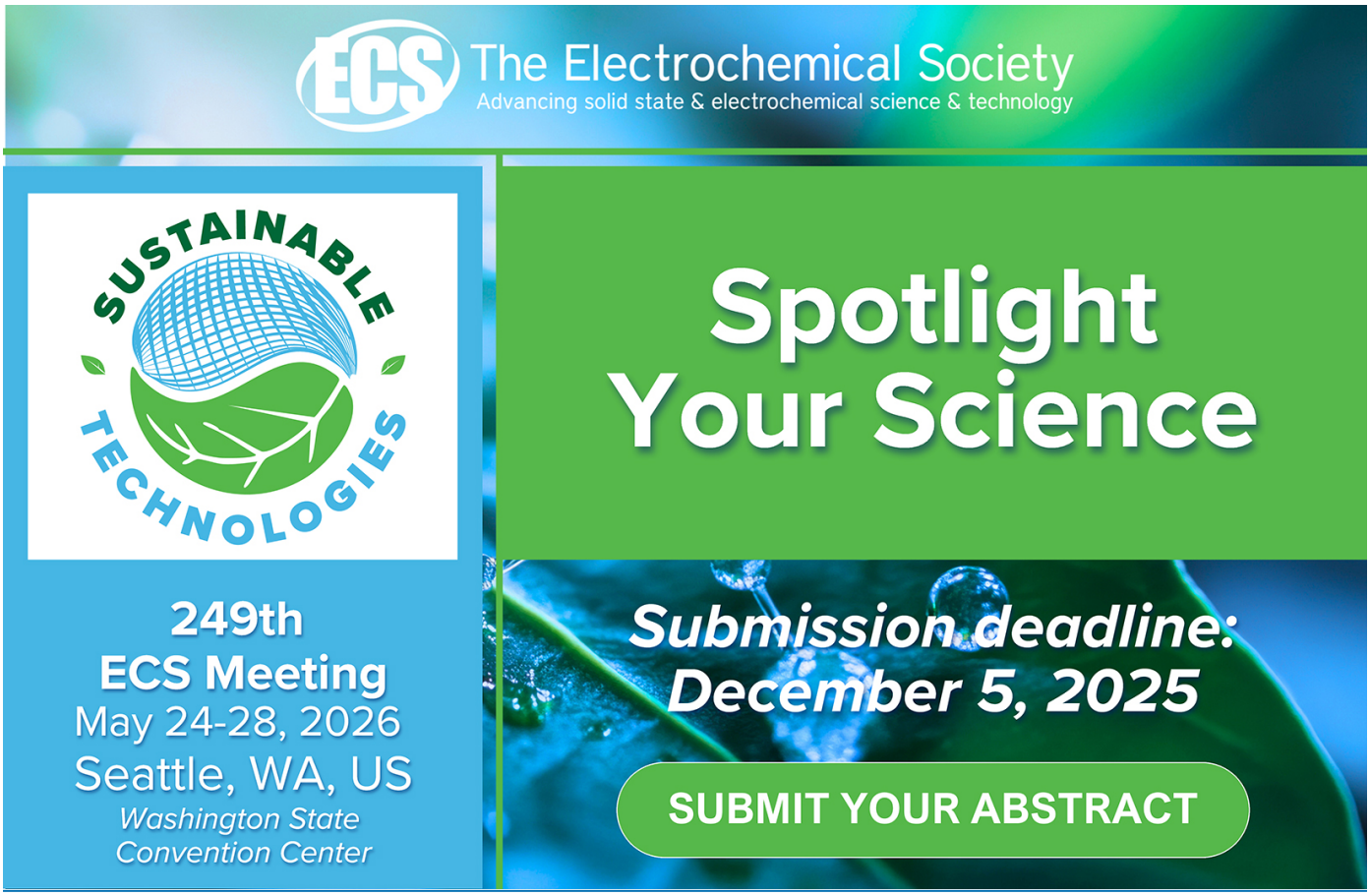
Measurement Science
and Technology

PAPER

Truncated and integrated class activation maps for weakly supervised defect detection

To cite this article: Hang-Cheng Dong *et al* 2025 *Meas. Sci. Technol.* **36** 066138



IOP Publishing Measurement Science and Technology
Meas. Sci. Technol. **36** (2025) 066138 (12pp) https://doi.org/10.1088/1361-6501/adcf3d

Truncated and integrated class activation maps for weakly supervised defect detection

Hang-Cheng Dong, Bingguo Liu, Dong Ye and Guodong Liu*

School of Instrumentation Science and Engineering, Harbin Institute of Technology, Harbin 150001, People's Republic of China

E-mail: hunsen_d,lgd@hit.edu.cn

Received 19 September 2024, revised 23 February 2025
Accepted for publication 22 April 2025
Published 18 June 2025

Abstract

Deep learning is now widely used for detecting surface defects, which is crucial for automated quality control in industries. However, getting lots of accurate labeled data is tough, and this slows down the progress of smart defect detection. To tackle this, we suggest using weakly supervised semantic segmentation (WSSS) methods, especially those based on class activation maps (CAMs). One issue with CAM is that the feature maps from the last layer of the neural network do not have high enough resolution. We want to create feature maps with more detail that can give us better semantic information. We take a new look at the semantic information in the early feature maps, finding that they have fine details but also mix in a lot of noise that is not relevant to our target. To fix this, we propose a simple way to reduce noise by cutting off positive gradients. This idea can be added to other CAM methods to help them get better CAMs. A large number of WSSS experiments were conducted on defect detection datasets. The results from these experiments consistently show that our method is effective for finding defects.

Keywords: weakly-supervised semantic segmentation, class activation maps, defect detection

1. Introduction

Surface defect detection is a crucial part of quality inspection and the final and most effective means to ensure product quality [1]. When substandard products enter the market, they not only affect the consumer experience but can also lead to safety accidents in severe cases. Traditional inspection methods rely on manual labor, which is time-consuming, inefficient, and highly subjective. With the rapid development of machine vision technology, image processing and computer vision techniques have been applied to defect detection tasks, initially achieving automated inspection. However, machine vision technology also has its limitations, as it finds it difficult to adapt to complex background environments and the variable appearances of products [2]. Moreover, due to the significant differences in the geometric characteristics of defects themselves, such as cracks and pits with diverse morphologies [3], classical machine vision methods struggle to extract key features.

With the advancement of artificial intelligence technology, detection methods based on deep learning have been empirically proven to effectively address the aforementioned challenges [4]. Compared to the manually designed features of traditional machine vision [5, 6], data-driven deep learning methods can extract more robust and generalized features. Consequently, defect detection methods based on convolutional neural networks have been widely applied, including tasks such as object detection and semantic segmentation. [7] studied a method for detecting defects in power line insulators using aerial imagery. The study proposed a two-stage cascaded

* Author to whom any correspondence should be addressed.

defect detection scheme, first using the first network to detect all potential defective insulators in the image, and then using the second network to detect and locate missing insulator caps in the cropped images. [8] proposed a network for steel surface defect detection. By fusing multiple pyramid feature maps with different resolutions, it retains texture information that may be lost due to downsampling, enhancing the detection capability for small defects.

However, the surface defect inspection task has encountered a number of challenges in terms of data acquisition. Firstly, the data accumulation process is lengthy, which is tied to the production cycle and the output rate of quality products. Collecting mature defect data from the production line is a slow process, and the time to accumulate a large dataset can be up to several years. Secondly, data labeling is not as efficient as it should be. Defect labeling requires the expertise of skilled workers, and tasks such as object detection and semantic segmentation are significantly more complex than image-level labeling tasks. Thirdly, data labeling is inherently challenging. The boundaries of certain defects are often blurred, which poses considerable challenges for pixel-level annotation. Furthermore, incorrect annotations can also seriously affect the performance of deep learning algorithms.

Weakly supervised learning (WSL) sheds light on reducing the manual labor required in deep learning. WSL does not require high-level labels such as semantic labels, and it is less dependent on the quantity of samples. The most widely used fundamental technology for applying WSL to tasks such as defect detection is class activation maps (CAMs). CAM is originally a method for interpretability in deep learning. CAM-based explanations generate heatmaps for CNN-based classification models where the highlighted regions have a higher probability of coinciding with the target object [9–11]. Therefore, the CAM-based approaches can enable object localization and segmentation under the supervision of image-level annotation, which greatly facilitates the development of weakly supervised target localization [12] and weakly supervised semantic segmentation (WSSS) [13, 14].

Nevertheless, most of the methods mentioned above involve only the feature maps of the last convolutional layer, whose spatial resolution is limited, resulting in a coarse localization and segmentation of the targets. A natural idea is whether it is possible to obtain semantic information about the target from shallow, high-resolution feature maps. As revealed by Relevance-CAM and LayerCAM, shallow feature maps do contain higher-resolution semantic information, albeit with more noise. FullGrad [15] goes a step forward by fusing gradient features from all convolutional layers. For the sake of brevity, we refer to the operation of fusing features from different layers as the layerization trick.

Inspired by the layerization trick, the question we want to determine is whether such numerous CAM variants can be implemented with it. Moreover, although the layerization trick enables better visual explanation, it is difficult to apply in WSSS if the noise cannot be suppressed, even if the feature resolution becomes higher.

To verify the issues described above, we investigate the performance of the basic Grad-CAM in shallow layers. Surprisingly, with only a gradient rectification, the performance of Grad-CAM can be improved to be comparable to or even surpassed by state-of-the-art methods. We further validate the gradient rectification and find that the method can also improve the performance of FullGrad and LayerCAM, which is termed the truncation trick. Generally, our main contributions are threefold:

- We propose LTGrad-CAM, which shows that global weights can still exploit shallow semantic information and are quite easy to deploy to any off-the-shelf CNN model.
- We summarize for the first time the layerization trick and the truncation trick. Moreover, we propose that these two techniques can be used as plug-ins for many gradient-weight CAM-based methods, which can provide fine-grained visual explanations and substantially improve the performance of weakly supervised tasks.
- We propose a weakly supervised detection process for surface defect detection, which only requires image-level labels to achieve high-resolution defect semantic segmentation.

2. Related works

2.1. Fully-supervised defect inspection

Deep learning-based defect detection methods can be categorized into classification, object detection, and semantic segmentation, with the latter two having a broader range of applications [1, 4]. Chen *et al* [16] studied convolutional neural networks for wafer defect classification and employed the attention spatial pyramid pooling module, which enhanced the accuracy of the CNN. However, classification networks cannot accurately obtain the location of defects, thus object detection methods have stronger applicability. Luo *et al* [17] proposed a decoupled two-stage defect detection algorithm for detecting surface defects on flexible printed circuit board. The proposed decoupled two-stage object detection framework achieves decoupling of localization and classification tasks by generating two specific features for the localization and classification tasks. Zheng *et al* [18] optimized the YOLOv3 network for the detection of tire defects in x-ray images. Furthermore, semantic segmentation models [5] have transformed the task of surface defect detection into a semantic segmentation and even instance segmentation issue, distinguishing defects from normal areas. By segmenting defect areas pixel by pixel, one can not only pinpoint the location and category of defects but also calculate other geometric properties such as length, area, and central position. With accurate semantic segmentation, it becomes feasible to assess the severity of defects, which is instrumental in controlling production processes and enhancing manufacturing efficiency. Dong *et al* [19] introduced PGANet, which facilitated the propagation of valid information from low-resolution fusion feature maps

to high-resolution fusion feature maps. PGANet also incorporated a boundary refinement module within its framework to enhance the prediction of object boundaries. Su *et al* [20] proposed a segmentation network guided by shape information, which integrated the shape knowledge of fasteners into the segmentation network, thereby improving the segmentation performance on images of the railway track fastener. Full supervision semantic segmentation is also applied in the field of chip defect detection. In [21], a global information fusion module is proposed to improve the performance of the segmentation model. This work also proposes synthetic sample generation and loss function adjustment solutions to address issues such as sample imbalance and limited sample quantity.

2.2. WSSS

WSL, in the context of defect detection tasks, generally refers to the use of only image-level labels to achieve tasks such as object detection and even semantic segmentation. This approach allows for the training of models with minimal annotation requirements, leveraging the availability of images that may not have detailed pixel-wise or bounding box annotations [22]. In early research, classification models were trained using multi-instance learning methods, and traditional image segmentation was relied upon to obtain foreground and background information [23]. However, this method of segmentation was found to be too rudimentary and slow-paced to meet the demands of industrial defect detection. An alternative approach involves the use of CAMs [9], which offers a more refined and efficient solution. Zhang *et al* [24] have extended CAM to the convolutional neural network architecture they designed, and by integrating model distillation technology, they have enhanced the detection capability for surface defects on glass. There have been many improvements to the CAM method [9–11, 15, 25, 26]. CAM takes advantage of the characteristic that the feature maps of the last convolutional layer have discriminative semantics. By assigning appropriate weights to each channel of the feature maps, their linear combination can be computed to obtain a heatmap that reflects the position of the target.

However, CAM has specific requirements for the structure of the model, i.e., a global average pooling layer that directly connects the classifiers. Grad-CAM [10] is a generalized version of CAM, employing the average of the gradients as the weights of the feature maps. The weights generated in Grad-CAM are considered to represent the importance of corresponding channels with respect to the target category. Then, Grad-CAM++ [26] reveals that using positive gradients can better indicate the features that have a positive impact on the prediction results. In addition, XGrad-CAM [27] proposes two axioms to correct the gradient weights for CNNs with only ReLU activations. Lift-CAM [28] and Relevance-CAM both use layer-wise relevance propagation [29] to calculate the weights of different channels, and Relevance-CAM [30] further finds that the problem of shattered gradients in shallow feature maps can be well alleviated. In addition to using the back-propagation gradient as a measure of importance, Score-CAM [25] and Ablation-CAM [31] use the change in the model output value to measure the weight of the corresponding feature maps. Unlike the global weights mentioned above, LayerCAM [11] suggests using back-propagated gradient matrices as local weights to highlight channel-wise feature maps. In summary, a large number of CAM variants focus on how to obtain a better weight assignment method.

These weakly supervised methods have also been widely applied in the field of defect detection in past practices. In [32], LayerCAM was used to generate rough localization of defects for monitoring the defect detection process of nuclear-fuel rod grooves. Zhang *et al* [33] directly used the CAM method to generate pseudo-labels for surface defects of no-service rails for further training. Li *et al* [34] employed the GradCAM method to locate crack positions in crack detection. These references indicated that the direct use of existing weakly supervised segmentation methods had already achieved promising performance.

Unfortunately, how to fundamentally enhance the resolution of CAM-like methods has always been a challenge. This work analyzes the root cause of this issue from the principle and finds that the fundamental reason affecting the segmentation performance of CAM methods is the resolution of features and the degree of feature separation, thus proposing a gradient truncation and fusion approach.

3. Background

3.1. CAMs

In this section, to illustrate explicitly, we first review 3 directly related methods, including Grad-CAM, LayerCAM, and FullGrad. Mathematically, consider a classifier f with L convolutional layers, whose parameters are $\boldsymbol{\theta}$. For a given input image $\boldsymbol{I}$ with category c, the prediction y^c before the softmax can be obtained by

$$y^c = f^c(\boldsymbol{I};\boldsymbol{\theta}). \tag{1}$$

Let $A^l \in \mathbb{R}^{W_l \times H_l \times C_l}$ denote the feature map of the kth channel generated by the lth layer in the, CNN, $l \in 1,2,\ldots,L$ where W_l and H_l are the width and height of lth feature map respectively, and C_l is the number of the channels in lth convolutional layer. The gradient of output score y^c with respect to the activation A^{kl} at location (i,j) is $g_{ij}^{ckl} = \frac{\partial y^c}{\partial A_{ij}^{kl}}$.

3.1.1. Grad-CAM.

When inputting an image I with category c, Grad-CAM acquires the channel-wise weight w_k^c by

$$w_{kl}^c = \frac{1}{Z_l}\sum_i\sum_j \frac{\partial y^c}{\partial A_{kl}^{ij}}, \tag{2}$$

where $Z_l = W_l \times H_l$. Then, the saliency map of specific layer l can be formed by a linear combination of the feature maps as

$$M^{cl}_{\text{Grad}-\text{CAM}} = \text{ReLU}\left(\sum_k w^c_{kl} \cdot A^{kl}\right), \tag{3}$$

where a ReLU function is applied to remove the negative responses from the saliency map.

3.1.2. LayerCAM. Unlike Grad-CAM, LayerCAM calculates the local importance instead of the average of the gradient matrix corresponding to a certain feature map. Formally, the element of the weight with spatial location (i,j) in the kth feature map and lth layer can be written as

$$w^{ckl}_{ij} = \text{ReLU}\left(\frac{\partial y^c}{\partial A^{kl}_{ij}}\right). \tag{4}$$

Finally, LayerCAM obtains the saliency map by

$$M^{cl}_{\text{Layer}-\text{CAM}} = \text{ReLU}\left(w^c_{kl} \otimes A^{kl}\right), \tag{5}$$

where $\otimes$ represents the Hadamard product.

3.1.3. FullGrad. LayerCAM is designed to better extract the semantic features at shallow layers, whereas FullGrad analyses the piecewise linear characteristics of ReLU CNNs and suggests using the gradient features of biases in each layer. It is worth noting that only gradient information and origin inputs are in consideration for FullGrad, which is formulated as

$$M^c_{\text{FullGrad}} = \frac{\partial y^c}{\partial \boldsymbol{x}} \otimes \boldsymbol{x} + \sum_{l=1}^{L}\sum_{k=1}^{C_l} \Psi\left(\frac{\partial y^c}{\partial b^{kl}} \otimes \boldsymbol{b}^{kl}\right), \tag{6}$$

where $\boldsymbol{x}$ is an input image, $\boldsymbol{b}^{kl}$ is the bias term of kth neuron in lth layer, and $\Psi(\cdot)$ is a post-processing function. Generally, $\Psi(\cdot)$ is used to align saliency maps, which is formulated as follows:

$$\Psi(\boldsymbol{x}) = \text{bilinearUpscale}(\text{normalize}(|\boldsymbol{x}|)), \tag{7}$$

where bilinearUpscale is a typical image interpolation operation that resizes its input image to target size, and $\text{normalize}(x_{ij}) = \frac{x_{ij} - \min \boldsymbol{x}}{\max \boldsymbol{x} - \min \boldsymbol{x}}$.

4. Methodology

4.1. Semantic information from shallow layers

In this section, we introduce the situation of semantic information in different layers of convolutional neural networks, and then introduce two ways to extract information, namely layerization and truncation tricks.

4.1.1. Layerization trick. Most CAM series methods obtain discriminative features from the output of the last convolutional layer. Inspired by GradFull and LayerCAM, we visualize feature maps of the different layers from a VGG16 [35] network trained on the KSDD dataset [36]. As shown in figure 1, we generate heatmaps for the output of the convolutional layers at different stages using the Grad-CAM method, which are referred to as S1–S5 in sequence. In contrast to the view in [11], we can observe that the high-resolution semantic features exist still in the shallow feature maps, while containing some noise. In detail, the outputs at different stages of convolutional neural networks contain varying degrees of semantic features. The shallower layers are richer in semantics and have higher resolution, which is related to the pooling structure. However, the highlighted areas in the shallow layers have more noise. While the deeper layers are the opposite, they have significantly less noise but the resolution of features is reduced, and detailed semantic information is lost.

Naturally, we introduce the technique of fusing shallow feature maps to improve the resolution of CAM-based methods as follows:

$$M^c_{\text{CAM}} = \sum_l \psi\left(M^{cl}_{\text{CAM}}\right) \tag{8}$$

where $\psi(\cdot)$ is the bilinear interpolation operation. It is worth to note that we find that the discriminability of the fused features may change, which can be solved by filtering, but for the sake of fairness we use equation (8) directly to generate the heatmap in the experiments.

4.1.2. Truncation trick. Fusing the high-level features, despite improving the resolution of the features, simultaneously introduces considerable noise. Therefore, we present the gradient truncation method to obtain more high-quality heatmaps. Specifically, we assume that the positive gradient reflects the features that play a positive role in classification. Moreover, the larger the gradient, the more important the features are, and the less noise there is. Formally, for all CAM series methods that rely on gradients to measure their weights, we can truncate the gradient as

$$M^{cl}_{\text{CAM}} = M^{cl}_{\text{CAM}}\left(\boldsymbol{x}, T\left(\frac{\partial y^c}{\partial A_l} \geqslant \delta\right) \otimes \frac{\partial y^c}{\partial A_l}\right), \tag{9}$$

where $T(\cdot)$ is the indicator function, A_l is the target feature map in the lth layer, and δ is a hyperparameter that is set to the δth percentile of the positive values in each feature map. Eventually, we obtain the heatmap by

$$S^c_{\text{CAM}} = \sum_l \psi\left(M^{CL}_{\text{CAM}}\left(\boldsymbol{x}, T\left(\frac{\partial y^{cl}}{\partial Z} \geqslant \delta\right)\right) \otimes \frac{\partial y^{cl}}{\partial Z}\right). \tag{10}$$

4.2. LTGrad-CAM

By combining the above layerization and truncation tricks, our method can be used as a plug-in for any gradient-weighted

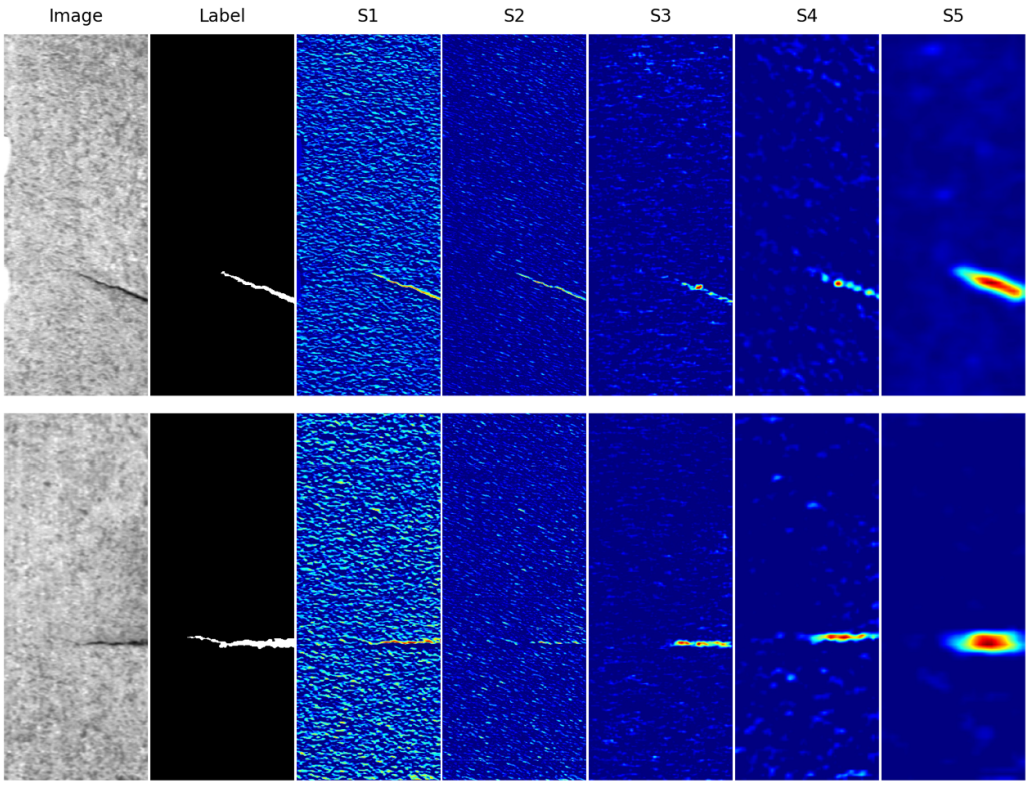


Figure 1. Heatmaps generated by Grad-CAM for 5 different stages in the VGG16 model. Images are randomly selected from the KSDD dataset.

CAM-based method. In particular, combining equations (2), (3) and (10), we can directly upgrade Grad-CAM as

$$S^c_{\mathrm{LTGrad-CAM}} = \sum_l \psi\left(\sum_k \left(\left(\frac{1}{Z_l}\sum_i\sum_j \frac{\partial y^c}{\partial A^{kl}_{ij}} \otimes T\left(\frac{\partial y^c}{\partial A^{kl}_{ij}} \geqslant \delta\right)\right)\right)\cdot A^{kl}\right). \quad (11)$$

As shown in figure 2, by integrating the shallow high-resolution semantic information, and the denoising effect of the gradient truncation, our LTGrad-CAM method is able to display high-resolution defective regions. Similarly, LayerCAM and FullGrad can be updated to LTLayerCAM and LTFullGrad, respectively.

We would like to emphasize that although LayerCAM has proposed the idea of fusing different layers, LayerCAM believes that the layerization trick is not suitable for other CAM methods. Our study corrects this point, and reveals that layerization tricks are potentially useful in different scenarios.

5. Experiment

5.1. Experimental setup

5.1.1. Datasets. We conduct WSSS experiments of the proposed method on two datasets. The first is KSDD [36] dataset, and the dataset contains 50 defective electron commutators. Eight different images were taken for each commutator, captured without any overlap of the image surfaces, for a total of 399 images. Of these, 52 images showed defects, while the remaining 347 did not have any such problems. For the purpose of this study, these images were strategically divided into two different subsets: a training set and a test set. The training set consists of 42 defective images and 277 non-defective images. On the other hand, the test set consists of 10 defective images and 70 defect-free images. The second dataset is the industrial surface defect detection dataset KSDD2 [22]. This dataset consists of over 3000 images with approximately 230 pixels in width and 630 pixels in height. The dataset is split into a training set with 2085 defect-free samples and 246 defective samples, and a test set with 894 defect-free samples and 110 defective samples.

5.1.2. Implementation details. We train the classification models on each dataset with image-level labels only, which adopts the VGG16 model pre-trained on ImageNet [37]. We employ the Otsu method to process the CAMs to produce the final segmentation results. All experiments are executed on a PC with an NVIDIA RTX 2080Ti. We prefix the method with 'LT' to indicate that the method uses the layerization trick and the truncation trick, l is the number of layers fused, and δ is the percentile of the truncated positive gradient. To be fair, we

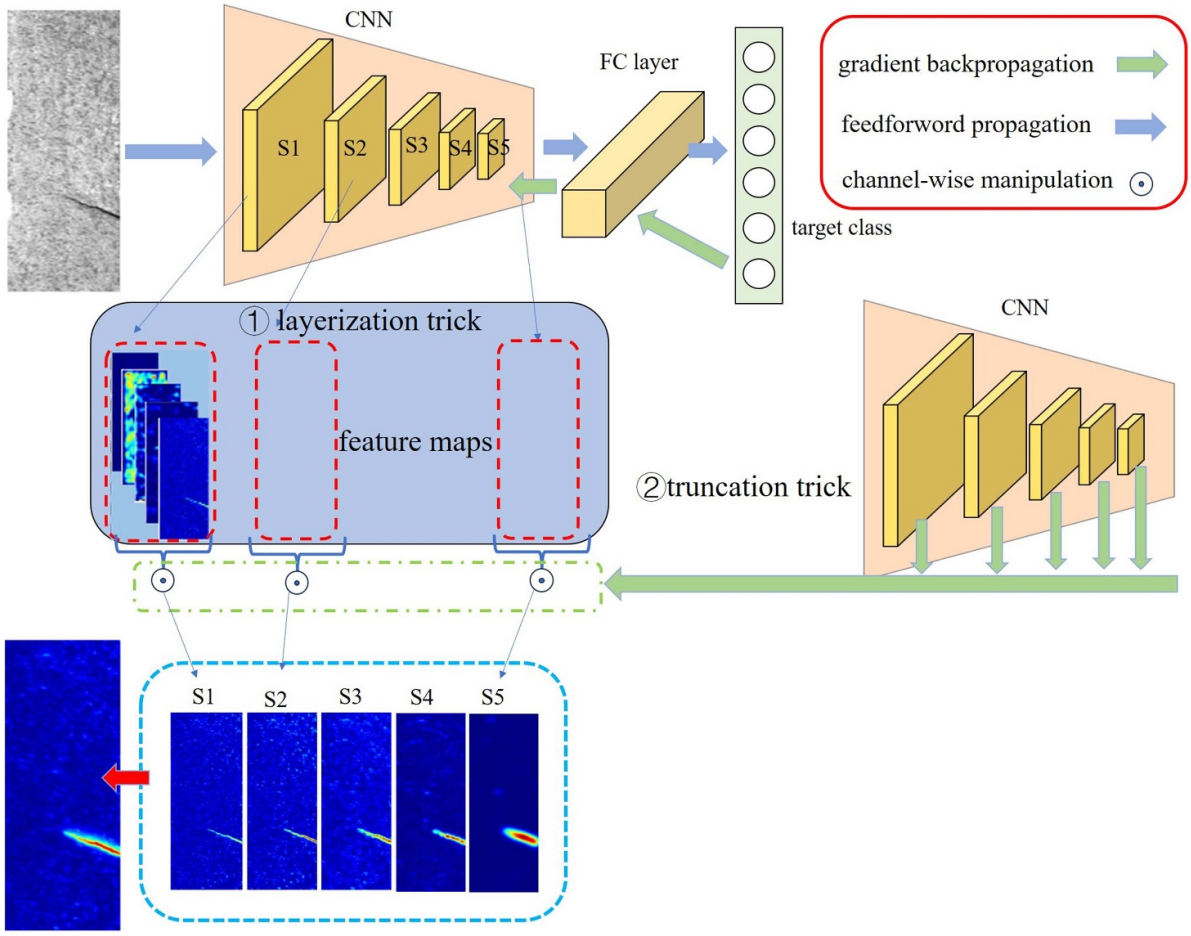


Figure 2. Flowchart of the proposed LTGrad-CAM by combining layerization and truncation tricks.

only get the output of all convolution layers in the FullGrad-related method, while the other methods only get the output of the stages by default.

5.1.3. Evaluation metrics. WSSS requires higher pixel accuracy and hence a higher requirement for the fine granularity of the CAMs. We adopt the intersection-over-union (IoU) to evaluate the segmentation results and use pixel recall, accuracy, and Mirco-F1 metrics for a comprehensive evaluation.

5.2. Comparison with state-of-the-art methods

We report the evaluation results of WSSS on both the defect detection datasets. Table 1 demonstrates the segmentation performance on the KSDD dataset. Our method LTGrad CAM improves 7.84% over LayerCAM in terms of IoU score, and also beats the best performing Ablation-CAM by 6.13%. It is important to note that we set the hyperparameters δ to 50, and the selection principle can be based on the results of the training or validation sets, which could refer to the analysis in section 6.1. As shown in figure 3, we show the qualitative results of LTGrad-CAM with other methods. It can be noticed that LTGrad-CAM generates high-resolution defect segmentation results and also introduces some background noise. While the gradient truncation technique mitigates the background noise but does not eliminate it completely, it can be further removed by simple image processing techniques as long as the features are properly highlighted. Moreover, comparing methods FullGrad and LTFullGrad, it can be found that the background noise interference is indeed weakened, which also proves the effectiveness of the proposed gradient truncation trick.

As shown in table 2, the superiority of LTGrad-CAM is also evident in KSDD2 dataset. It is worth noting that $\delta = 0$

Table 1. The weakly supervised semantic segmentation performance on the KSDD dataset. The results in bold indicate the best performance.

Method	IoU (%)	Precision (%)	Recall (%)	Micro-F1 (%)
Grad-CAM	17.50	19.27	65.56	29.79
Grad-CAM++	17.22	18.97	65.14	29.38
XGrad-CAM	17.06	18.86	64.10	29.15
Ablation-CAM	17.96	18.97	64.08	30.46
Score-CAM	17.63	19.50	64.76	29.98
LayerCAM	16.25	17.76	65.61	27.95
FullGrad	13.19	13.58	**82.21**	23.31
LTGrad-CAM	**24.09**	**27.21**	67.75	**38.83**

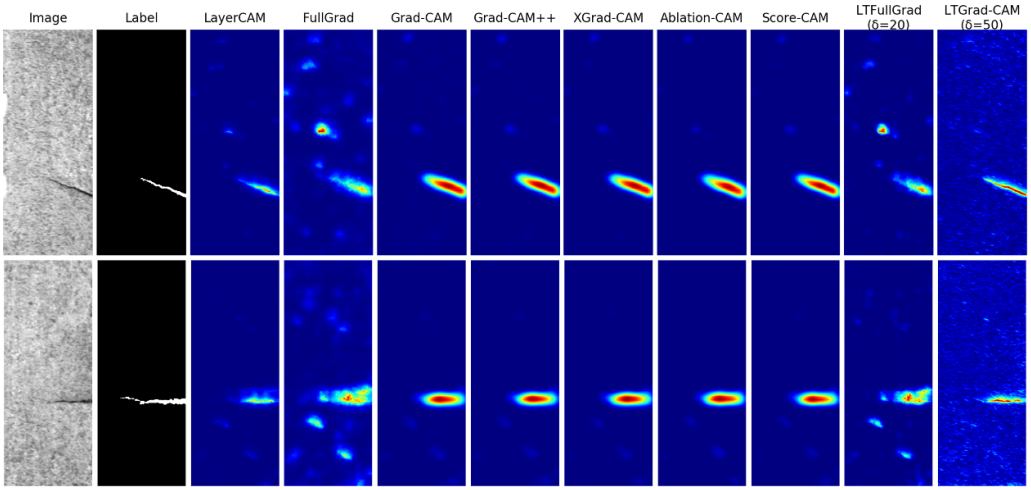


Figure 3. Qualitative visualization of class activation maps generated by different methods in KSDD dataset.

Table 2. The segmentation performance on the KSDD2 dataset. The results in bold indicate the best performance.

Method	IoU (%)	Precision (%)	Recall (%)	Micro-F1 (%)
Grad-CAM	38.67	49.19	64.41	55.78
Grad-CAM++	35.17	44.27	63.10	52.04
XGrad-CAM	39.67	51.03	64.06	56.81
Ablation-CAM	26.06	31.02	61.99	41.35
Score-CAM	40.80	56.72	59.25	57.96
LayerCAM	42.01	60.46	57.92	59.17
FullGrad	37.95	46.29	**67.81**	55.02
LTGrad-CAM	**42.67**	**66.98**	54.03	**59.81**

we have chosen at this experiment is screened based on the data in the training set. If the default value of 0 is used as the positive truncation, the performance of LTGrad-CAM can reach 42.67%. Nevertheless, an outperforms the IoU score of the state-of-the-art LayerCAM by 0.66%, and for the original Grad-CAM there is an even 4% increase. As shown in figure 4, we show in detail the heatmap produced by different methods on the KSDD2 dataset. Compared to LayerCAM, the defective regions of our method LTGrad-CAM are more prominent and complete, although some background noise is also introduced. Comparing FullGrad and LTFullGrad at the same time, the effective suppression of background noise by our plug-in strategy is demonstrated.

5.3. Is gradient truncation beneficial for extracting shallow features?

As shown in figures 5 and 6, we selected images on the KSDD dataset and the KSDD2 dataset, respectively, to compare the layer-by-layer heatmaps generated by GradCAM and LTGrad-CAM. The truncation values used remain at the settings described earlier. It can be found that the gradient truncation method can achieve effective extraction of shallow features, mainly by expanding the contrast between the background and the target. This is consistent with our intuition that the higher the gradient value, the more likely the corresponding region is to be the target region.

5.4. Layerization and truncation trick as plug-in methods

We apply the proposed plug-in approach to the FullGrad method as shown in table 3. It can be found that after adding the plug-in method, the IoU score of FullGrad is improved by 3.29% and 1.43% on two different datasets, KSDD and KSDD2 respectively. In addition, the hyperparameter δ are still chosen using the estimation results on the training set. As shown in figure 7, compared to the original method, we can find that by adding the proposed plug-in method we can enhance the defective region and reduce the background noise, which greatly improves the performance of the original method.

6. Discussions

6.1. Influence of the truncation hyperparameters δ

The hyperparameter δ is crucial for the proposed method. We tested different truncation percentages (10 sample intervals from 0 to 90) on the KSDD dataset and plotted the variation of IoU scores on the training and test sets. We chose three typical methods, i.e., Grad-CAM, LayerCAM, and FullGrad, and plugged them with the proposed plug-in method, denoted as LTGrad-CAM, LTLayerCAM, and LTFullGrad, respectively.

As shown in figure 8, all three curves show an increasing and then decreasing trend. An obvious explanation is that as the number of truncated quantum dots increases, noise that does not contribute much to the predicted value is filtered out first. Subsequently, the target features also start to decrease. As shown in figure 9, the trend on the training set is very similar to

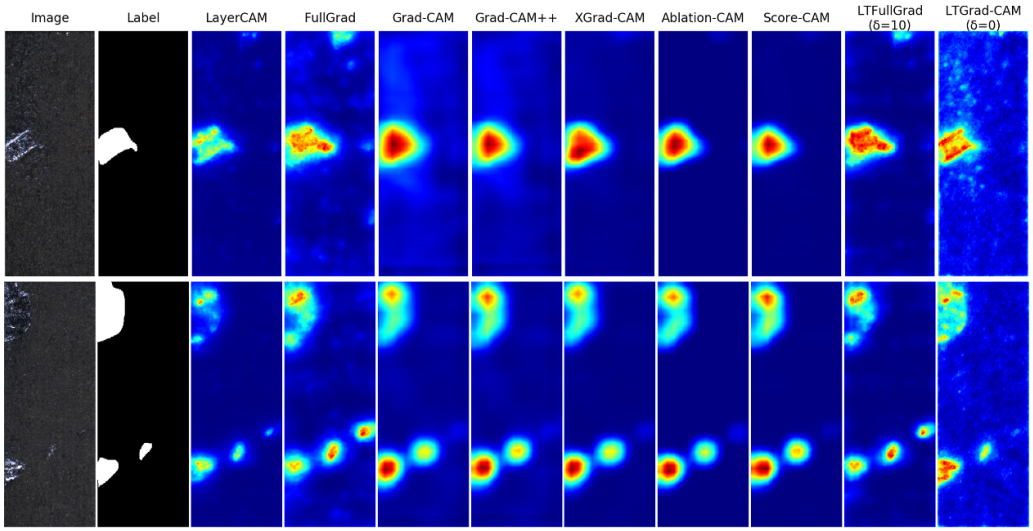


Figure 4. Qualitative visualization of class activation maps generated by different methods in KSDD2 dataset.

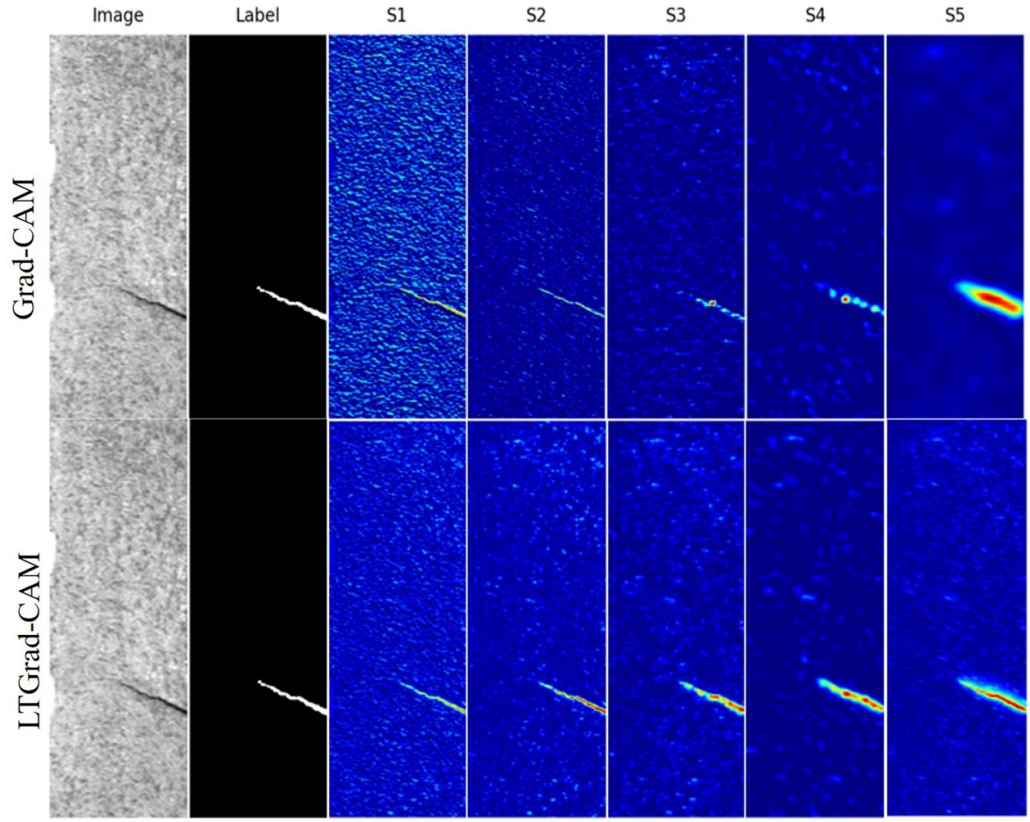


Figure 5. Layer-by-layer comparison of the heatmaps generated by Grad-CAM and LTGrad-CAM on KSDD dataset.

that on the test set, i.e. suitable gradient truncation hyperparameters can be found on the training set. This suggests that our proposed method can be used as a plug-and-play module to help other related CAM methods to benefit from this concise operation. Finally, the fact that our proposed LTGrad-CAM performs substantially better than the other two methods in the defect detection task also suggests that earlier proposed CAM methods may have greater potential.

6.2. The limitation of WSSS in defect detection

At present, the direct segmentation performance of WSSS still needs to be improved, yet this task still holds significant practical importance in engineering practice. Firstly, in situations where data is scarce and annotation is difficult, it provides meaningful segmentation results for reference. Secondly, the training results from weakly supervised

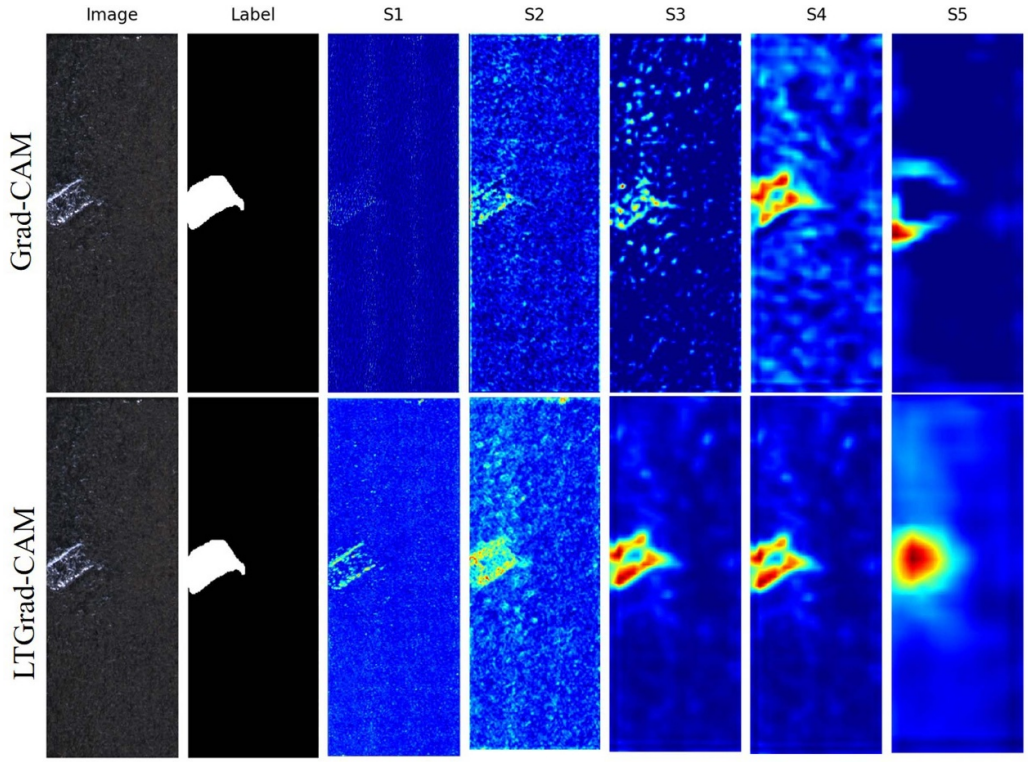


Figure 6. Layer-by-layer comparison of the heatmaps generated by Grad-CAM and LTGrad-CAM on KSDD2 dataset.

Table 3. Results of the plug-in method LTFullGrad versus the original FullGrad on the KSDD and KSDD2 dataset.

	Method	mIoU (%)	Precision (%)	Recall (%)	F1-score (%)
KSDD	FullGrad	13.19	13.58	82.21	23.31
	LTFullGrad	16.84	19.50	51.54	28.30
KSDD2	FullGrad	37.95	46.29	67.81	55.02
	LTFullGrad	39.38	53.79	59.53	56.51

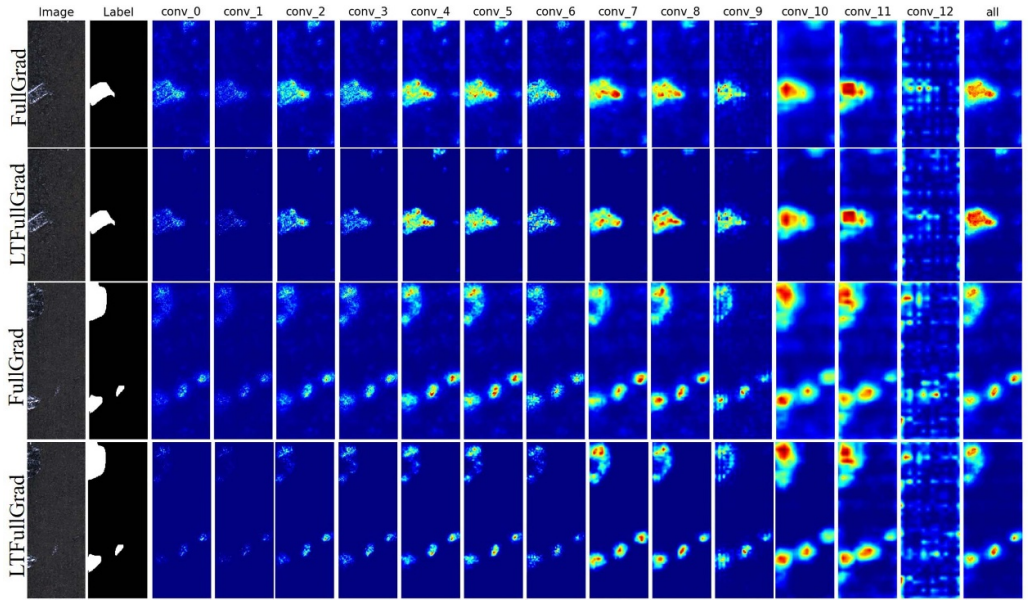


Figure 7. Comparison of the heatmaps generated by FullGrad and LTFullGrad layer-by-layer.

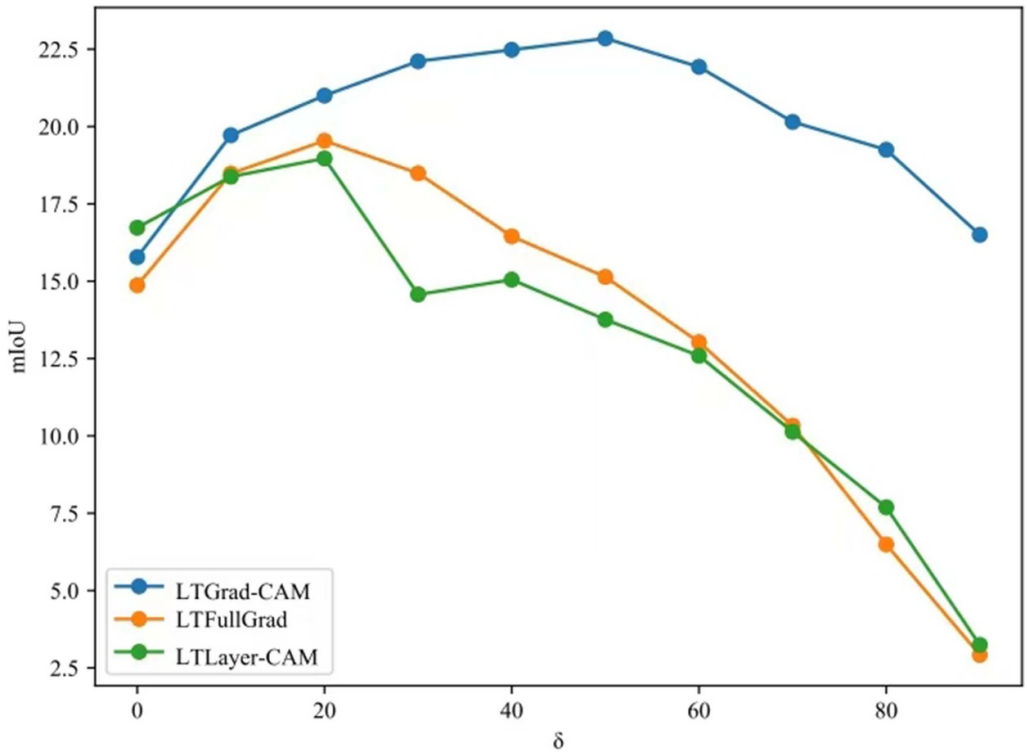


Figure 8. Variation of semantic segmentation performance of three methods with different truncation values δ on the training set. The blue line indicates LTGrad-CAM, the orange line indicates LTFullGrad, and the green line indicates LTLayerCAM.

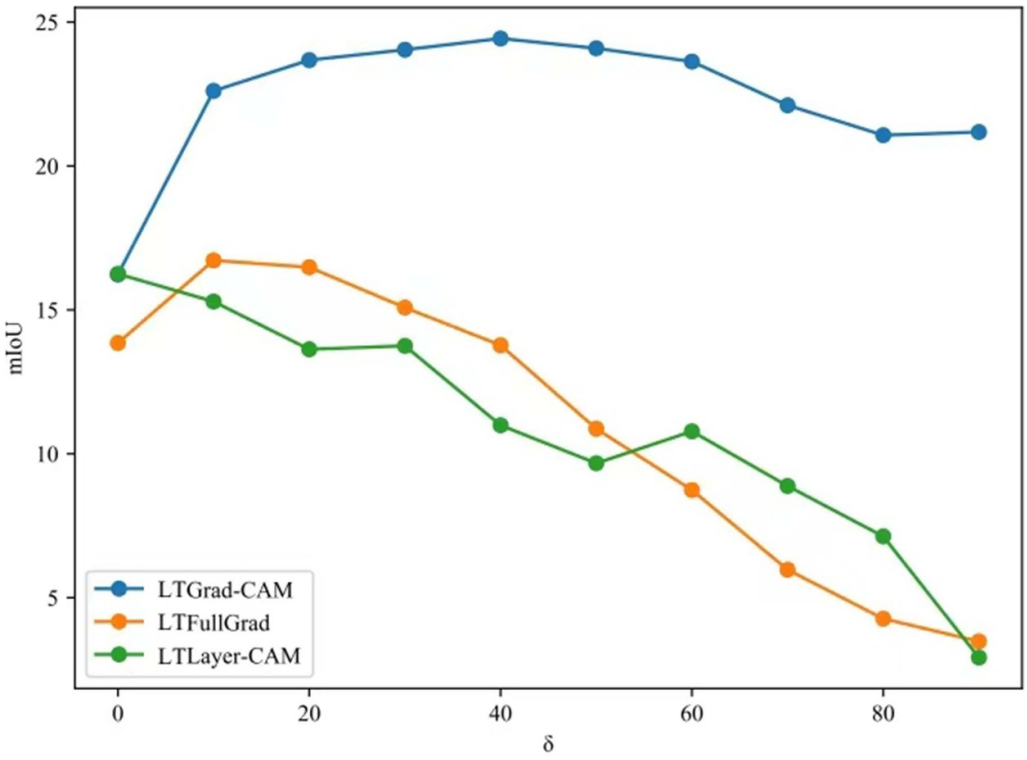


Figure 9. Variation of semantic segmentation performance of three methods with different truncation values δ on the testing set. The blue line indicates LTGrad-CAM, the orange line indicates LTFullGrad, and the green line indicates LTLayerCAM.

methods can serve as pseudo-labels, offering additional training information. Thirdly, since semantic segmentation is a high-level task, if evaluated using object detection or object localization as metrics, WSSS methods can deliver performance comparable to fully supervised object detection. This is of great importance for rapidly and effectively establishing intelligent systems in engineering practice.

7. Conclusion

In this work, we revisited the key factors affecting the quality of CAMs. We found that the outputs at different stages of convolutional neural networks contain semantic features to varying degrees. The shallower layers are rich in semantic features but have more noise, while the deeper layers have reduced semantic feature resolution but also less noise. The magnitude of gradients can measure whether features belong to the target or are noise. Therefore, we propose a gradient truncation technique to filter out noise in the shallow layer feature maps, making it possible to fuse semantic information from different layers. Experiments have proven that this method can be well combined with various gradient-weighted CAM-related methods. In the future, we will explore how to further improve the resolution of CAM methods to enhance the performance of WSSS.

Data availability statement

No new data were created or analyzed in this study.

ORCID iDs

Hang-Cheng Dong https://orcid.org/0000-0002-4880-6762
Guodong Liu https://orcid.org/0000-0002-3300-7845

References

[1] Gao Y, Li X, Wang X V, Wang L and Gao L 2022 A review on recent advances in vision-based defect recognition towards industrial intelligence *J. Manuf. Syst.* **62** 753–66
[2] Ren Z, Fang F, Yan N and You W 2022 State of the art in defect detection based on machine vision *Int. J. Precis. Eng. Manuf. Green Technol.* **9** 661–91
[3] Chen Z, Zhang J, Lai Z, Zhu G, Liu Z, Chen J and Jianqiang L 2023 The devil is in the crack orientation: a new perspective for crack detection *Proc. IEEE/CVF Int. Conf. on Computer Vision* pp 6653–63
[4] Zheng X, Zheng S, Kong Y and Chen J 2021 Recent advances in surface defect inspection of industrial products using deep learning techniques *Int. J. Adv. Manuf. Technol.* **113** 35–58
[5] Sultana F, Sufian A and Dutta P 2020 Evolution of image segmentation using deep convolutional neural network: a survey *Knowl. Based Syst.* **201** 106062
[6] Ahmad H M and Rahimi A 2022 Deep learning methods for object detection in smart manufacturing: a survey *J. Manuf. Syst.* **64** 181–96
[7] Tao X, Zhang D, Wang Z, Liu X, Zhang H and Xu D 2018 Detection of power line insulator defects using aerial images analyzed with convolutional neural networks *IEEE Trans. Syst. Man Cybern. Syst.* **50** 1486–98
[8] Cui L, Jiang X, Mingliang X, Wanqing L, Pei L and Zhou B 2021 SDDNet: a fast and accurate network for surface defect detection *IEEE Trans. Instrum. Meas.* **70** 1–13
[9] Zhou B, Khosla A, Lapedriza A, Oliva A and Torralba A Learning deep features for discriminative localization *2016 IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2016* (*Las Vegas, NV, USA*, *27–30 June 2016*) pp 2921–9
[10] Selvaraju R R, Cogswell M, Das A, Vedantam R, Parikh D and Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization *IEEE Int. Conf. on Computer Vision, ICCV 2017* (*Venice, Italy*, *22–29 October 2017*) pp 618–26
[11] Jiang P-T, Zhang C-B, Hou Q, Cheng M-M and Wei Y 2021 LayerCAM: exploring hierarchical class activation maps for localization *IEEE Trans. Image Process* **30** 5875–88
[12] Zhang D, Han J, Cheng G and Yang M-H 2022 Weakly supervised object localization and detection: a survey *IEEE Trans. Pattern Anal. Mach. Intell.* **44** 5866–85
[13] Sun K, Shi H, Zhang Z and Huang Y 2021 ECS-Net: improving weakly supervised semantic segmentation by using connections between class activation maps *Proc. IEEE/CVF Int. Conf. on Computer Vision* pp 7283–92
[14] Wang Y, Zhang J, Kan M, Shan S and Chen X 2020 Self-supervised equivariant attention mechanism for weakly supervised semantic segmentation *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*
[15] Srinivas S and Fleuret F 2019 Full-gradient representation for neural network visualization *Advances in Neural Information Processing Systems 32: Annual Conf. on Neural Information Processing Systems 2019, NeurIPS 2019* (*Vancouver, BC, Canada*, *8–14 December 2019*) ed H M Wallach, H Larochelle, A Beygelzimer, F D'alch´e-buc, E B Fox and R Garnett pp 4126–35
[16] Chen S, Huang Z, Wang T, Hou X and Jun M 2023 Wafer map defect recognition based on multi-scale feature fusion and attention spatial pyramid pooling *J. Intell. Manuf.* **36** 271–84
[17] Luo J, Yang Z, Shipeng L and Yilin W 2021 FPCB surface defect detection: a decoupled two-stage object detection framework *IEEE Trans. Instrum. Meas.* **70** 1–11
[18] Zheng Z, Zhang S, Shen J, Shao Y and Zhang Y 2021 A two-stage CNN for automated tire defect inspection in radiographic image *Meas. Sci. Technol.* **32** 115403
[19] Dong H, Song K, He Y, Jing X, Yan Y and Meng Q 2019 PGA-Net: pyramid feature fusion and global context attention network for automated surface defect detection *IEEE Trans. Ind. Inform.* **16** 7448–58
[20] Su S, Du S, Wei X and Lu X 2022 RFS-Net: railway track fastener segmentation network with shape guidance *IEEE Trans. Circuits and Syst. Video Technol.* **33** 1398–412
[21] Li M, Chen N, Hu Z, Li R, Yin S and Liu J 2024 A global feature interaction network (GFINet) for image segmentation of GaN chips *Adv. Eng. Inf.* **62** 102670
[22] Bozic J, Tabernik D and Skocaj D 2021 Mixed supervision for surface-defect detection: from weakly to fully supervised learning *Comput. Ind.* **129** 103459
[23] Wan F, Liu C, Wei K, Xiangyang J, Jiao J and Qixiang Y 2019 C-MIL: continuation multiple instance learning for weakly supervised object detection *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 2199–208

[24] Zhang J, Su H, Zou W, Gong X, Zhang Z and Shen F 2021 CADN: a weakly supervised learning-based category-aware object detection network for surface defect detection *Pattern Recognit.* **109** 107571
[25] Wang H, Wang Z, Du M, Yang F, Zhang Z, Ding S, Mardziel P and Xia H Score-CAM: score-weighted visual explanations for convolutional neural networks *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition, CVPR Workshops 2020* (*Seattle, WA, USA*, *14–19 June 2020*) pp 111–9
[26] Chattopadhay A, Sarkar A, Howlader P and Balasubramanian V N 2018 Grad-CAM++: generalized gradient-based visual explanations for deep convolutional networks *2018 IEEE Winter Conf. on Applications of Computer Vision (WACV)* (IEEE) pp 839–47
[27] Fu R, Hu Q, Dong X, Guo Y, Gao Y and Biao L Axiom-based Grad-CAM: towards accurate visualization and explanation of CNNs *31st British Machine Vision Conf. 2020, BMVC 2020, Virtual Event* (*UK*, *7–10 September 2020*) (https://doi.org/10.1109/IEMBS.2008.4650227)
[28] Jung H and Youngrock O 2021 Towards better explanations of class activation mapping *Proc. IEEE/CVF Int. Conf. on Computer Vision* pp 1336–44
[29] Bach S, Binder A, Montavon G, Klauschen F, M¨uller K-R and Samek W 2015 On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation *PLoS One* **10** e0130140
[30] Lee J R, Kim S, Park I, Eo T and Hwang D 2021 Relevance-CAM: your model already knows where to look *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 14944–53
[31] Ramaswamy H G *et al* 2020 Ablation-CAM: visual explanations for deep convolutional network via gradient-free localization *Proc. IEEE/CVF Winter Conf. on Applications of Computer Vision* pp 983–91
[32] Li M, Chen N, Suo X, Yin S and Liu J 2023 An efficient defect detection method for nuclear-fuel rod grooves through weakly supervised learning *Measurement* **222** 113708
[33] Zhang D, Song K, Jing X, Dong H and Yan Y 2022 An image-level weakly supervised segmentation method for no-service rail surface defect with size prior *Mech. Syst. Signal Process.* **165** 108334
[34] Li Y, Bao T, Xu B, Shu X, Zhou Y, Du Y, Wang R and Zhang K 2022 A deep residual neural network framework with transfer learning for concrete dams patch-level crack classification and weakly-supervised localization *Measurement* **188** 110641
[35] Simonyan K and Zisserman A 2015 Very deep convolutional networks for large-scale image recognition *3rd Int. Conf. on Learning Representations, ICLR 2015* (*San Diego, CA, USA*, *7–9 May 2015*) ed Y Bengio and Y LeCun (Conf. Track Proc.)
[36] Tabernik D, ˇSela S, Skvarˇc J and Skoˇcaj D 2020 Segmentation-based deep-learning approach for surface-defect detection *J. Intell. Manuf.* **31** 759–76
[37] Russakovsky O *et al* 2015 ImageNet large scale visual recognition challenge *Int. J. Comput. Vision* **115** 211–52