Smart Structures and Systems, *Vol. 35*, *No. 3* (*2025*) *153-162*
https://doi.org/10.12989/sss.2025.35.3.153

Binary information enhancement network for efficient steel defects detection and classification

LiJun Wu, Qingqi Chen, Jingxuan Su, Zhicong Chen* and Shuying Cheng

College of Physics and Information Engineering, *Fuzhou University*, *Fuzhou 350116*, *China*

(*Received January 13*, *2025*, *Revised March 19*, *2025*, *Accepted April 3*, *2025*)

Abstract. Defect detection plays a crucial role in ensuring the safety and longevity of structures, with defect region classification particularly beneficial for focusing efforts on potential defect areas. Traditional deep convolutional neural networks (DCNNs) based defect classification networks still have a high number of parameters and computational demands, making them unsuitable for embedded systems. This paper proposes the Adaptive Prior Activation-Based Binary Information Enhancement Network (AOIE-Net), which significantly reduces computational requirements by binarizing weights and activations. Specifically designed for steel defect detection, AOIE-Net optimizes the binary quantization process and enhances feature representation to improve the performance of BNNs in steel defect detection tasks. AOIE-Net introduces a Dual Batch Normalization-based Information Enhancement Block (DBN-IEB) and an Adaptive Binary Activation Independent Optimization (ABA-IO) method to reduce computational complexity while boosting classification accuracy. Experimental results demonstrate that AOIE-Net outperforms state-of-the-art binary neural network models on CIFAR-10, ImageNet, and the NEU-CLS steel defect dataset, achieving classification accuracy of 90.6%, 72.1%, and 99.4%, respectively. The proposed method offers an efficient, low-complexity solution for real-time defect classification in large-scale structural inspections and holds significant potential for practical applications.

Keywords: binary neural network; deep learning; enhanced binary information; image classification; steels defect

1. Introduction

Defect detection is critical for the safety and longevity of structures, as undetected defects can lead to structural failure and accelerated deterioration. Early detection helps minimize maintenance costs, prevent economic losses, and ensure continued functionality. Consequently, it has been a key area of research in both industry and academia (Wang *et al*. 2024, Duan *et al*. 2024). Cracks are the most common and easy to identify defects. In recent years, artificial intelligence assisted crack detection (Ko *et al*. 2021), segmentation, and measurement methods have made remarkable progress (Schanz *et al*. 2023, He *et al*. 2024). With the advent of technologies such as drones equipped with cameras (Qiu and Lau 2023, Tse *et al*. 2024, Zhong *et al*. 2024), surface crack scanning has become more flexible and efficient, facilitating the widespread application of crack detection and segmentation tasks. Compared with crack classification, crack detection (segmentation) involves more complex model structures and longer inference times, as they not only classify the image content but also localize cracks and assign a category to each pixel (Badea *et al*. 2016). Given that structural surface cracks are typically sparse, indiscriminately applying complex detection (segmentation) algorithms to large-scale structure surfaces with extensive crack-free areas not only limits real-time performance but also constitutes a waste of time. To address this issue, cascaded approach is proposed for efficient and real-time crack detection when being implemented on large-scale images (Chen *et al*. 2024, Tang *et al*. 2022). Basically, using classification algorithms to classify cracks in the surface images of the entire structure, and employing a cascade method to eliminate backgrounds without cracks, thereby filtering out the areas containing cracks. Since the classification task is simpler and faster than the segmentation task, a lightweight and accurate classification network can quickly exclude most non-cracked background regions. Then, crack segmentation algorithms are applied only to the sparse potential crack regions, which can significantly improve crack segmentation efficiency, as shown in Fig. 1 work is carried out to quickly exclude a large number of non-crack background areas. For the sparse crack areas obtained through classification, complex crack detection and segmentation algorithms are then applied, so as to achieve efficient building crack detection work, as shown in Fig. 1 Therefore, finding a high-speed and real-time image classification algorithm is of great importance for improving the efficiency of building crack scanning.

In this cascaded crack segmentation approach, an efficient and real-time image classification algorithm is essential. Deep Convolutional Neural Networks (DCNNs) have demonstrated powerful performance in image-related tasks, as well as in crack classification, detection, and segmentation (He *et al*. 2016, Wang *et al*. 2022a, Ren *et al*. 2016, Guo *et al*. 2022, Long *et al*. 2015, Xie *et al*. 2021, Huang *et al*. 2018, König *et al*. 2022). However, the large

*Corresponding author, Ph.D., Professor,
E-mail: zhicong.chen@fzu.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

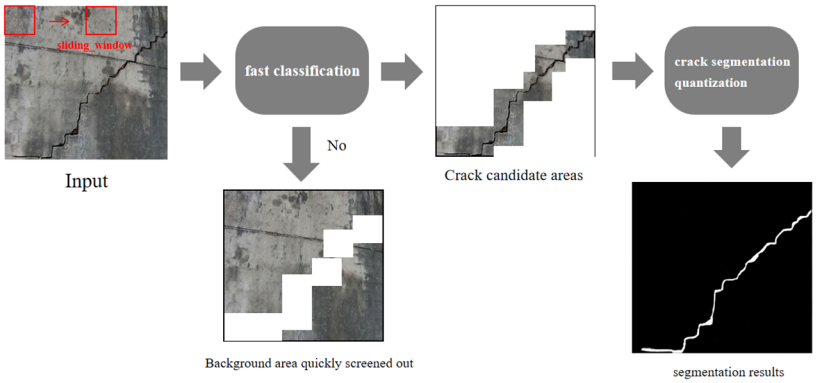


Fig. 1 Fast crack classification and detection

number of training parameters and high computational demands of DCNNs limit their application on resource-constrained platforms. Several strategies, such as optimizing model structures (Zhu and Newsam 2017), reducing parameter redundancy (Li *et al*. 2024), and enhancing computational efficiency (Ruan *et al*. 2023, Choukroun *et al*. 2019), have been proposed to address these issues. Among them, Binarized neural networks (BNN) (Hubara *et al*. 2016) extremely compress the full-precision activation values and weights of the neural networks and binarize them into single-bit discrete values {–1, +1}, greatly reducing the storage requirement. Moreover, by transforming floating-point operations into simple operations like bitwise XOR and bit counting, BNNs enable efficient deployment of complex models in resource-constrained environments and has received extensive attention from researchers.

BNNs compress the redundant parameters of full-precision networks but inevitably reduce parameter representation and introduce information loss during model training and inference. Research has focused on reducing quantization errors and optimizing network structures to mitigate accuracy loss. As known, binarization converts 32-bit weights and activations to single-bit values, and quantization errors are influenced by factors like scaling and quantization functions. Early work optimized scaling factors (Rastegari *et al*. 2016, Bulat and Tzimiropoulos 2019) to retain more feature information. Recent advancements in quantization include dynamic threshold adjustments and new binary functions to reduce information loss (Wang *et al*. 2020, Lin *et al*. 2022). However, most methods limit binary values to fixed set{–1, 0, 1}, making it hard to represent the full range of full-precision coefficients. AdaBin (Tu *et al*. 2022) addresses this by introducing adaptive binary quantization, expanding the value range and reducing loss, but still assumes a bell-shaped distribution of the original full-precision weights and activation values, which may not match actual data.

By optimizing network structures, the expressive ability of BNNs has been greatly improved. Previous works have enhanced convolutional layers, grouping strategies, and residual connections within classic networks (Liu *et al*. 2019, Zhuang *et al*. 2019, Shen *et al*. 2020, Martinez *et al*. 2020), leading to improvements in expressive ability and accuracy of BNNs. Recent developments have introduced dedicated BNN architectures such as MeliusNet (Bethge *et al*. 2020) and ReActNet (Liu *et al*. 2020), which use MobileNet as the baseline. These architectures incorporate shortcut connections, reducing computational costs while maintaining accuracy comparable to full-precision lightweight networks. However, there are still areas for improvement (Tu *et al*. 2022, Zhang *et al*. 2021). For example, FracBNN introduces a Batch Normalization (BN) layer after each shortcut in ReActNet to balance the number of positive and negative values after activation, improving Top1 accuracy on ImageNet to 71.8%. Instead of simple RSign function, IE-Net (Ding *et al*. 2022) adds an information enhancement module that uses RSign with learnable thresholds for each channel so as to effectively reduce the information loss. However, such information enhancement modules involve multiple binary convolutions, which increases computational cost and IE-Net still faces limitations in overall performance due to its reliance on traditional neural network structures.

This paper proposes the Adaptive Prior Activation-Based Binary Information Enhancement Network (AOIE-Net) to address challenges in BNNs so as to provide quickly crack regions classification. Building on the FracBNN architecture, AOIE-Net introduces an adaptive binary quantization function to replace the RSign function, optimizing binary values to better match real-value distributions. In particular, an information integration module with dual BN inputs is proposed, which adjusts the various data distribution to approximately normal distribution before binarization in each module and uses a learnable threshold for the BN bias term, so as to efficiently enhancing feature representation after binary activation. Moreover, to address the increased parameter count from the dual-input module in IE-Net, this paper introduces a lightweight Adaptive Binary Activation Independent Optimization Method (ABA-IO). ABA-IO separates the adaptive binary activation from binary convolution operations, so as to avoid performing dual convolutions in Adaptive Prior Activation-Based Binary Information Enhancement module. This approach can significantly enhance the capacity and performance with little increase in the number of model parameters. The overall structure of the AOIE-Net is given in Fig. 2 Experimental results show that AOIE-Net achieves 90.6% accuracy on CIFAR-10 (ResNet-20) that is 2.4% higher than that of FracBNN,

72.1% Top-1 accuracy on ImageNet that is outperforming other state-of-the-art BNNs, and 99.4% accuracy on the NEU-CLS steel defect dataset that is 1.5% higher than IE-Net with half computational complexity (FLOPs), outperforming state-of-the-art BNNs with reduced computational complexity. These results demonstrate the potential of AOIE-Net for efficiently identifying crack area and excluding background in structure surface inspection.

2. Related works

Convolutional neural networks (CNNs) have achieved significant success in detecting and segmenting cracks (Maruthi and Lee 2024, Jin *et al*. 2022). However, traditional CNN models often have large numbers of parameters, increasing complexity and computational load, which limits their real-time application in resource-constrained environments like embedded systems (Chen *et al*. 2024, Huang *et al*. 2024, Guo *et al*. 2020). Binarized neural networks (BNNs) have gained attention due to their high compression ratios, as they binarize weights and activations, replacing arithmetic operations with bitwise operations. To minimize quantization errors, methods like XNOR-Net, XNOR-Net++, and IR-Net (Qin *et al*. 2020) have introduced optimizations such as channel-wise scaling factors for the binarized value, normalization and balancing operations before binarization operation, so as to maximize the information entropy of weights and activations.Moreover, SI-BNN adopted flexible binary sets for activations of {0, +1} and weights of {–1, +1}, effectively reduce the quantization loss. SiMaN introduced an optimization objective function based on angle alignment and restricted weight binarization to {0, +1} according to the magnitude of the weights. Despite these advancements, the above BNNs still rely on fixed binary value sets, limiting their ability to adapt to diverse feature distributions in deep CNNs. AdaBin assumed that full-precision weights and activations followed a bell-shaped distribution with tails and dynamically determined the optimal binary quantization set for each layer to reduce quantization loss. However, its adaptive binary activation quantization was applied to the original input data, which may not exhibit a concentrated, bell-shaped distribution. As a result, even with an adaptively selected binary interval, mismatches in data distribution can lead to significant information loss.

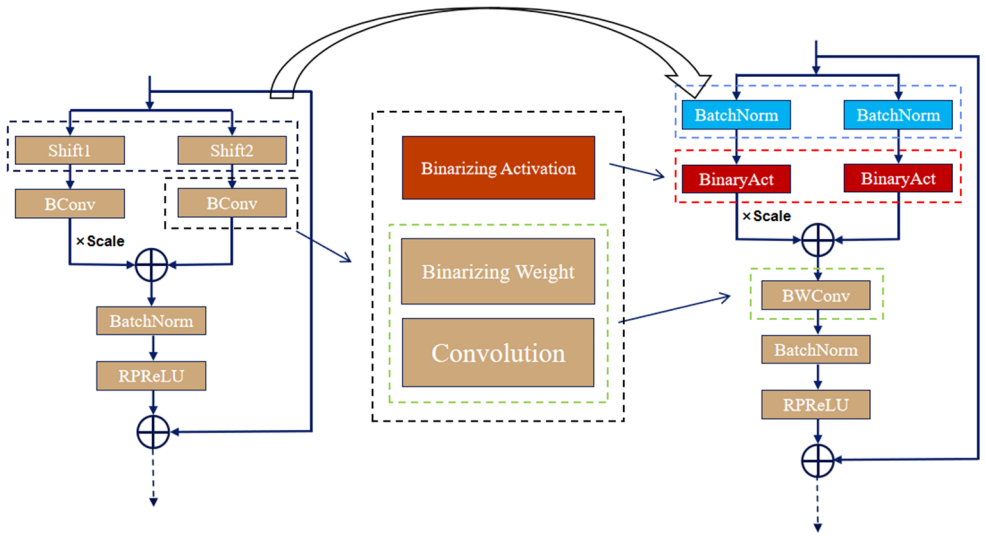
Regarding the improvement of the binary neural network structure, Bi-Real-Net (Liu *et al*. 2018) introduced a variant of the residual structure with shortcut connections before the binarization sign function, preserving real activation information and adding it to the output of subsequent convolution or BN layers. BNSC-Net (Wu *et al*. 2021), innovatively decomposing the 3×3 convolutional kernel directly into a series of 3×1 and 1×3 convolutional kernels double the number of skip connections in the network, enhancing the network's expressive capability. However, their weights are concentrated near zero, far from the binary values of ±1, leading to large quantization errors. MeliusNet and ReActNet developed dedicated BNN architectures, MeliusNet ingeniously integrates Dense Block and Improvement Block, employing dense connections between layers, where each layer transmits its own feature map to all subsequent layers, greatly facilitating feature reuse and significantly enhancing the network's ability to capture complex features. ReActNet introduces significant improvements by replacing the PReLU (He *et al*. 2015) activation function with a learnable RPReLU, and substituting the traditional sign function with RSign, which incorporates adjustable thresholds. These modifications optimize the feature maps between convolutional blocks. FracBNN extended ReActNet's structure by repositioning the BN layer and activation functions to balance positive and negative activations, improving expressiveness. It also combined fractional activation to blend full-precision and binarized convolutions, slightly increasing parameters while improving accuracy. Although the learnable RSign function can enhance the flexibility of the binarization process, it still binarizes the input information to the set {–1, +1}. Therefore, AdaBin addresses the limitation of binary intervals by introducing an adaptive binary quantization function. IE-Net introduced the Information Enhanced Binary Convolution (IE-BC) module with multiple sign functions and learnable thresholds to enrich binarized information. However, this added significant parameters due to the inclusion of dual-binary convolution modules. This highlights the need for lightweight adaptive prior activation binary quantization methods that retain multiple sign functions, enhancing expressiveness while optimizing feature distribution and minimizing parameter increase.

3. Principle of AOIE-Net

AOIE-Net adopts a dual-input information enhancement approach, replacing dual-input biases with dual Batch Normalization (BN) layers. It introduces a Binary Activation Information Enhancement Block (DBN-IEB) to optimize input distribution and retain more information after binary activation. Additionally, AOIE-Net separates adaptive binary activation from convolution and weight binarization, retaining only the binary activation step in the dual-input branches, while placing convolution and weight binarization in the summary path for a lightweight design.

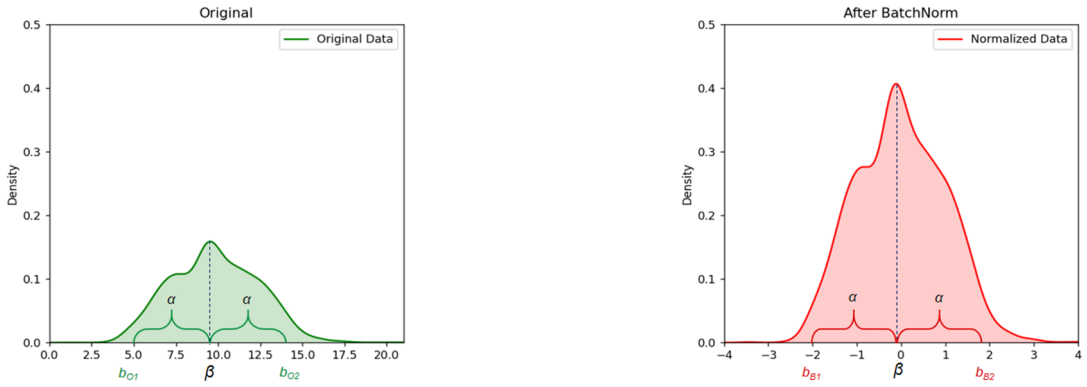
3.1 Binary activation Information Enhancement Block based on Dual BN Structure (DBN-IEB)

In binarized networks, activations are more sensitive to the binarization process than weights, and directly binarizing full-precision activations can lead to significant information loss. To address this, IE-Net introduced multiple binary activations using RSign functions with learnable thresholds across channels, improving the representational ability of binary neural networks. Inspired by this, we propose a Binary Activation Information Enhancement Block based on a Dual BN Structure (DBN-IEB). As shown in Fig. 2, the data-adaptive bias term in the introduced BN layer replaces the learnable threshold of



(a) The dual-input module (b) Original convolution block (c)The overall structure

Fig. 2 Block diagram of the implementation of ABA-IO



(a) Adaptive binary set selection of the original input (b) Adaptive binary set selection after BN

Fig. 3 Schematic diagram of the principle of improving the selection of adaptive binary sets after the input information passes through the BN layer

RSign and further optimizes the data distribution, enhancing the network's expressiveness.

As illustrated in Fig. 3, the BN layer tends to adjust the data distribution, making it more concentrated and approximating a normal distribution. Therefore, the adaptive binary set $\{b_{B1}, b_{B2}\}$ obtained after BN better represents the cluster centers of the input data compared to the binary set $\{b_{O1}, b_{O2}\}$ obtained without BN, and hence effectively reducing information loss.

3.2 Adaptive Binary Activation Independent Optimization Method (ABA-IO)

The binarization of activation values and weights, as well as the corresponding convolution operations, are usually bound together, as shown in Fig. 2(b). In this case, if we want to optimize binary activation using dual-input BN, we will need to perform convolution operations on both branches, which will lead to a doubling of the convolution computation in the network. Research has shown that binarizing activation values has a greater impact on performance than binarizing weights. Therefore, this work proposes a strategy that separates the binarization of weights, activations, and the convolution process. By extracting the adaptive binary activation method from the convolution and placing it behind the BN layer of the dual-input branch, the dual-input branches only perform lightweight adaptive binary activation optimization (ABA-IO), while weight binarization and the computationally intensive convolution are handled in the merged channel, as shown in Fig. 2. This approach effectively reduces the computational complexity of the dual-input information enhancement block, and independent optimization of adaptive activation has been achieved.

The merged binary activations will be input into the binarized weight convolution module, as shown in Eq. (1)

$$O = \mathrm{BWConv}(W^b, X^b_{APA}) \tag{1}$$

Among them, W^b is the binary weight, X^b_{APA} is the binary activation after being enhanced by dual input, and O is the output of the convolution operation.

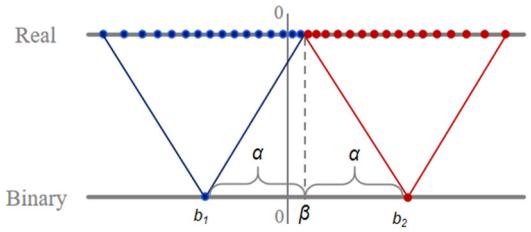


Fig. 4 Schematic diagram of the binarization mapping process of the adaptive method

The binarization methods for adaptive activation are shown in Eq. (2) and Fig. 4.

$$b = \begin{cases} b_1 = \beta - \alpha, & \beta > \alpha \\ b_2 = \beta + \alpha, & \beta \le \alpha \end{cases} \tag{2}$$

Among them, correspond to the two endpoints of the adaptive binary set in Fig. 3, α and β are the distance and the center of the set respectively. For the binarization of weights and activations, we use the same form of AdaBin, but with a different optimization strategy.

In the weight binarization method, better distribution matching is achieved by minimizing KLD. The KLD of the real-valued distribution and the binary distribution can be expressed as Eq. (3)

$$D_{KL}(P_r||P_b) = \int x \in \mathrm{W\&W}_b P_r(x)\, log \frac{P_r(x)}{P_b(x)} dx \tag{3}$$

where the $P_r(x)$ and $P_b(x)$ denote the distribution probability of real-valued weights and binarized weights. The binary distribution becomes more balanced by aligning the center of the binary distribution (the position of the mean) with the real-valued distribution

$$\beta_w = E(w) \approx \frac{1}{c \times k \times k} \sum_{m=0}^{c-1} \sum_{j=0}^{k-1} \sum_{i=0}^{k-1} W_{m,j,i} \tag{4}$$

Since the weights are usually normally distributed, and the two sides are symmetrical around the center, we can obtain α_w as in Eq. (5), with a detailed proof in the supplementary file of AdaBin (Tu *et al*. 2022).

$$\alpha_w = \frac{\| \mathrm{w} - \beta_w \|_2}{\sqrt{c \times k \times k}} \tag{5}$$

Binary weights are represented as

$$w_b = \alpha_w b_w + \beta_w, \quad b_w \in \{-1, +1\} \tag{6}$$

For activation binarization, optimization is performed through backpropagation of gradients, in order to ensure that the training process can converge, we need to clip out the gradient of large activation values in the backward as Eq. (7)

$$\frac{\partial L}{\partial a} = \frac{\partial L}{\partial a_b} * \mathbb{I}_{|\frac{a-\beta_a}{\alpha_a}| \le 1} \tag{7}$$

Where L is the output loss, a is the real-valued activation,

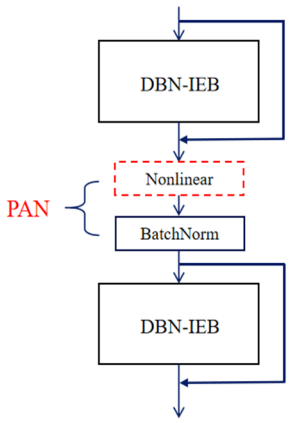


Fig. 5 Network structure with PAN added

a_b is the binary activation, $\mathbb{I}_{|x| \le 1}$ denotes the indicator function, which equals 1 if $|x| \le 1$ is true, and 0 otherwise. This functionality can be achieved by a composite function of hard tanh and sign

$$a_b = \alpha_a \times \mathrm{Sign}\left(\mathrm{Htanh}\left(\frac{a - \beta_a}{\alpha_a}\right)\right) + \beta_a \tag{8}$$

For simplicity, denote $g(\mathrm{x}) = \mathrm{Sign}(\mathrm{Htanh}(x))$, the gradients of α_a and β_a can be represented as

$$\begin{aligned} \frac{\partial L}{\partial \alpha_a} &= \frac{\partial L}{\partial a_b}\frac{\partial a_b}{\partial \alpha_a} \\ &= \frac{\partial L}{\partial a_b}\left(g\left(\frac{a-\beta_a}{\alpha_a}\right) - \frac{a}{\alpha_a} g'\left(\frac{a-\beta_a}{\alpha_a}\right)\right), \\ \frac{\partial L}{\partial \beta_a} &= \frac{\partial L}{\partial a_b}\frac{\partial a_b}{\partial \beta_a} = \frac{\partial L}{\partial a_b}\left(1 - g'\left(\frac{a-\beta_a}{\alpha_a}\right)\right), \end{aligned} \tag{9}$$

where $g'(\mathrm{x})$ is the derivative of $g(\mathrm{x})$. By setting the initial values of the center position β_a and the distance α_a to 0 and 1, the initial effect of binary quantization is equivalent to the sign function. Binary activations are represented as

$$a_b = \alpha_a b_a + \beta_a, b_a \in \{-1, +1\} \tag{10}$$

3.3 Network structure with PAN added

FracBNN proposed using BN (Batch Normalization) layer instead of RPLReLU between two convolution blocks, which can balance the number of positive and negative values in the activation values and effectively enhance the expressive ability of the network (Zhang *et al*. 2021). However, considering that the output of the first convolution block is a combination of activation values and shortcuts, using a BN layer cannot provide the ability of RPLReLU to change and reshape the feature maps. Therefore, in this work, we add an activation function before the BN layer to modify and reshape the feature map. We chose to use a nonlinear activation function (Nonlinear activation function, Nonlinear) to replace the original RPReLU to implement the pre-activated network structure (PAN), as shown in Fig. 5. The nonlinear activation

Table 1 Test results of different combinations on the CIFAR-10 dataset (%)

Methods	Flops (GFlops)	Accuracy
FracBNN	0.207	88.2
FracBNN + ABA-IO	0.185	88.5
FracBNN + DBN-IEB	0.330	88.6
FracBNN + PAN	0.218	88.7
AOIE-Net	0.374	90.6

function referenced is proposed in the AdaBin method. According to the research findings, the adaptive binary values of AdaBin are almost all positive except for a very few layers, which renders the nonlinearity of PReLU ineffective. To further enhance the representation of feature mappings, Maxout is utilized to achieve stronger nonlinearity. The structure of introducing a nonlinear activation function before the BN layer not only balances the number of positive and negative activation values in the feature maps but also ensures that the features undergo adaptive activation adjustments before batch normalization, helping to further improve the network's feature learning ability.

4. Experiments

To evaluate the effectiveness of each module in AOIE-Net, ablation experiments were conducted on the CIFAR-10 dataset, with comparisons made to full-precision classification networks and other binarized networks. The performance of AOIE-Net was further validated on the ImageNet and NEU-CLS datasets. All experiments were performed on a single NVIDIA RTX 4090 using the PyTorch framework.

For CIFAR10, the training set contains 50000 images and the test set contains 10000 images. We use random horizontal flipping and random cropping to enhance the input images. The model was trained for 300 epochs with a batch size of 128 and an initial learning rate of 1e-3, the learning rate decays linearly to 0 in each epoch.

Table 2 Comparison with state-of-the-art methods on the CIFAR-10 dataset (%)

Models	Years	W/A	Accuracy
ResNet-20	2016	32/32	91.7
DoReFa	2016	1/1	79.3
DSQ	2019	1/1	84.1
IR-Net	2020	1/1	86.5
RBNN	2020	1/1	87.8
FracBNN	2021	1/1	88.2
AdaBin	2022	1/1	88.2
IE-Net	2022	1/1	88.5
AOIE-Net (Ours)	-	1/1	**90.6**

For ImageNet, the training set contains 1.2 million images and the validation set contains 50000 images. We perform random cropping of each image to a size of 224×224 and randomly flip the image horizontally. Additionally, we apply random lighting variations to the input images. We train the model for 120 epochs at each step. The batch size is 256. The initial learning rate is 5e-4 and decays linearly to 0 in each epoch.

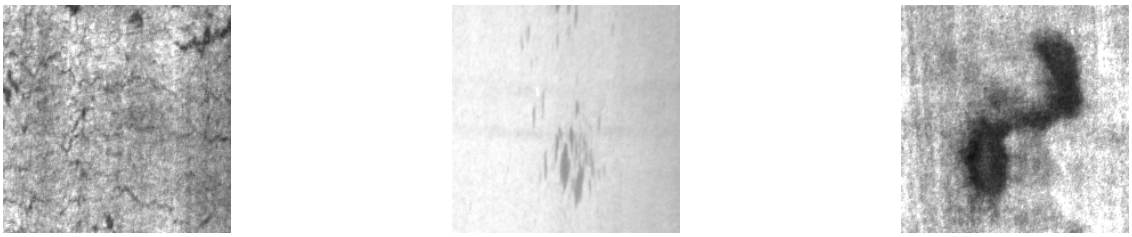
For NEU-CLS, this paper divides the dataset into training and testing sets at a ratio of 8:2. We trained this model for 90 epochs, with the initial learning rate set to 0.001, and decayed it by a factor of 10 after 45 epochs.

4.1 Ablation experiments and performance comparison experiments based on CIFAR-10

In this subsection, ablation experiments were conducted by adding the ABA-IO, DBN-IEB and PAN modules to the FracBNN baseline network to assess their impact on binarized classification neural networks. The results in Table 1 show that introducing the PAN module improves accuracy by 0.5% while maintaining computational

Table 3 Comparison with state-of-the-art methods on the ImageNet dataset (%)

Models	Years	W/A	Top-1	Top-5	Memory (Mbit)
ResNet-18	2016	32/32	73.3	91.3	374.1
BNN	2016	1/1	42.2	-	62.3
XNOR-Net	2016	1/1	51.2	73.2	33.7
Bi-RealNet-34	2018	1/1	62.2	83.9	33.6
Real-to-Binary Net	2020	1/1	65.4	86.2	-
IR-Net-34	2020	1/1	62.9	84.1	33.6
RBNN-34	2020	1/1	63.1	84.4	-
MeliusNet-29	2020	1/1	65.8	86.2	-
MeliusNet-42	2020	1/1	69.2	88.3	-
ReActNet-A	2020	1/1	69.4	-	26.6
FracBNN	2021	1/1	71.8	90.1	26.6
IE-Net-34	2022	1/1	64.6	85.2	33.8
AOIE-Net (Ours)	-	1/1	**72.1**	**90.2**	**26.6**



(a) Crack (b) Inclusion (c) Patch

(d) Pitted Surface (e) Scale (f) Scratch

Fig. 6 Samples of the NEU-CLS Dataset

complexity, highlighting the benefit of adding a nonlinear activation function before the BN layer to balance the number of positive and negative activation values in the features. Adding the ABA-IO module improves accuracy by 0.3% on average, while reducing floating-point computations. Further incorporation of the DBN-IEB module boosts accuracy to 88.6%, demonstrating that batch normalization enhances the information retained after binary activation. Ultimately, integrating these modules to form the AOIE-Net network results in a 2.4% accuracy improvement over FracBNN, achieving 90.6% accuracy for the binarized classification network.

Furthermore, a performance comparison was conducted between AOIE-Net and binarized classification networks with network structures based on ResNet-20, such as DoReFa (Zhou *et al*. 2016), DSQ (Gong *et al*. 2019), IR-Net, RBNN (Lin *et al*. 2020), FracBNN, AdaBin, and IE-Net, where W/A denotes the bit width of weights and activations. As can be seen from the experimental results in Table 2, Compared with other binary networks, AOIE-Net achieved the best performance, the accuracy rate reached 90.6%. In addition, the precision and recall rates of 90.7% and 90.6% were achieved, respectively. The gap in classification accuracy between it and the full-precision model narrowed to 1.1%. Meanwhile, compared with IE-Net, which currently has the highest accuracy, the accuracy of AOIE-Net is improved by 2.1%.

4.2 Performance verification of AOIE-Net on the ImageNet and NEU-CLS datasets

In this subsection, performance and model parameters of AOIE-Net were evaluated on ImageNet, and compared with recent high-performance BNNs as well as full-precision ResNet-18. As shown in Table 3, AOIE-Net outperforms networks such as IE-Net-34, MeliusNet, ReActNet-A, and FracBNN in both Top-1 and Top-5 accuracy on ImageNet. AOIE-Net achieves a Top-1 accuracy of 72.1% and Top-5 accuracy of 90.2%. It is comparable to FracBNN in terms of parameters but shows a 0.3% improvement in Top-1 accuracy. Compared to IE-Net, AOIE-Net achieves higher accuracy with fewer parameters.

Table 4 Comparison with state-of-the-art methods on the NEU-CLS dataset (%)

Models	Years	W/A	Flops (GFlops)	Accuracy
DRCDCT-Net	2022	32/32	-	99.7
IE-Net	2022	1/1	0.395	97.9
Adabin	2022	1/1	0.147	98.1
AOIE-Net (Ours)	-	1/1	0.166	99.4

Furthermore, AOIE-Net was applied to the NEU-CLS (Song and Yan 2013) dataset, which includes six defect categories for multi-class classification, offering a more comprehensive evaluation of the network's performance than binary classification. The categories of defects in NEU-CLS include scale, crack, inclusion, rust patch, scratch, and pitted surface, and 300 images per category, as shown in Fig. 6.

As shown in Table 4, AOIE-Net achieves 99.4% classification accuracy, and both precision and recall have also reached 99.4%, outperforming recent binarized networks like IE-Net and AdaBin, while maintaining a low computational cost. Its performance is close to the full-precision model DRCDCT-Net (Wang *et al*. 2022b), demonstrating its potential for practical applications in defect classification.

5. Conclusions

This paper presents AOIE-Net, an innovative Binarized Neural Network (BNN) architecture designed to address the limitations of traditional Deep Convolutional Neural Networks (DCNNs) in real-time defect detection tasks.

By binarizing network activations and parameters, AOIE-Net significantly compresses the space required and uses simple bitwise operations instead of floating-point

calculations, making it suitable for resource-constrained embedded platforms. It introduces two key contributions:

- the Binary Activation Information Enhancement Block based on Dual BN Structure (DBN-IEB), which standardizes input information through parallel batch normalization layers, optimizing data distribution before binary activation and improving the model's expressive ability.
- the Lightweight Adaptive Binary Activation Independent Optimization Method (ABA-IO), which independently optimizes binary activations prior to convolution, enhancing activation information without significantly increasing computational complexity.

Experimental results on CIFAR-10, ImageNet, and the NEU-CLS steel defect dataset show that AOIE-Net outperforms current state-of-the-art BNN models, achieving classification accuracy of 90.6%, 72.1%, and 99.4%, respectively. Through this efficient and high-precision binary classification network, combined with a cascade strategy, we can achieve fast and detailed classification processing of image regions. Specifically, this method uses a scanning frame to sequentially scan each block of the image and accurately classify each block with the help of a binary classification network. This process quickly identifies and eliminates areas that do not contain defects while precisely retaining critical areas that do contain defects. As a result, we not only significantly improve the speed of defect detection but also ensure the accurate localization of defect areas, laying a solid foundation for subsequent defect analysis and treatment. The promising results on real-world datasets demonstrate AOIE-Net's potential for practical applications in infrastructure monitoring.

Future works

In our future work, we plan to address the Possible limitations by conducting more comprehensive experiments and discussions on the robustness of AOIE-Net in real-world scenarios. Specifically, we aim to evaluate the performance of AOIE-Net in the presence of various types of noise, such as Gaussian noise and salt-and-pepper noise, to verify the noise resistance capability. Additionally, we will investigate how lighting changes, including different levels of brightness and color temperatures, affect the model's accuracy and reliability. Furthermore, we plan to explore the impact of shooting angle differences by testing the model on images captured from various perspectives and orientations. These experiments will provide insights into the generalization capabilities of AOIE-Net and help us identify areas where further improvements are needed to enhance its robustness in real-world applications.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under the Grants (Nos: 62271151 and W2421092).

References

Badea, M.S., Felea, I.I., Florea, L.M. and Vertan, C. (2016), "The use of deep learning in image segmentation, classification and detection", arXiv preprint. https://arxiv.org/abs/1605.09612

Bethge, J., Bartz, C., Yang, H., Chen, Y. and Meinel, C. (2020), "Meliusnet: Can binary neural networks achieve mobilenet-level accuracy?", arXiv preprint. https://arxiv.org/abs/2001.05936

Bulat, A. and Tzimiropoulos, G. (2019), "Xnor-net++: Improved binary neural networks", arXiv preprint. https://arxiv.org/abs/1909.13863

Chen, S.Y., Wang, Y.W., Ni, Y.Q. and Zhang, Y. (2024), "When Transfer Learning Meets Dictionary Learning: A New Hybrid Method for Fast and Automatic Detection of Cracks on Concrete Surfaces", *Struct. Control Health Monitor.*, **2024**(1), p. 3185640. https://doi.org/10.1155/2024/3185640

Choukroun, Y., Kravchik, E., Yang, F. and Kisilev, P. (2019), "Low-bit quantization of neural networks for efficient inference", In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 3009-3018. https://doi.org/10.1109/ICCVW.2019.00363

Ding, R., Liu, H. and Zhou, X. (2022), "IE-Net: Information-enhanced binary neural networks for accurate classification", *Electronics*, **11**(6), p. 937. https://doi.org/10.3390/electronics11060937

Duan, S., Zhang, M., Qiu, S., Xiong, J., Zhang, H., Li, C., Jiang, Q. and Kou, Y. (2024), "Tunnel lining crack detection model based on improved YOLOv5", *Tunnell. Undergr. Space Technol.*, **147**, p. 105713. https://doi.org/10.1016/j.tust.2024.105713

Gong, R., Liu, X., Jiang, S., Li, T., Hu, P., Lin, J., Yu, F. and Yan, J. (2019), "Differentiable soft quantization: Bridging full-precision and low-bit neural networks", *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4852-4861. https://doi.org/10.1109/ICCV.2019.00495

Guo, L., Li, R. and Jiang, B. (2020), "A cascade broad neural network for concrete structural crack damage automated classification", *IEEE Transact. Industr. Inform.*, **17**(4), 2737-2742. https://doi.org/10.1109/TII.2020.3010799

Guo, J.M., Yang, J.S., Seshathiri, S. and Wu, H. W. (2022), "A light-weight CNN for object detection with sparse model and knowledge distillation", *Electronics*, **11**(4), 575. https://doi.org/10.3390/electronics11040575

He, K., Zhang, X., Ren, S. and Sun, J. (2015), "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1026-1034. https://doi.org/10.1109/ICCV.2015.123

He, K., Zhang, X., Ren, S. and Sun, J. (2016), "Deep residual learning for image recognition", In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

He, Z., Su, C. and Deng, Y. (2024), "A novel MO-YOLOv4 for segmentation of multi-class bridge damages", *Adv. Eng. Inform.*, **62**, p. 102586. https://doi.org/10.1016/j.aei.2024.102586

Huang, H.W., Li, Q.T. and Zhang, D.M. (2018), "Deep learning based image recognition for crack and leakage defects of metro shield tunnel", *Tunnell. Undergr. Space Technol.*, **77**, 166-176. https://doi.org/10.1016/j.tust.2018.04.002

Huang, Y., Liu, Y., Liu, F. and Liu, W. (2024), "A lightweight feature attention fusion network for pavement crack segmentation", *Comput.-Aided Civil Infrastr. Eng.* https://doi.org/10.1111/mice.13225

Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R. and Bengio, Y. (2016), "Binarized neural networks", *Adv. Neural Inform. Process. Syst.*, p. 29.

Jin, T., Ye, X.W. and Li, Z.X. (2022), "Smartphone-based structural crack detection using pruned fully convolutional networks and edge computing", *Smart Struct. Syst., Int. J.*, **29**(1), 141-151. https://doi.org/10.12989/sss.2022.29.1.141

Ko, P., Prieto, S.A. and de Soto, B.G. (2021), "ABECIS: An automated building exterior crack inspection system using UAVs, open-source deep learning and photogrammetry", In: *ISARC Proceedings of the International Symposium on Automation and Robotics in Construction*: IAARC Publications, pp. 637-44. https://doi.org/10.22260/ISARC2021/0086

König, J., Jenkins, M., Mannion, M., Barrie, P. and Morison, G. (2022), "What's cracking? A review and analysis of deep learning methods for structural crack segmentation, detection and quantification", arXiv preprint. https://arxiv.org/abs/2202.03714

Li, J., Nie, Q., Fu, W., Lin, Y., Tao, G., Liu, Y. and Wang, C. (2024), "LORS: Low-rank Residual Structure for Parameter-Efficient Network Stacking", In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15866-15876.

Lin, M., Ji, R., Xu, Z., Zhang, B., Wang, Y., Wu, Y., Huang, F. and Lin, C.W. (2020), "Rotated binary neural network", *Adv. Neural Inform. Process. Syst.*, **33**, 7474-7485. https://doi.org/10.48550/arXiv.2009.13055

Lin, M., Ji, R., Xu, Z., Zhang, B., Chao, F., Lin, C. W. and Shao, L. (2022), "Siman: Sign-to-magnitude network binarization", *IEEE Transact. Pattern Anal. Mach. Intell.*, **45**(5), 6277-6288. https://doi.org/10.1109/TPAMI.2022.3212615

Liu, Z., Wu, B., Luo, W., Yang, X., Liu, W. and Cheng, K.T. (2018), "Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm", *Proceedings of the European Conference on Computer Vision* (*ECCV*), pp. 722-737. https://doi.org/10.1007/978-3-030-01267-0_44

Liu, C., Ding, W., Xia, X., Zhang, B., Gu, J., Liu, J., Ji, R. and Doermann, D. (2019), "Circulant binary convolutional networks: Enhancing the performance of 1-bit dcnns with circulant back propagation", In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2691-2699.

Liu, Z., Shen, Z., Savvides, M. and Cheng, K.T. (2020), "Reactnet: Towards precise binary neural network with generalized activation functions", In: *Proceedings of Computer Vision–ECCV 2020*: *16th European Conference*, Glasgow, UK, August, pp. 143-159. https://doi.org/10.1007/978-3-030-58568-6_9

Long, J., Shelhamer, E. and Darrell, T. (2015), "Fully convolutional networks for semantic segmentation", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431-3440. https://doi.org/10.1109/CVPR.2015.7298965

Martinez, B., Yang, J., Bulat, A. and Tzimiropoulos, G. (2020), "Training binary neural networks with real-to-binary convolutions", arXiv preprint. https://arxiv.org/abs/2003.11535

Maruthi, M. and Lee, D.E. (2024), "Enhancement of concrete crack detection using U-Net", In: *International Conference on Construction Engineering and Project Management*, pp. 152-159. https://doi.org/10.6106/ICCEPM.2024.0152

Qin, H., Gong, R., Liu, X., Shen, M., Wei, Z., Yu, F. and Song, J. (2020), "Forward and backward information retention for accurate binary neural networks", *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* pp. 2250-2259. https://doi.org/10.1109/CVPR42600.2020.00232

Qiu, Q. and Lau, D. (2023), "Real-time detection of cracks in tiled sidewalks using YOLO-based method applied to unmanned aerial vehicle (UAV) images", *Automat. Constr.*, **147**, p. 104745. https://doi.org/10.1016/j.autcon.2023.104745

Rastegari, M., Ordonez, V., Redmon, J. and Farhadi, A. (2016), "Xnor-net: Imagenet classification using binary convolutional neural networks", In: *European Conference on Computer Vision*, pp. 525-542. https://doi.org/10.1007/978-3-319-46493-0_32

Ren, S., He, K., Girshick, R. and Sun, J. (2016), "Faster R-CNN: Towards real-time object detection with region proposal networks", *IEEE Transact. Pattern Anal. Mach. Intell.*, **39**(6), 1137-1149. https://doi.org/1137-1149.10.1109/TPAMI.2016.2577031

Ruan, J., Xie, M., Gao, J., Liu, T. and Fu, Y. (2023), "Ege-unet: an efficient group enhanced unet for skin lesion segmentation", In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 481-490. https://doi.org/10.1007/978-3-031-43901-8_46

Schanz, T., Tenscher-Philipp, R., Marschall, F. and Simon, M. (2023), "AI-powered multi-class defect segmentation in industrial CT data", eJNDT.

Shen, M., Liu, X., Gong, R. and Han, K. (2020), "Balanced binary neural networks with gated residual", In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), pp. 4197-4201. https://doi.org/10.1109/ICASSP40776.2020.9054599

Song, K. and Yan, Y. (2013), "A noise robust method based on completed local binary patterns for hot-rolled steel strip surface defects", *Appl. Surf. Sci.*, **285**, 858-864. https://doi.org/10.1016/j.apsusc.2013.09.002

Tang, W., Wu, R.T. and Jahanshahi, M.R. (2022), "Crack segmentation in high-resolution images using cascaded deep convolutional neural networks and Bayesian data fusion", *Smart Struct. Syst., Int. J.*, **29**(1), 221-235. https://doi.org/10.12989/sss.2022.29.1.221

Tse, K.W., Pi, R., Yang, W., Yu, X. and Wen, C.Y. (2024), "Advancing UAV-based Inspection System: The USSA-Net Segmentation Approach to Crack Quantification", *IEEE Transact. Instrum. Measur.*, **73**. https://doi.org/10.1109/TIM.2024.3418073

Tu, Z., Chen, X., Ren, P. and Wang, Y. (2022), "Adabin: Improving binary neural networks with adaptive binary sets", In: *European Conference on Computer Vision*, pp. 379-395. https://doi.org/10.1007/978-3-031-20083-0_23

Wang, P., He, X., Li, G., Zhao, T. and Cheng, J. (2020), "Sparsity-inducing binarized neural networks", In: *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**(07), pp. 12192-12199. https://doi.org/10.1609/aaai.v34i07.6900

Wang, X., Ren, H. and Wang, A. (2022a), "Smish: A novel activation function for deep learning methods", *Electronics*, **11**(4), 540. https://doi.org/10.3390/electronics11040540

Wang, J., Zhang, Q. and Liu, G. (2022b), "DRCDCT-net: a steel surface defect diagnosis method based on a dual-route cross-domain convolution-transformer network", *Measur. Sci. Technol.*, **33**(9), p. 095404. https://doi.org/10.1088/1361-6501/ac6fb2

Wang, Z., Leng, Z. and Zhang, Z. (2024), "A weakly-supervised transformer-based hybrid network with multi-attention for pavement crack detection", *Constr. Build. Mater.*, **411**, p. 134134. https://doi.org/10.1016/j.conbuildmat.2023.134134

Wu, L., Lin, X., Chen, Z., Huang, J., Liu, H. and Yang, Y. (2021), "An efficient binary convolutional neural network with numerous skip connections for fog computing", *IEEE Internet Things J.*, **8**(14), 11357-11367. https://doi.org/10.1109/JIOT.2021.3052105

Xie, X., Bai, L. and Huang, X. (2021), "Real-time LiDAR point cloud semantic segmentation for autonomous driving", *Electronics*, **11**(1), p. 11.

https://doi.org/10.3390/electronics11010011

Zhang, Y., Pan, J., Liu, X., Chen, H., Chen, D. and Zhang, Z. (2021), "FracBNN: Accurate and FPGA-efficient binary neural networks with fractional activations", In: *The 2021 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pp. 171-182. https://doi.org/10.1145/3431920.3439296

Zhong, X., Peng, X., Duan, B., Zhou, K., Li, Q. and Zhao, C. (2024), "UAV-based bridge crack discovery via deep learning and tensor voting", *Smart Struct. Syst., Int. J.*, **33**(2), 105-118. https://doi.org/10.12989/sss.2024.33.2.105

Zhou, S., Wu, Y., Ni, Z., Zhou, X., Wen, H. and Zou, Y. (2016), "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients", arXiv preprint. https://arxiv.org/abs/1606.06160

Zhu, Y. and Newsam, S. (2017), "Densenet for dense flow", In: *2017 IEEE International Conference on Image Processing* (*ICIP*), pp. 790-794. https://doi.org/10.1109/ICIP.2017.8296389

Zhuang, B., Shen, C., Tan, M., Liu, L. and Reid, I. (2019), "Structured binary neural networks for accurate image classification and semantic segmentation", In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 413-422. https://doi.org/10.1109/CVPR.2019.00050