Efficient Anomaly Detection of Structural Health Monitoring Based on Improved MobileViTv3

Shuoting Zhao [1,2], and Zhiyi Tang[1,2*]

[1] Faculty of Civil Engineering and Mechanics, Kunming University of Science and Technology, Kunming, 650500, China
[2] Intelligent Infrastructure Operation and Maintenance Technology Innovation Team of Yunnan Provincial, Department of Education, Kunming University of Science and Technology, Kunming 650500, China
tang@kust.edu.cn

Abstract: Actual structural health monitoring systems inevitably generate abnormal monitoring data, which affects the detection of structural damage. To improve the efficiency and accuracy of detecting various types of abnormal data in monitoring, this study proposes a structural health monitoring anomaly data diagnosis method based on an improved MobileViTv3 algorithm. This method aims to maintain high-precision monitoring while significantly reducing the model's weight and prediction time. To achieve this, two strategies are introduced: first, reducing the number of main network blocks to decrease the model's parameters; second, employing an increased expansion factor strategy to enhance the model's ability to capture features at different scales. These improvements allow the algorithm to better meet the needs of anomaly data detection. Additionally, a comparative analysis was conducted on a large bridge structural health monitoring system dataset against mainstream models such as VGG, ResNet, MobileNet, and EfficientNet. The study results show that the improved MobileViTv3 algorithm achieved the highest recognition accuracy of 96.8% in anomaly data detection tasks, reducing the model size to 1.4 MB, and also demonstrated significant advantages in training and prediction efficiency.

Keywords: Structural health monitoring, Data anomaly detection, MobileViTv3, Lightweight algorithm.

1 Introduction

In recent decades, Structural Health Monitoring (SHM) systems have been widely applied to long-span bridges [1-2]. By monitoring external loads, environmental conditions, and structural responses in real-time, the structural performance of bridges can be assessed, and damage can be diagnosed [3-4]. Dynamic response monitoring is crucial in SHM of long-span bridges. However, in practical applications, dynamic response monitoring data inevitably contain numerous anomalies due to sensor failures, transmission faults, radio interference, sensor component aging, and other

W. Guo et al. (eds.), *Proceedings of the 3rd International Conference on Green Building, Civil Engineering and Smart City (GBCESC 2024)*, Advances in Engineering Research 264,
https://doi.org/10.2991/978-94-6463-728-1_74

reasons [5]. Anomalous monitoring data cannot provide any useful information for dynamic characteristic identification and structural performance evaluation but may lead to false or missed reports of structural damage or degradation. Therefore, assessing the data quality of dynamic response monitoring data is crucial, and low-quality data should be removed from the monitoring database.

Traditional anomaly detection methods mainly include statistical probability-based and clustering-based approaches [6-9]. In general, these traditional anomaly detection methods are simple to operate; however, they can only be used to detect outliers.

In recent years, with the rapid development of artificial intelligence technologies, deep learning and machine vision technologies have been widely applied in the field of SHM [10-14]. Despite significant efforts in anomaly detection, the following deficiencies still exist: existing machine learning-based anomaly data classification frameworks solely use general CNNs as feature extractors, such as VGG, ResNet, MobileNet, and EfficientNet. These were originally designed to classify thousands of object types. However, typical anomaly data types are few, and general models are large, leading to low efficiency when used as backbone networks for monitoring anomaly data and increased hardware resource demands. This limits their use in anomaly detection, especially when handling large datasets and running on small devices with limited computational capacity.

However, due to the inherent locality of convolution operations, CNNs are limited to focusing on local information and find it challenging to capture global dependencies within images. Consequently, they cannot effectively integrate contextual information from the entire input image, leading to a higher false detection rate. In contrast, the attention mechanism in Transformer models can fully exploit the global contextual information in images, effectively overcoming the aforementioned limitations of CNNs.

Accordingly, this paper proposes an improved lightweight MobileViTv3 network [14]. This network combines the strengths of CNNs and Transformers to achieve efficient feature extraction. Two strategies are proposed to improve the original MobileViTv3 algorithm from different perspectives: First, the number of main network blocks is reduced to decrease the model's parameter count. Second, to enhance the model's ability to capture features at different scales, a strategy of increasing the expansion factor is adopted.

2 Methodology

MobileViTv3 is a lightweight neural network model designed for mobile vision tasks, achieving efficient feature extraction by combining the characteristics of CNNs and Transformers. In MobileViTv3, as shown in Fig .1, the front end of the network uses MobileNetV2(MV2) blocks, while the back end uses MobileViT blocks. It has been demonstrated that MobileViTv3, due to its structural features, significantly reduces training time and decreases the number of model parameters and the amount of training compared to state-of-the-art models [14].

2.1 MobileNetV2 block

Down-sampling in the early stages is a common strategy among most lightweight deep learning-based classification models, including those that incorporate CNNs and Transformers, to optimize computational efficiency. The goal is to reduce the size of feature maps at an early stage, effectively reducing the computational load of subsequent layers and accelerating the training and inference speed of the model. The MV2 block is a typical example, showcasing excellent performance in mobile and embedded vision tasks due to its clever design and efficient computation. The core of the MV2 block is the inverted residual structure, a special type of residual network structure. In standard residual networks, convolutional layers typically first expand the number of channels and then reduce them; however, in the inverted residual structure, this process is reversed.

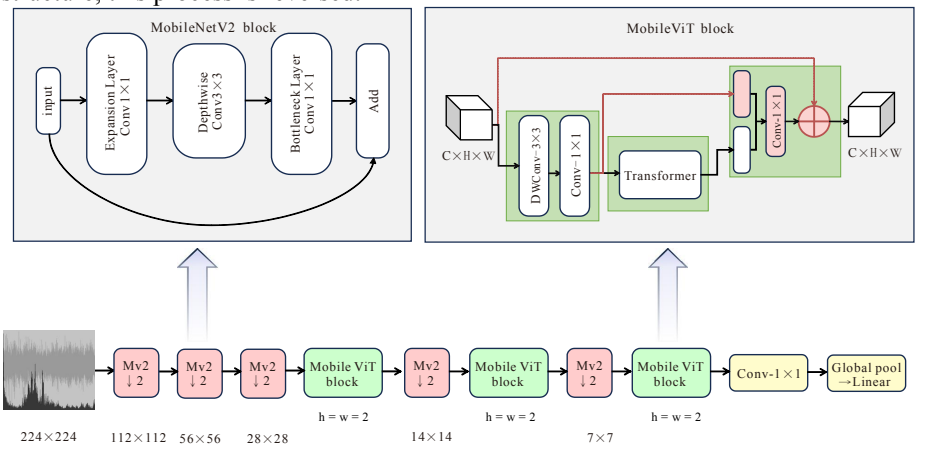


Fig. 1. Overview of the MobileVITv3 architecture

As shown in Fig. 1: (1) 1x1 Pointwise Convolution (Expansion Layer): This layer employs a 1x1 convolution kernel to perform pointwise convolution on the input feature map. Each input channel generates an output channel, thus increasing the channel count of the feature map and enhancing the capability for feature representation. (2) Depth-wise Separable Convolution: This method performs convolution operations separately on each channel of the input. The filters of each channel operate independently and do not share weights with filters of other channels, a process referred to as depth-wise convolution. Its primary purpose is to reduce the channel count. (3) 1x1 Linear Pointwise Convolution (Bottleneck Layer) is a key feature of the inverted residual block in the MV2 block. It uses a 1x1 convolution kernel to perform pointwise convolution, mapping the expanded feature map back to the original low-dimensional space through a linear transformation. The purpose is to reduce the number of parameters and the computational load of the model while maintaining the network's classification ability.

2.2 MobileViT block

The MobileViT block is a core component of the MobileViTv3 architecture, combining the characteristics of convolution and Transformer to efficiently encode local and global information. The MobileViT block receives an input tensor X with dimensions H×W×C, where H and W represent the height and width of the tensor, respectively, and C represents the number of channels. First, X passes through an $n \times n$ Depth-wise Separable Convolution (where n is the kernel size) to encode the local spatial information of the input tensor. Then, a pointwise convolution layer is used to project the convolved tensor into a higher dimensional space (d-dimensional space), where d is greater than the number of input channels C, enabling the model to capture more complex features in the higher-dimensional space. Next, the output of the $n \times n$ convolution layer, denoted as X_L, is unfolded into N non-overlapping image patches, denoted as X_U. Each image patch X_U is then processed with a self-attention mechanism, allowing each pixel to gather information from other regions of the image through self-attention to obtain X_G. Here is calculation as follows:

$$X_G = \text{Transformer}(X_U) \tag{1}$$

The calculation formula for attention is as follows:

$$Attention(Q, K, V) = \sigma(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

Here, σ is the softmax function; Q, K, and V represent the Query, Key, and Value matrices obtained by linearly transforming the input features.

The global information processed by the self-attention mechanism, X_G, is folded back to the original H×W dimensions, yielding X_F. During the feature fusion stage, the output of the transformer layer, X_F, which contains the global information feature tensor, is mapped back to the original number of channels through a 1×1 pointwise convolution layer, X_L. This is then concatenated or added to the input tensor to achieve an effective fusion of local and global features.

$$X_{out} = X_L + X_U \tag{3}$$

3 Improvement Strategy

3.1 Strategy 1: Reducing Model Depth

Increasing the depth of the model enhances its feature extraction capabilities for anomalous data, but may compromise efficiency; conversely, reducing the model's depth might decrease its feature extraction capability while lowering its parameter count. This section investigates the optimal model depth to balance accuracy and efficiency in anomaly data classification. The initial model structure is depicted in Fig. 1 and Table 1 details its configuration and parameters. By adjusting the number of MV and MobileViT blocks in the initial network, four variants were derived: model 1 reduces both MV and MobileViT blocks to one layer; model 2 reduces only the MV block to one layer; model 3 increases the MobileViT block to five layers; and

model 4 increases both blocks to five layers. Each variant maintains identical settings and parameters across all network blocks, differing only in the number of MV and MobileViT blocks.

Table 1. Initial model structure and main parameters. The symbol "↓" in the table indicates down-sampling.

Network block	Size	Layer	Expansion Factor
Image	224×224		
MV2, ↓2	112×112	1	2
MV2, ↓2	56×56	3	2
MV2, ↓2	28×28	2	2
MobileViT block	28×28		
MV2, ↓2	14×14	4	2
MobileViT block	14×14		
MV2, ↓2	7×7	3	2
MobileViT block	7×7		
Conv-1×1	1×1		

Fig. 2 illustrates the comparative results in accuracy, loss, training time, and model size across these models. As shown in Fig. 2(a), models with increased depth (models 3 and 4) achieved higher accuracy on both training and validation sets than the initial model, while those with reduced depth (models 1 and 2) exhibited lower accuracy. These findings suggest that a greater number of network blocks may marginally enhance classification accuracy, whereas fewer blocks tend to diminish it. Fig. 2(b) details the differences in training time and model size; the initial model required 453.2 s and had a size of 2.62 MB. Relative to this, model 1 exhibited a 37.1% decrease in training time and a 53.1% decrease in size, while model 4 showed increases of 45.0% in time and 33.4% in size. Despite these variations, all five models demonstrated nearly identical performance in terms of accuracy and loss. Consequently, reducing the number of MV and MobileViT blocks significantly decreases both training time and model size, albeit at a slight cost to accuracy.

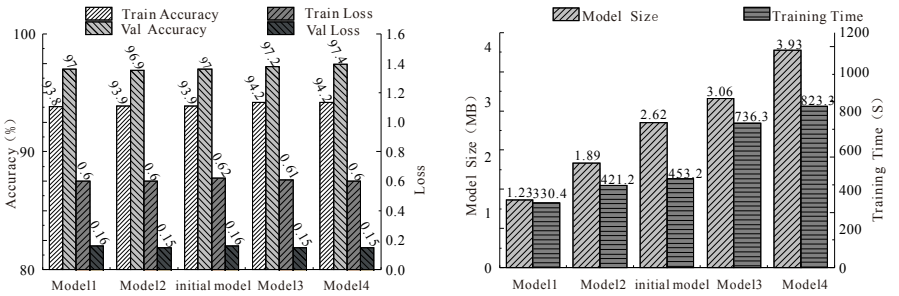
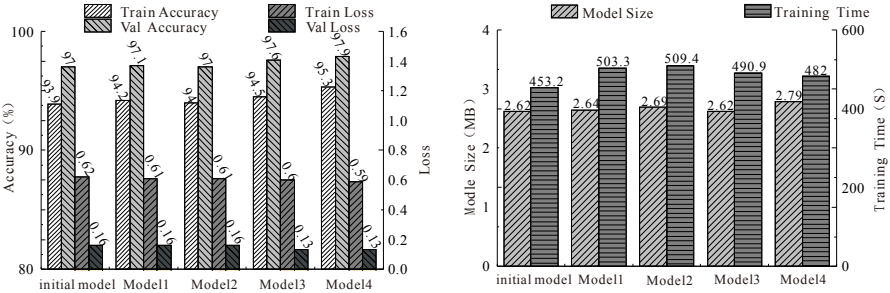


Fig. 2. Comparison of strategy 1 models on the training set: model 1 has reduced network blocks; model 2 has reduced MV blocks; model 3 has increased MV blocks; model 4 has increased network blocks.

3.2 Strategy 2: Increasing the Expansion Factor of Network Blocks

Increasing the expansion factors of MV and MobileViT blocks can enhance accuracy but also extend training duration. To determine which modification optimally improves anomaly data classification, several tests were executed. Modifications were made to the initial model's MV and MobileViT blocks, resulting in four variations: model 1, where the expansion factor of the front-end MV block was increased to 6; model 2, with the back-end MV block's expansion factor raised to 6; model 3, where the front-end MobileViT block's expansion factor was increased to 6; and model 4, where the back-end MobileViT block's expansion factor was elevated to 6. Fig. 3 illustrates the differences in accuracy, loss, training time, and model size between the four modified models and the initial model. As depicted in Fig. 3(a), the modified models consistently outperform the initial model in both training and validation accuracy, indicating that higher expansion factors improve classification accuracy but at the cost of increased training time and model size. Specifically, training accuracy improved from 94.2% in model 1 to 95.3% in model 4, while validation accuracy rose from 97.1% to 97.9%. Concurrently, both training and validation losses decreased across the models. Thus, modifications to the back-end network blocks were more effective at enhancing anomaly data classification capabilities than those made to the front-end. Fig. 3(b) presents a comparison of training times and model sizes; all modified models requiring more time than the initial model, there was a noticeable reduction from 503.3 s in model 1 to 482 s in model 4, suggesting a less pronounced impact on efficiency when enhancing the back-end network blocks. trend underscores a greater improvement in classification capabilities and efficiency closer to the back-end network blocks. Hence, enhancing the back-end network blocks may slightly reduce training efficiency but significantly boosts accuracy.



(a) Accuracy and loss (b) Training time and model size

Fig. 3. Comparison of strategy 2 models on the training set: model 1 enhances the front-end MV blocks; model 2 enhances the back-end MV blocks; model 3 enhances the front-end MobileViT blocks; model 4 enhances the back-end MobileViT blocks.

In summary, two strategies for optimizing models in anomaly data classification have emerged: (1) Reducing the number of main network blocks to strike an optimal balance between accuracy and efficiency, (2) Increasing the expansion factor of back-end network blocks to enhance accuracy with shorter training durations. The

improved MobileViTv3 model, along with its structural and principal parameters, is detailed in Table 2.

Table 2. Improved MobileViTv3 model structure and main parameters. The symbol "↓" in the table indicates down-sampling.

Network block	Size	Layer (Strategy 1)	Expansion Factor (Strategy 2)
Image	224×224		
MV2, ↓2	112×112	1	2
MV2, ↓2	56×56	1	2
MV2, ↓2	28×28	1	2
MobileViT block	28×28		
MV2, ↓2	14×14	1	2
MobileViT block	14×14		
MV2, ↓2	7×7	1	6
MobileViT block	7×7		
Conv-1×1	1×1		

4 Dataset Construction

Using the annual data from 38 accelerometers of a health monitoring system of a long-span cable-stayed bridge in 2012, the data were segmented hourly, visualized in both time and frequency domains, and converted into RGB images. These images were then superimposed to form dual-channel composite images, resulting in a total of 333,792 (38×24×366) images data, where 38 represents the number of channels, 24 represents the hours per day, and 366 represents the number of days. Each composite image has a pixel size of 224×224. Seven data types were defined: normal, missing, minor, outlier, square, trend, and drift. Samples of time and frequency domains for each category are shown in Fig. 4. The data amounts for all categories are summarized in Table 3.

Table 3. Dataset Information

Data Types	Number	Proportion (%)
Normal	211,246	63.29
Missing	17,061	5.11
Minor	55,483	16.62
Outlier	12,390	3.71
Square	16,900	5.06
Trend	17,812	5.34
Drift	2,900	0.87
Total	333,792	100

Training with consecutive data might lead to poor classifier robustness; therefore, 1% (3,335 images) of the dual-channel dataset was randomly selected for training and validating the deep learning model, while the remaining 99% (330,457 images) were used as a large-scale test set to evaluate the model's robustness.

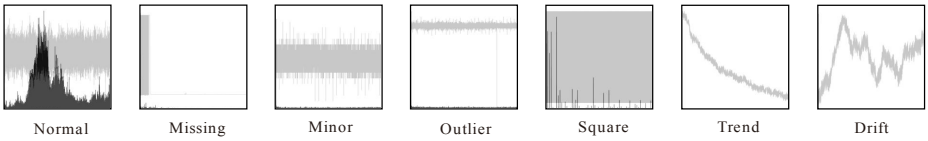


Fig. 4. Time-frequency domain diagrams of 7 data types

5 Experiments and analysis

5.1 Experimental environment and hyperparameter settings

The computational setup includes a 12th generation Intel i7-12700 processor with 64GB of system memory and an NVIDIA GeForce RTX 3060 graphics card equipped with 12GB of video memory. The software environment comprises a 64-bit Windows 11 operating system, utilizing Python as the high-level programming language within the Pytorch deep learning framework.

Hyperparameters were set as detailed in Table 4; notably, the training epochs and batch size were configured to 20 and 8, respectively, for the improved MobileViTv3 and the general CNNs employed in structural health monitoring for anomaly diagnosis. The Adam optimizer was uniformly used across models with an identical learning rate.

Table 4. Hyperparameter settings of computational models

Optimizer	Adam
Learning rate	0.0002
Epoch	20
Loss function	Cross entropy
Batch size	16

5.2 Evaluation indicators

The evaluation metrics for classification ability are generally measured by training accuracy and validation accuracy, while the evaluation metrics for model efficiency are quantified by model size and training time. Training accuracy represents the model's ability to classify anomalous data within the training set. During validation, the model inherits the weights from training, and the validation set is independent of the training set. Therefore, the validation accuracy on the validation set is a more reliable indicator of the model's generalization ability.

The confusion matrix and its related metrics on the test set are also used to evaluate the model's classification ability. Table 5 lists the confusion matrix for binary

classification. Based on the confusion matrix, evaluation metrics can be defined. In the table, *TP* (True Positive), *FP* (False Positive), *TN* (True Negative), and *FN* (False Negative) refer to the correspondence between the test results and actual classifications. In anomaly data detection, *TP* indicates the number of samples that are positive and predicted

Table 5. Binary confusion matrix

	Predicted positive	Predicted negative
Actual positive	True positive (*TP*)	False negative (*FN*)
Actual negative	False positive (*FP*)	True negative (*TN*)

as positive; *FP* indicates the number of samples that are negative but predicted as positive; *TN* indicates the number of samples that are negative and predicted as negative; *FN* indicates the number of samples that are positive but predicted as negative. Multiple metrics can be used to evaluate the performance of anomaly data classification. The closer the metric is to 1, the better the classification performance. The mathematical definitions of these metrics are shown in formulas (4) to (6).

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

5.3 Comparison of training and validation results

The monitoring data images were classified using models including MobileViTv3, EfficientNetV2, MobileNetV3, and VGG16. During training, the training accuracy and validation accuracy of the five models were calculated. As shown in Fig. 5, and Fig. 6, the improved MobileViTv3 demonstrated the highest convergence speed. As shown in Table 6, the model size and training time of the improved MobileViTv3 are 1.4MB and 344.9 s, respectively, which are 46.2% smaller and 23.9% shorter than the original MobileViTv3; and 76.7% smaller and 48.0% shorter than MobileNetV3. Additionally, the improved MobileViTv3 is significantly more efficient than all other CNNs, with the smallest model size and the shortest training time. The training accuracy and validation accuracy of the improved MobileViTv3 reached 94.7% and 98.0%, respectively, the best performance among all models.

The test set in the dataset used a confusion matrix to evaluate the generalization ability of all models, as shown in Fig. 7. In the confusion matrix, each column represents the true labels, and each row represents the predicted labels, in the order of normal, missing, minor, outlier, square, trend, and drift. The rightmost column of the confusion matrix is the recall, representing the percentage of actual positive cases correctly identified as positive. The bottom row of the confusion matrix is the precision, indicating the percentage of true positive cases among all cases predicted as

positive. The bottom-right corner of the confusion matrix shows the overall accuracy of the model. As demonstrated in the results, the improved MobileViTv3 exhibits the best overall performance; for instance, it correctly predicted 1971 drift samples, compared to 1914

Table 6. Classification accuracy, model size, and training time of 5 models after 20 epochs.

Model	Accuracy		Model size/MB	Train time/s
	Train	Val		
Improved MobileViTv3	94.7%	98.0%	1.4	344.9
MobileViTv3	94.0%	97.0%	2.6	453.2
EfficientNetV2	93.9%	96.1%	79.7	1794.6
MobileNetV3	93.0%	96.5%	6.0	663.1
VGG16	92.3%	95.3%	524.0	1580.3

for the original MobileViTv3, 1177 for EfficientNetV2, 891 for MobileNetV3, and 844 for VGG16. The overall accuracy of the improved MobileViTv3 reaches 96.8%, compared to 95.8% for the original MobileViTv3, 96.2% for EfficientNetV2, 96.1%for MobileNetV3, and 95.5% for VGG16. The results indicate that the proposed improved MobileViTv3 demonstrates excellent classification capability in anomaly data classification tasks while maintaining high operational efficiency.

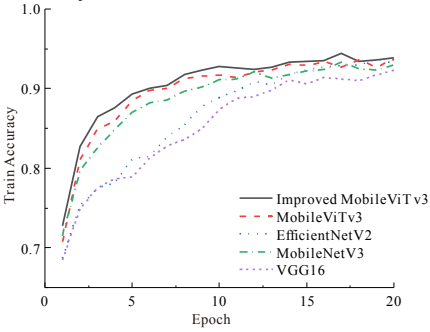


Fig. 5. Train accuracy curve

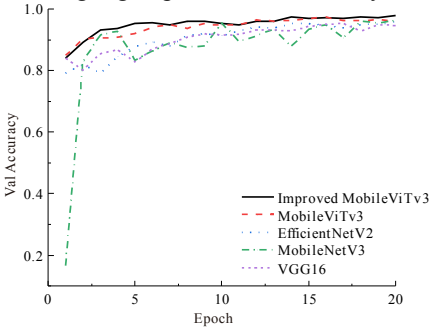


Fig. 6. Val accuracy curve

206225		1614	971	284	40		98.6 %
11	16676		154	6	41	3	98.7 %
1228		51806	668	300	921	6	94.3 %
864	57	1153	10121	68	1	3	82.5 %
176		172	32	16352			97.8 %
1	63	331	184	6	16825	225	95.4 %
1		25	16		858	1971	68.7%
98.9%	99.3 %	94.0 %	83.3 %	96.1 %	90.0 %	89.3 %	96.8 %

(a) Improved MobileViTv3

203713		3340	1703	272	6		97.4 %
4	16464		416		6	1	97.5 %
1015		51315	2233	230	132	4	93.4 %
754	70	212	11220	8		4	91.4 %
281		388	102	15960			95.4 %
2	82	1085	279	2	15844	340	89.8 %
2		18	167		771	1914	66.6%
99.0 %	99.0 %	91.0 %	69.6 %	96.9 %	94.5 %	84.6 %	95.8 %

(b) MobileViTv3

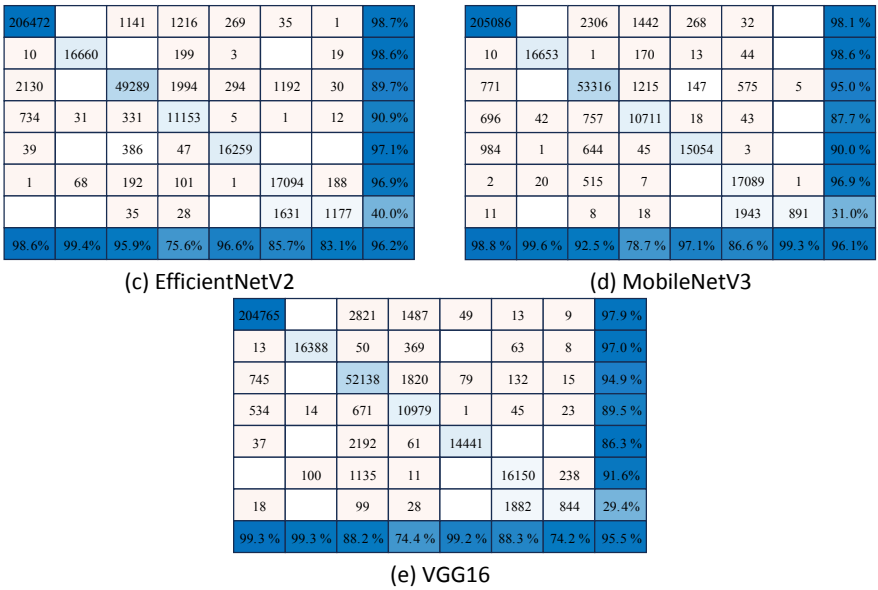


Fig. 7. Confusion matrix of models on the test set. The matrix is arranged from top to bottom and from left to right in the following order: normal, missing, minor, outlier, square, trend, and drift.

6 Conclusions

This paper is based on the annual monitoring acceleration data from 2012 of a large-span cable-stayed bridge. Using the MobileViTv3 algorithm, two strategies were introduced to improve the network's accuracy in classifying anomalous data and to reduce the network parameters. The following conclusions were drawn:

(1) To reduce the computational load and processing time of the model, two strategies were introduced that significantly reduce the weight file size and training time of the MobileViTv3 model while maintaining high accuracy. The improved MobileViTv3 model is 46.2% smaller in size and reduces training time by 23.9% compared to the initial MobileViTv3 model.

(2) Comparative experiments were conducted to confirm the advantages of the improved MobileViTv3 in anomaly data classification. The results show that, compared to general CNNs such as EfficientNetV2, MobileNetV3, and VGG16, the improved MobileViTv3 model has the smallest size at 1.4 MB, achieves the highest efficiency with a training time as low as 344 s, and its accuracy surpasses that of the general CNNs.

(3) The model was applied to the analysis of a year's worth of monitoring data from a long-span cable-stayed bridge in 2012, achieving an anomaly data accuracy rate of 96.8%. This indicates that the constructed network has high accuracy and strong robustness in monitoring data classification tasks. However, the proposed modifications to MobileViTv3 may not achieve high accuracy in other classification

tasks, as the deep-stage structural features of the improved MobileViTv3 were specifically developed for the classification of monitoring data.

Acknowledgement

This research was supported by the Yunnan Fundamental Research Projects (Grant No. 202301AT070394, 202301BE070001-037).

References

1. Gosliga J, Hester D, Worden K, et al. On population-based structural health monitoring for bridges. Mechanical Systems and Signal Processing, 173, 108919(2022).
2. Zhou Y, Sun L. A comprehensive study of the thermal response of a long-span cable-stayed bridge: from monitoring phenomena to underlying mechanisms. Mechanical Systems and Signal Processing, 124, 330-348(2019).
3. Sun Z, Sun H. Jiangyin Bridge: An example of integrating structural health monitoring with bridge maintenance. Structural Engineering International, 28(3), 353-356(2018).
4. Zhao H W, Ding Y L, Li A Q, et al. Evaluation and early warning of vortex-induced vibration of existed long-span suspension bridge using multisource monitoring data. Journal of Performance of Constructed Facilities, 35(3), 04021007(2021).
5. Tang Z. Diagnosis and Reconstruction of Anomalous Data in Structural Health Monitoring Based on Deep Learning. Harbin: Harbin Institute of Technology, 2021.
6. Yuan X, Chen H, Liu B. Point cloud clustering and outlier detection based on spatial neighbor connected region labeling. Measurement and Control, 54(5-6), 835-844(2021).
7. Yuen K V, Mu H Q. A novel probabilistic method for robust parametric identification and outlier detection. Probabilistic Engineering Mechanics, 30, 48-59(2012).
8. Tabatabaei M, Kimiaefar R, Hajian A, et al. Robust outlier detection in geo-spatial data based on LOLIMOT and KNN search. Earth Science Informatics, 14(2), 1065-1072(2021).
9. Campello R J G B, Moulavi D, Zimek A, et al. Hierarchical density estimates for data clustering, visualization, and outlier detection. ACM Transactions on Knowledge Discovery from Data (TKDD), 10(1), 1-51(2015).
10. Fan S, Zhang J, Zhang Q. Data Anomaly Identification in Structural Health Monitoring Systems. Computer-Aided Engineering, 25(05), 60-65(2016).
11. Bao Y, Tang Z, Li H, et al. Computer vision and deep learning–based data anomaly detection method for structural health monitoring. Structural Health Monitoring, 18(2), 401-421(2019).
12. Tang Z, Chen Z, Bao Y, et al. Convolutional neural network - based data anomaly detection method using multiple information for structural health monitoring. Structural Control and Health Monitoring, 26(1), e2296(2019).
13. Liu G, Niu Y, Zhao W, et al. Data anomaly detection for structural health monitoring using a combination network of GANomaly and CNN. Smart Struct Syst, 29(1), 53-62(2022).
14. Wadekar S N, Chaurasia A. Mobilevitv3: Mobile-friendly vision transformer with simple and effective fusion of local, global and input features. arXiv preprint arXiv:2209.15159 (2022).