



OPEN Improved road traffic sign recognition from feature reconstruction

Gang Huang^{1,3}, Huiling Cao¹, Jiayue Sun¹, Zhenling Chen¹ & Zhe Zhou^{2,3}✉

Road traffic sign recognition is an essential function of modern autonomous driving systems. Accurate recognition plays an important role in ensuring vehicle safety and enabling subsequent operations. This paper introduces a Siamese architecture that is combined with feature reconstruction to overcome current method limitations. In detail, the proposed method enhances recognition accuracy by reconstructing convolutional features within a Siamese Neural Network (SNN) framework, integrating an improved Mamba network with mainstream CNN architectures, including VGG-16, AlexNet, ResNet, or MobileNetV2. Through comprehensive architectural comparisons, the optimal network configuration is determined for different application scenarios. Utilizing this approach, the accuracy of traffic sign recognition is substantially improved, addressing the shortcomings of existing technologies. Extensive experimental validation was conducted on multiple datasets. Taking VGG-16 network as an example, the experimental results demonstrated the method's effectiveness, achieving accuracies of 99.83% on the GTSRB dataset, 99.13% on the TSRD dataset, and 99.07% on the TT100K dataset.

Keywords Traffic sign recognition, Mamba, Siamese neural networks, Feature reconstruction

Traffic sign recognition (TSR) technology¹ occupies a core role in intelligent transportation systems. It not only helps with real-time traffic monitoring but also is an indispensable part of autonomous driving. In Advanced Driver Assistance Systems, TSR technology can compensate for the driver's visual oversight. In autonomous driving (e.g., SAE Levels 3 to 5), the detection and recognition of traffic signs, as the basis of vehicle perception, is the premise to ensure the safety of autonomous driving². However, due to various factors, TSR still faces numerous challenges. Improving the precision and reliability of TSR remains a pressing challenge requiring immediate attention.

Traditional methods for traffic sign recognition (TSR) primarily rely on geometric or handcrafted features, such as shape, color, or spatial relationships. Among them, the Histogram of Oriented Gradients (HOG) combined with a Support Vector Machine (SVM) classifier has been one of the most widely used pipelines. However, handcrafted features like HOG remain highly sensitive to variations in illumination, shadows, and adverse weather (e.g., rain, fog, or snow), even when preprocessing techniques such as Gamma correction are applied. This sensitivity limits their robustness in complex real-world scenarios, where environmental conditions vary significantly. Consequently, traditional feature-based approaches often struggle to maintain stable recognition performance without the aid of additional sensors or adaptive mechanisms. These challenges have motivated the development of deep learning-based methods^{3–6} (CNNs) that can automatically extract and integrate more discriminative features.

Aiming at improving accuracy and designing a flexible fusion network, this study proposes a Siamese neural network (SNN) framework that integrates an improved Mamba network with mainstream CNN architectures (including VGG-16, ResNet, AlexNet, or MobileNetV2), named Siamese-Mamba&CNN (SM&CNN). Then, a Heterogeneous Network Feature Reconstruction (HN-FR) module is introduced to fuse feature information across different scales. Based on the fusion of neural network modules, this approach integrates an improved Mamba network, which integrates the advantages of traditional Convolutional Neural Networks (CNNs) and State Space Models (SSMs). Specifically, the proposed improved Mamba replaces the convolutional structures by introducing the Depth Separable convolution Block (DS Block) based on the Med-Mamba network⁷, reducing computational complexity. The framework's compatibility with various CNNs enables adaptive feature

¹School of Automobile and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430081, People's Republic of China. ²Hubei Key Laboratory of Cooperative Vehicle Infrastructure and Traffic Control, Xiangyang 441053, China. ³Hubei Key Laboratory of Power System Design and Test for Electrical Vehicle, Hubei University of Arts and Science, Xiangyang 441053, China. ✉email: zzhou@hbuas.edu.cn

extraction, such as using ResNet's residual learning for complex patterns, MobileNetV2's lightweight design for mobile deployment, or VGG-16's hierarchical features for structural recognition. Through comprehensive architectural comparisons, we establish optimal network configurations for different operational scenarios. The HN-FR module dynamically reconstructs features from these heterogeneous networks, achieving better representation capability. The novelty of this study is demonstrated through the following contributions: (1) The designed fusion network effectively integrates the lightweight architecture of the improved Mamba network and the feature extraction capability of the flexibly replaceable CNN. This flexibility allows the model to be applied to different network configurations, providing adaptability across various deployment needs while preserving reliable feature representation. In the improved Mamba network, DS Block is incorporated into the traditional Med-Mamba structure, achieving performance improvements with a reduction of 6.94 M parameters, thereby enhancing computational and memory efficiency. (2) In the fusion process of heterogeneous networks, we propose an HN-FR method to integrate heterogeneous network features and reconstruct convolution features through feature mapping, attention weighting, and full connection, achieving fast and excellent performance. Experiments show that the backbone network (such as MobileNetV2, VGG-16, ResNet50, AlexNet, etc.) improves its accuracy by 0.95% to 36.2% after integrating the proposed HN-FR method, with an average improvement of 10.8%.

Related work

In recent years, the field of TSR has attracted extensive research attention and spawned many innovative solutions. In this section, we analyze the research works in TSR over the years. Traditional methods primarily rely on geometric features⁸ and handcrafted features^{9,10} including the form and color attributes of traffic signs. In contrast, the use of neural networks to extract convolutional features, especially the integration of diverse neural network components to identify and classify traffic signs, usually promotes performance enhancement.

Traditional methods

The traditional TSR methods mainly includes the extraction of geometric features or handcrafted features. The TSR process of extracting geometric features focuses on identifying the specific shapes and geometric patterns of traffic signs, which are very important for the sign recognition process. The common methods include the Hough transform, the Canny operator, and other techniques. For example, Satti et al.¹¹ proposed an integrated method to detect traffic signs and potholes on Indian roads. This method uses the joint application of Features from the Accelerated Segmentation Test (FAST) and the Random Sample Consensus (RANSAC) algorithm to extract and match features in the process of detecting and identifying traffic signs. Pothole detection employed an enhanced Canny Edge detector alongside a bio-inspired contour detection technique. The experimental data confirmed that the comprehensive model developed in this study surpassed the existing models in key indicators such as accuracy, specificity, and sensitivity. In the TSR process of extracting handcrafted features, traffic sign recognition mainly relies on artificially designed features. Typical handcrafted features consist of Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT)¹², and Local Binary Pattern (LBP), among others. Kassani et al.⁹ proposed an innovative sparse model for traffic sign classification, utilizing a Soft Histogram of Oriented Gradients (SHOG) combined with sparse classifiers. They proposed a variant of HOG, named SHOG, which leverages the symmetry of traffic sign images to determine the optimal cell locations for histogram computation, enhancing discriminative power over traditional HOG. Furthermore, two analytical classifiers based on sparse polynomial models were introduced to implicitly select features and mitigate overfitting. The method was tested on the GTSRB dataset and 16 additional datasets from the UCI repository, showing competitive results when compared to cutting-edge approaches.

However, in practical applications, geometric or handcrafted feature-based methods suffer from illumination or weather changes, sign obstruction, or breakage. Consequently, traditional approaches^{9,12} generally exhibit limited consistency in recognition accuracy and robustness under diverse real-world conditions, compared with the methods based on deep learning widely used in recent years.

Deep Learning-Based methods

Neural networks, particularly deep learning models, surpass traditional approaches by extracting extensive features from images and autonomously learning complex feature representations from large-scale datasets. Deep learning methods excel in both accuracy and efficiency, and advancements in neural networks—such as AlexNet¹³, LeNet¹⁴, VGG-16^{15,16}, DenseNet¹⁷, and ResNet^{18,19}—have significantly propelled the development of object detection and recognition technologies.

Classical deep learning architectures

In the field of TSR, the development of convolutional neural networks (CNNs) has markedly advanced the accuracy of image classification. Existing approaches have generally adopted two strategies. On one hand, techniques such as optimized deep network architectures aim to maximize recognition performance under sufficient computational resources. On the other hand, lightweight network designs, including MobileNet and other parameter-efficient architectures, are employed to enable inference in resource-constrained environments. In the first strategy, Castruita et al.²⁰ combine R-CNN or YOLO v3 for detection with a modified ResNet-50 for classification, achieving robust performance on a newly created Mexican traffic sign dataset with mAP up to 95.33% and overall classification accuracy of 99.00%, even under occlusion and non-sign scenarios. Saha et al.²¹ introduce a unified residual convolutional network with hierarchical dilated skip connections and a low-memory dilated residual representation, enabling high accuracy across multiple benchmarks. Zatroch et al.²² proposed a Siamese network with one-shot learning and synthetic data augmentation, allowing effective recognition from very few samples per class. Collectively, these innovations—integrated detection-classification pipelines,

advanced residual architectures, and few-shot learning with data augmentation—demonstrate effective strategies for achieving high-precision TSR under diverse conditions. These studies highlight the strong recognition capability of deep CNNs, yet their large model sizes and high computational cost hinder real-time deployment on resource-limited platforms. To enhance inference efficiency, some studies have focused on network optimization and lightweight designs. Zhang et al.²³ proposed the Transformer-TSD-DETR, introducing ConvNormLayer and TSDBlock for multi-scale feature extraction, which reduces parameters, improves inference speed. Lin et al.²⁴ developed YOLO-LLTS, incorporating HRFM-SOD and multi-branch feature interactive attention to enhance small-target detection in low-light conditions, reaching 77.2% accuracy on the TT100K-night dataset while maintaining YOLO's efficiency. Despite these gains, lightweight strategies can slightly reduce accuracy under complex or low-contrast conditions.

Apart from the aforementioned strategies, recent years have also witnessed significant representative works, such as attention mechanisms and networks based on state-space models (e.g., Mamba), which further advance traffic sign recognition. Shen et al.²⁵ proposed a multi-scale grouping attention pyramid network. This method effectively aggregates feature information of different scales by constructing a multi-scale attention module, suppresses background interference, and enhances the perception of small-sized traffic signs. In order to realize the detection of small traffic signs under complex weather conditions, Qu et al.²⁶ introduced the Coordinate Attention (CA) mechanism into the backbone network, embedded the location information into the channel attention, and improved the network's perception of spatial details. Li et al.²⁷ proposed a sample defense method for traffic sign confrontation based on an attention mechanism. This method simulates the human visual system, focuses on the key areas of the target in the image, and ignores the background disturbance, thereby improving the robustness of the model against attacks. The Mamba architecture, as a state space model (SSM), has provided a new direction for visual modeling. Therefore, more researchers^{28–30} have improved the Mamba and achieved a wide range of applications. For example, Liu et al.³¹ proposed a lightweight visual network LightViM based on Mamba for resource-constrained scenarios. By constructing the LGF-Mamba module, the bidirectional state space model is used to extract the global context features (such as shape contour) of the image, and the local convolution module is used to capture high-frequency details (such as edge texture) to achieve multi-frequency information complementarity. Validated on data sets such as medical endoscopes and bird recognition, its accuracy exceeds that of lightweight models such as MobileViT by more than 3.7%, providing a new sequence modeling paradigm for edge-end visual tasks. Despite the significant advantages of deep learning-based TSR methods in accuracy and robustness.

Multi-Network fusion methods

The increasing availability of high-performance computing resources has enabled the effective fusion of neural networks, which demonstrates significant advantages in traffic sign recognition and classification. Yu et al.¹⁶ proposed a multi-image fusion model method that combines YOLO-V3 and VGG-16 networks. By utilizing correlations among multiple images, this model efficiently detects and identifies traffic signs in driving videos with high accuracy. Testing results on public datasets indicate that the recognition accuracy of this model exceeds 90%, and its performance in detecting and recognizing all types of traffic signs under various conditions outperforms baseline methods. Li et al.³² introduced an effective two-stage fusion neural network framework. Unlike traditional cascaded methods, this framework predicts categories directly in the first-stage detection and enhances overall robustness by fusing category predictions from both stages. In feature fusion, He et al.³³ developed the Attentional Aggregative Interaction Network (AAIN), a deep learning model that enhances feature interactions in recommender systems through a cyclic explicit module and attention mechanism, capturing high-order features effectively. The model was integrated with DNN to form DAAIN, outperforming existing models in feature interaction and accuracy. Jin et al.³⁴ proposed MF-SSD based on the SSD framework. Through a feature fusion strategy, the effective channel is strengthened, which improves the detection effect of small-sized traffic signs. Yu et al.³⁵ proposed a real-time target detection network based on multi-channel feature fusion. Aiming at the shortcomings of YOLOv8 in the detection of small targets such as traffic signs, a multi-path attention mechanism and a two-way fusion structure are introduced. Through multi-dimensional feature tensor fusion, this method adaptively enhances the spatial and channel information in shallow features, effectively suppresses redundant background noise, and improves the discrimination of small targets. These multi-network fusion strategies commonly exploit hierarchical and multi-scale feature representations to integrate complementary information from different layers or channels, thereby enhancing recognition accuracy.

Compared with traditional deep learning methods, multi-network fusion approaches can learn and integrate features from different scales and hierarchical levels, thereby reducing the risk of model overfitting. Future work may explore the fusion of more diverse feature types and the design of more flexible fusion strategies to further enhance recognition performance.

Proposed method

Figure 1 presents the comprehensive framework of the proposed method. Based on our previous work in³⁶, and inspired by the work of Med-Mamba⁷ and AAIN³³, we propose a method for TSR by using an SNN framework to fuse an improved Mamba network with mainstream CNN architectures, including VGG-16, ResNet-50, MobileNetV2, etc. The purpose is to improve recognition performance by integrating the advantages of heterogeneous networks. Specifically, a Heterogeneous Network Feature Reconstruction (HN-FR) module is proposed to reconstruct the features. Figure 1 illustrates that the proposed method is composed of three main components. Firstly, within the framework of the SNN, we define positive and negative sample pairs. Secondly, we introduce the customized improved Mamba network and selectable CNN backbones, proposing a Heterogeneous Network Feature Reconstruction (HN-FR) module for reconstructing convolutional features.

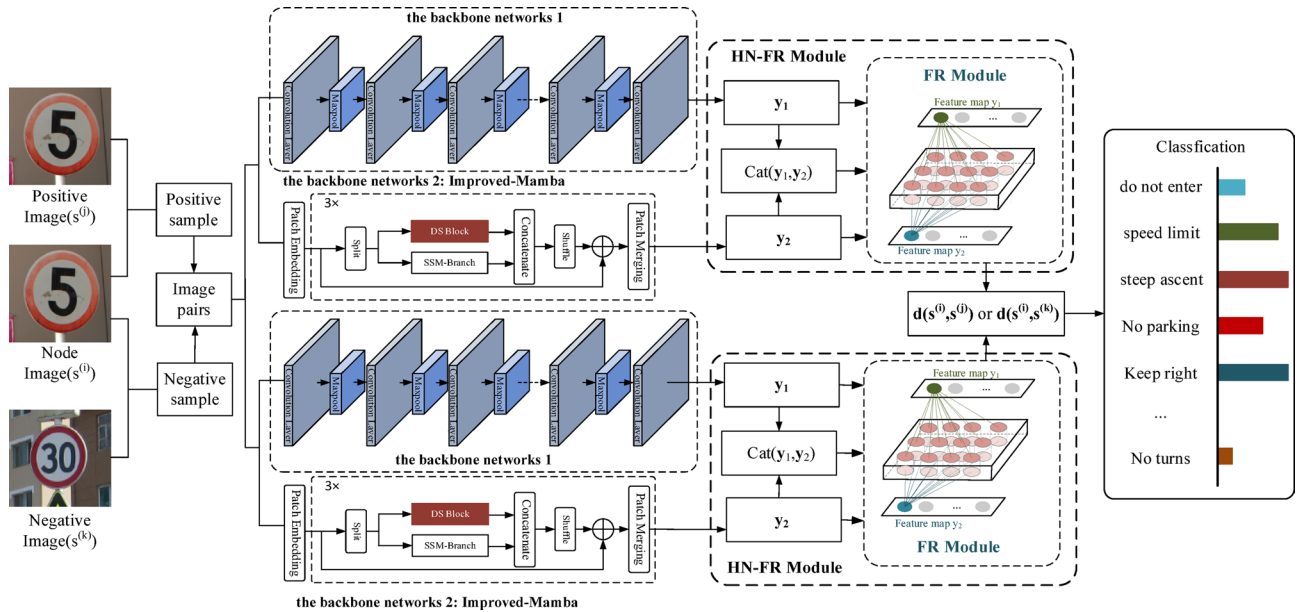


Fig. 1. Sketch of the SM&CNN. The backbone network 1 can be any existing model, such as AlexNet, ResNet-50, VGG-16, or MobileNetV2; and the improved-Mamba is another backbone network.

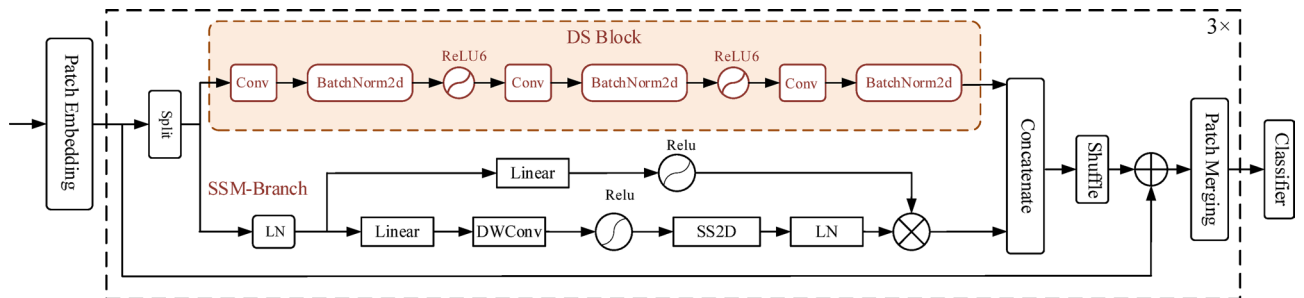


Fig. 2. Sketch of the improved mamba network.

Finally, we train the network by setting training objectives and loss functions. The following chapters will explain our proposed method in detail.

State space features from improved Mamba network

Figure 2 describes the basic structure of the improved Mamba network. The overall framework of this model is primarily based on the Med-Mamba network, which is proposed by Yue et al.⁷ It comprises a patch embedding layer, multiple stacked SS-Conv-SSM modules, patch merging layers for down-sampling, and a classifier. The architecture integrates State Space Models (SSMs), represented by Mamba, and a series of convolutional branch operations. Based on the Med Mamba network, we have made further improvements. The SSM-Branch branch of the Med-Mamba network is retained. Affected by the depth separable convolution design, we introduce a Depth Separable Block (DS Block) to substitute the original Conv-Branch, aiming to enhance computational efficiency. The detailed operation of the DS Block is presented in Fig. 3.

In contrast to traditional Conv-Branch operations, the DS Block utilizes a depth separable convolution design, where the smaller number of network parameters benefits both model training and inference speed. Under the same hardware environment, DS Block improves the execution speed of the model while maintaining strong expressiveness and has the potential to be deployed on portable devices. Furthermore, the use of regularization operation and activation function enhances training stability and non-linear expressive capabilities, ensuring the model achieves both high accuracy and enhanced processing speed. Therefore, the DS Block is a lightweight and efficient module that, through its rational structural design, effectively reduces computational costs and improves the model's inference speed, making it an efficient module that balances speed and performance.

Given a module input $R^{H \times W \times C}$ and a module output $y \in R^{H \times W \times C}$, and then there is

$$x \in R^{H \times W \times C}, x_{i=1,2} \in R^{H \times W \times \frac{C}{2}} \tag{1}$$

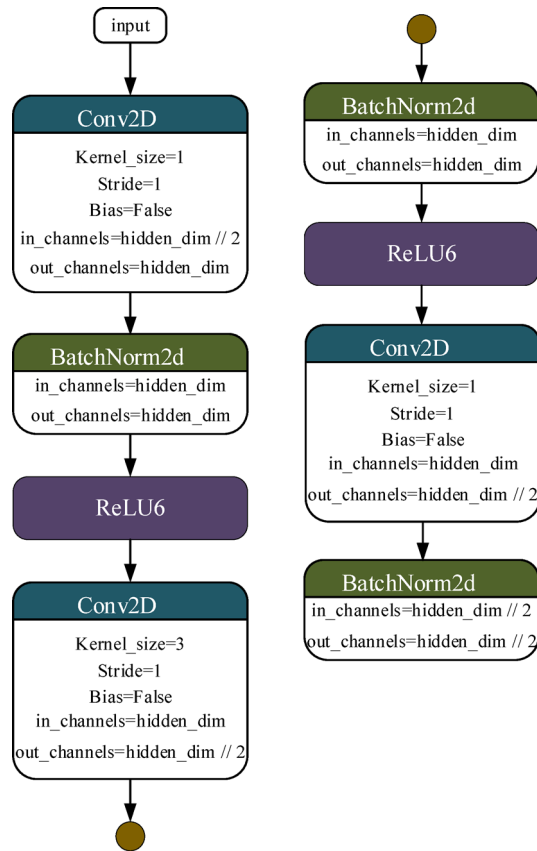


Fig. 3. Flow chart of DS block.

where x_1 and x_2 are the inputs of the DS block and SSM -Branch, respectively. Based on the above, the construction process of DS Block can be defined as follows:

$$\begin{cases} x' = \vartheta(\varphi(x_1 \times \gamma_{1 \times 1}^1)) \\ x'' = \vartheta(\varphi(x' \times \gamma_{3 \times 3}^{depthwise})) \\ \tilde{x}_1 = \varphi(x'' \times \gamma_{1 \times 1}^2) \end{cases} \quad (2)$$

where $\gamma_{1 \times 1}$ refers to the depth separable convolution with a 1×1 kernel; $\gamma_{3 \times 3}$ refers to the depth separable convolution with a 3×3 kernel; φ is a regularization operation; ϑ is the activation function.

The f^{-1} and g are used to represent channel-concatenation and channel-shuffle respectively. The channel cascade is used to restore the size of the channel dimension, and the channel shuffling is used to shuffle the feature map on the channel dimension to avoid the loss of information between channels caused by the grouping convolution operation. In summary, the feature representations outputted by the improved Mamba network are as follows:

$$y_2 = x \oplus g(f^{-1}(\tilde{x}_1, \tilde{x}_2)) \quad (3)$$

where \tilde{x}_2 are the output features of the SSM-Branch in the Med-Mamba network⁷.

Configurable CNN backbone

The SSM in Mamba has the advantages of dynamic long-range context capture and direction-sensitive feature selectivity in feature modeling. Traditional CNN also has advantages in strong local feature representation and spatial translation invariance. As a result, we try to fuse the advantages of those two different networks to improve the accuracy of TSR. For demonstrate the flexible of our network, we choose four different CNN networks as replaceable CNN components.

Case1
$$y_1 = MP\left(\delta\left(\text{Conv}_{3 \times 3}^{(n)}\left(\dots \delta\left(\text{Conv}_{3 \times 3}^{(1)}(x_{in})\right)\dots\right)\right)\right) \in R^{\frac{H}{2^K} \times \frac{H}{2^K} \times C^k} \quad (4)$$

Case2
$$y_2 = \mathcal{F}(x_{in}, \{W_i\}_{i=1}^3) + x_{in} \in R^{H \times W \times C} \quad (5)$$

$$\text{Case3} \quad \mathbf{y}_3 = \text{Conv}_{1 \times 1}(\sigma(\text{DWConv}_{3 \times 3}(\sigma(\text{Conv}_{1 \times 1}(x_{in})))))) \in R^{H \times W \times 6C} \quad (6)$$

$$\text{Case4} \quad \mathbf{y}_4 = \text{MP}(\sigma(\text{Conv}_{11 \times 11}(x_{in}))) \in R^{(H-10) \times (W-10) \times 96} \quad (7)$$

where \mathbf{y}_1 in Case 1 is the output characteristic of VGG-16³⁷; \mathbf{y}_2 in Case 2 is the output feature of ResNet-50³⁸; \mathbf{y}_3 in Case 3 is the output feature of MobileNetV2³⁹; \mathbf{y}_4 in Case 4 is the output feature of AlexNet⁴⁰.

Proposed feature fusing method

A Heterogeneous Network Feature Reconstruction module (HN-FR) is designed to integrate features generated by the improved Mamba and the mainstream CNN architectures. Within complex models, cross-feature interaction fusion³³ facilitates comprehensive modeling of the interplay between disparate feature vectors, capturing the underlying correlations and intricate relationships among these features. This approach is particularly applicable in scenarios that require the capture of complex feature interactions, multimodal data integration, or the processing of high-dimensional sparse data, and it offers extensive application potential in fields like recommendation systems, natural language processing, and computer vision. The HN-FR is proposed for the purpose of merging and dimension-reducing feature vectors from different networks to enhance performance in classification tasks. Specifically, the HN-FR module takes intermediate layer feature representations from two distinct networks (e.g., Mamba and VGG-16) and integrates these into a low-dimensional, semantically enriched feature representation through a learned attention mechanism. This attention mechanism dynamically computes the importance of each input feature, assigning varying weights to different features by inspired in⁴¹. It ensures the model can autonomously identify and highlight the most distinctive features while minimizing redundant and less relevant information. Compared with existing attention-based and feature fusion methods discussed in the related work, the attention mechanism in the proposed HN-FR module is specifically designed to perform arbitration between two heterogeneous networks, emphasizing the selection and weighting of the more reliable feature sources during fusion. Figure 4 provides the structure diagram of the HN-FR module. In the HN-FR module, the middle layer features of the extracted CNN are used to generate the attention weight θ through the learnable attention parameters and the Sigmoid activation function, and the feature \mathbf{y}_1 is weighted element by element to obtain the weighted feature Ψ . Then, Ψ is concatenated with the original feature \mathbf{y}_2 and sent to the first fully connected layer, producing the intermediate feature \mathbf{z}_1 . Subsequently, \mathbf{z}_1 is combined with \mathbf{Y} as input to the second fully connected layer, resulting in the final fused feature \mathbf{z}_2 . The module realizes the fusion and reconstruction of cross-source features through attention mechanism and a nonlinear transformation and provides high-quality feature representation for subsequent tasks. The specifics are outlined below.

We concatenate the intermediate layer features of the Mamba network with those from the CNN backbone into a unified representation vector:

$$\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2] \in R^{N \times 2D} \quad (8)$$

and feed \mathbf{Y} back into the network.

An attention mechanism is applied to adjust the CNN backbone features, with the attention weights $\theta \in R^{N \times 1}$ calculated as follows:

$$\theta = \delta(\mathbf{y}_1 \mathbf{a}) \quad (9)$$

where $\mathbf{a} \in R^{2D \times 1}$ is the learnable attention parameter, and δ denotes the Sigmoid activation function. These attention weights are then used to scale the features elementwise:

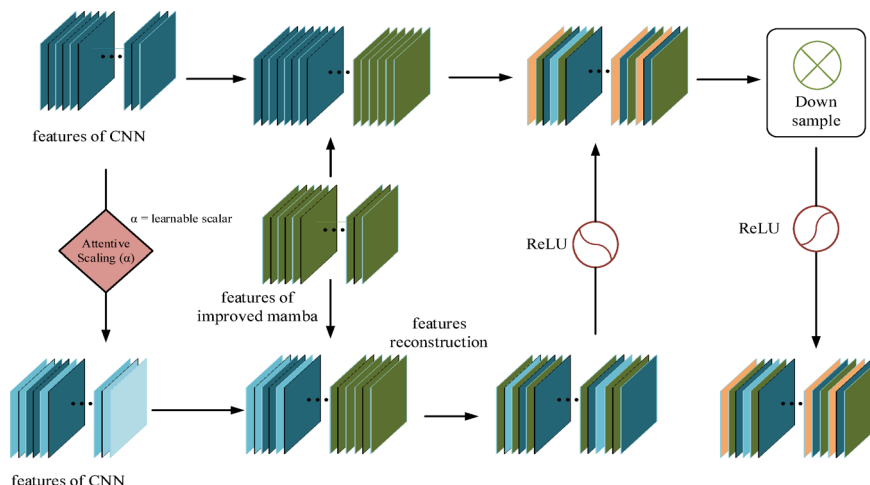


Fig. 4. Sketch of the HN-FR Module.

$$\Psi = \mathbf{y}_1 \odot \theta \quad (10)$$

We take the results of the splicing of Ψ and \mathbf{y}_2 as the input of the first full connection, and take the output of the first full connection and the splicing result of \mathbf{Y} as the input of the second full connection operation, which is expressed as follows :

$$\mathbf{X}_1 = [\Psi, \mathbf{y}_2] \quad (11)$$

$$\mathbf{X}_2 = [\mathbf{z}_1, \mathbf{Y}] \quad (12)$$

where \mathbf{X}_1 is the input of the first full connection; \mathbf{X}_2 is the input of the second full connection.

Then, the weighted features undergo two fully connected operations with weight matrices and bias vectors. Next, ReLU is introduced as the activation function, and the detailed formula is given below:

$$\mathbf{z}_1 = \ell(\mathbf{X}_1 \mathbf{W}_1 + \mathbf{b}_1) \quad (13)$$

$$\mathbf{z}_2 = \ell(\mathbf{X}_2 \mathbf{W}_2 + \mathbf{b}_2) \quad (14)$$

where $\mathbf{W}_1 \in R^{D \times 2D}$ and $\mathbf{W}_2 \in R^{D \times D}$ represent the weight matrices of two connected layers, respectively. $\mathbf{b}_1 \in R^D$ and $\mathbf{b}_2 \in R^D$ denote the corresponding bias vectors, and ℓ refers to ReLU, a type of nonlinear activation function.

The reconstruction process allows the model to automatically focus on the most critical part of the input features through the attention mechanism, thereby enhancing the integration of complementary information from \mathbf{y}_1 and \mathbf{y}_2 . Additionally, feature reconstruction and down-sampling dimensionality reduction are achieved through two fully connected operations.

Loss function

The SNN primarily consists of a backbone feature extraction network and a comparison network. The distance between a positive sample pair, such as a node image, and its matching (positive) image, is denoted by $d(s^{(i)}, s^{(j)})$. Similarly, $d(s^{(i)}, s^{(k)})$ represents the distance for a negative sample pair, which is the distance between a node image and a non-matching (negative) image.

$$d(s^{(i)}, s^{(j)}) = \|f(s^{(i)}) - f(s^{(j)})\|^2 \leq d(s^{(i)}, s^{(k)}) = \|f(s^{(i)}) - f(s^{(k)})\|^2 \quad (15)$$

In the binary classification problem of SNN, BCEWithLogitsLoss is often used serving as the loss function to evaluate the similarity between two input samples. It combines the sigmoid activation with the cross-entropy loss, avoiding the direct application of the sigmoid function on the output layer and maintaining numerical stability. In SNN, the output d typically represents a distance metric, and BCEWithLogitsLoss can effectively handle this distance-based metric by accurately computing the difference between the similarity score and the label, aiding the network in distinguishing between similar and dissimilar samples. The formula for BCEWithLogitsLoss is as follows:

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^N [y_i \times \log(\sigma(\hat{y}_i)) + (1 - y_i) \times \log(1 - \sigma(\hat{y}_i))] \quad (16)$$

where N denotes the number of samples; y_i represents the model output (i.e., logits) for the i -th sample, which has not undergone sigmoid activation; $\sigma(\hat{y}_i) = \frac{1}{1+e^{-\hat{y}_i}}$ is the probability value obtained by applying the sigmoid activation function to \hat{y}_i , ranging between $[0,1]$; and y_i is the true label for the i -th sample, taking a value of either 0 or 1.

Experiment

The experiment was performed on a system equipped with Intel's 12th-generation i5 CPU and NVIDIA RTX 2080 Ti GPU, providing efficient computation for model training and inference. The environment was configured using Anaconda, with the model developed and trained in Python leveraging the PyTorch framework. Conda was used to manage package dependencies, ensuring the environment's reproducibility and stability. This hardware and software configuration provided robust computational support for model training and validation, enabling the experiment to be completed within a reasonable timeframe.

Evaluation indicators

In TSR, mAP (mean Average Precision) and accuracy can be used as evaluation indicators to evaluate model performance. However, the accuracy assesses the model's performance across the entire set of test samples, which makes it more suitable for those categories with a large number of images. In contrast, mAP incorporates both accuracy and recall to assess the model's effectiveness. It evaluates the average accuracy under the precision-recall curve under different thresholds and indicates the performance of the model under different categories, especially in scenarios with numerous categories or unbalanced data. Therefore, in the three data sets involved in this paper, according to their respective characteristics, GTSRB and TSRD use accuracy as the evaluation index, and mAP is beneficial for performance verification of TT100K, as presented by⁴².

Accuracy is a basic and intuitive evaluation metric that can provide effective performance evaluation in the case of balanced categories or simple tasks. The computation is as follows:

$$Accu = \frac{P_{true} + N_{ture}}{P_{ture} + N_{ture} + P_{false} + N_{false}} \quad (17)$$

where P_{true} denotes the count of samples accurately classified as positive, N_{ture} refers to the count of samples accurately classified as negative, P_{false} indicates the count of samples mistakenly classified as positive, and N_{false} signifies the count of samples mistakenly classified as negative.

Because Precision and Recall have an inherently inverse relationship, it is challenging to comprehensively assess the effectiveness of the test results using these two metrics alone, so we use the average accuracy for quantitative analysis. The performance evaluation diagram is drawn and the P-R curve is obtained. Then, the precision-recall (P-R) curve is integrated to calculate the average precision (AP). The formula for calculation is as follows:

$$AP = \int_0^1 P(r)dr \quad (18)$$

To obtain the overall evaluation for multi-class traffic sign recognition, mAP is determined by averaging the AP values across all categories. The formula for calculation is:

$$mAP = \frac{1}{m} \sum_{i=1}^N AP_i \quad (19)$$

Datasets

The experiment is based on three well-known traffic sign datasets: the German Traffic Sign Recognition Benchmark (GTSRB), the Chinese Traffic Sign Recognition Dataset (TSRD), and Tsinghua-Tencent 100 K (TT100K). Table 1 details the total number of images, including the number of training and testing images, and the categories of traffic signs included in each dataset.

The GTSRB dataset comprises approximately 50,000 traffic sign images across 43 categories, captured under diverse lighting and weather conditions, which helps to improve the adaptability of the model to the actual environment. The GTSRB dataset presents various challenges arising from image conditions, as illustrated in Fig. 5. Certain images suffer from issues such as low resolution, inadequate contrast, insufficient brightness, blurriness, poor lighting, and motion-induced tilt. The TSRD dataset focuses on Chinese traffic signs, including images of six main categories of traffic signs, totaling about 10,000 images, providing training data for the model to adapt to local Chinese traffic signs. The distortion in the TSRD dataset is similar to that of the GTSRB. As depicted in Fig. 5, the TSRD dataset is mainly influenced by issues such as image aging, inadequate brightness, and blurriness. In contrast, the TT100K dataset provides a large collection of more than 100,000 labeled images, specifically for traffic sign detection and recognition tasks, and covers a variety of traffic signs widely used in China. The dataset is widely used in autonomous driving and intelligent transportation system research. TT100K supports images collected under various conditions (different illumination, weather, and shooting angle), so there is also deformation, blur, occlusion, and more. It supports traffic sign classification and target detection tasks, which helps promote automatic TSR system.

In the experiment we designed, the dataset was divided into training and testing sets according to the author's original classification. The use of three datasets ensures the model's classification performance across a variety of traffic sign categories and complex real-world scenarios.

Experimental training

The experimental training is based on the SM&CNN framework, designed to implement an efficient training process that maximizes the performance of deep learning models. The experiments are conducted in a GPU environment supporting CUDA, with automatic mixed precision training (AMP) enabled by default to reduce memory usage, and distributed data parallelism (DP) is utilized to maximize the utilization of hardware resources. The experimental model is based on an enhanced Siamese network architecture. In order to maintain the stability of the training process, the initial learning rate is set to 0.01, the weight attenuation coefficient is set to 5×10 , the optimization algorithm selects the Stochastic Gradient Descent (SGD) and adds a momentum factor of 0.9.

Training data is enhanced to improve generalization performance through data augmentation strategies, including random rotation, cropping, flipping, and other operations. Considering the properties of the input

Dataset	Number of classes	Testing images	Training images	Total images
GTSRB	43	12,630	39,209	51,839
TSRD	58	1994	4170	6164
TT100K	173	8415	16,922	25,337

Table 1. Datasets used in experiments.

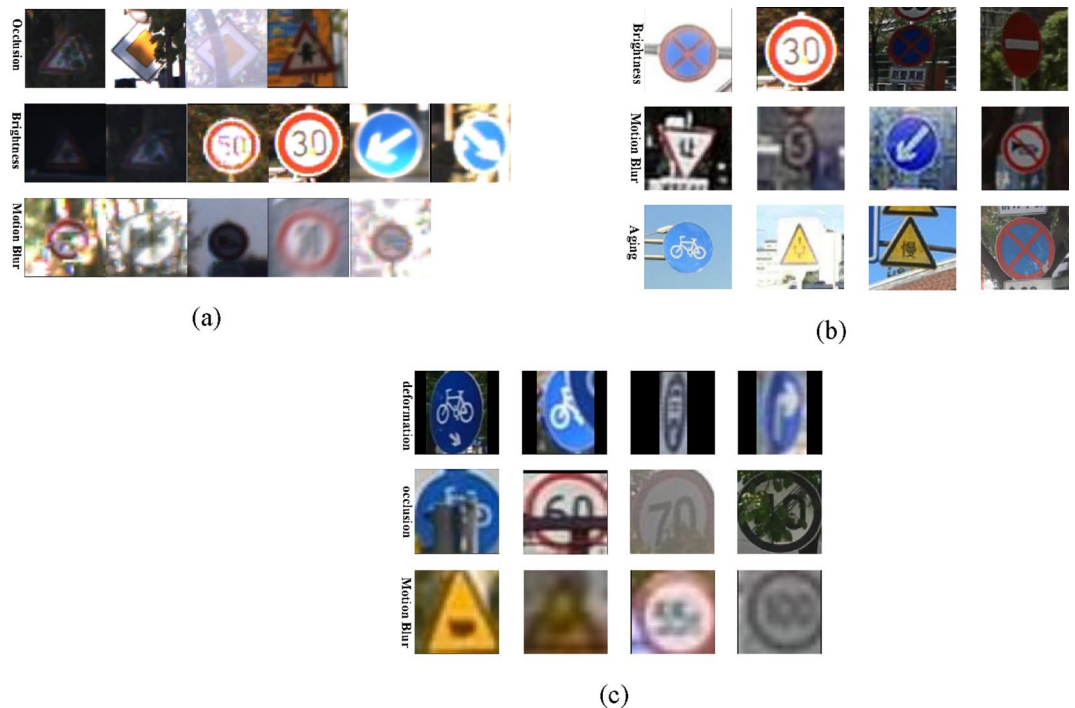


Fig. 5. GTSRB, TSRD and TT100K image variations. (a) GTSRB images variations, (b) TSRD images variations, (c) TT100K images variations.

Labels	Method	Accuracy (%)
1	CNN ³	99.66
2	RIECNN ⁴	99.75
3	LeNet-5 ¹⁴	97.53
4	VGG-16 ¹⁵	99.21
5	DeepThin ⁴³	99.72
6	ours	99.83

Table 2. ACCURACY results on the GTSRB dataset.

images, the data loader groups the images into batches, enhancing the efficiency of data flow transfer and accelerating the training process.

Throughout the training, we employ the cosine annealing strategy to dynamically adjust the learning rate:

$$lr_t = lr_{\min} + \frac{1}{2}(lr_{\max} - lr_{\min})(1 + \cos(\frac{t}{T}\pi)) \quad (20)$$

where lr_{\max} is the initial learning rate, lr_{\min} is the minimum learning rate, and T is the number of training iterations. This strategy smooths the variation of the learning rate, enhancing the stability of training. The training procedure consists of two stages: the freezing stage and the unfreezing stage. During the freezing stage, the majority of the network's parameters remain fixed, and only specific layers are trained to ensure the robustness of basic feature extraction. In the unfreezing phase, network layers are gradually unfrozen, optimizing global parameters and enhancing the model's adaptability in diverse scenarios.

Experimental analysis

We designed the following experiments to test the performance of our algorithm on different datasets, including comparative experiments on GTSRB, TSRD, and TT100K. Taking the VGG-16 network as an example (Siamese-Mamba&VGG16), we select the current mainstream methods, including CNN, LeNet, MobileNet, VGG-16, ANN, etc., for comparative experiments. Most of these methods are based on deep learning network architecture. Meanwhile, the methods we choose also include the improvement of some mainstream network architecture, or the combination of multiple network architectures, such as DeepThin, RIECNN, etc.

Table 2 presents the comparison results on the GTSRB dataset. The method proposed in this study exceeds all comparison models and achieves 99.83% accuracy. This exceeds other high-performing models, such as RIECNN⁴ (99.75%) and DeepThin⁴³ (99.72%). While traditional architectures like LeNet-5¹⁴ exhibit a

Labels	Method	Accuracy (%)
1	CNN ⁴⁴	99.16
2	CNN ⁴⁵	95.00
3	MLP, DWT, DCT [46]	94.90
4	PCA, HOG, LBP, SVM [47]	84.44
5	Prototypes [48]	90.13
6	ours	99.13

Table 3. Accuracy results on the TSRD dataset.

Labels	Method	mAP (%)
1	CCSPNet [50]	90.9
2	Faster R-CNN [51]	89.5
3	CAMF [42]	92.7
4	FCOS [52]	88.1
5	FII-CenterNet [53]	89.4
6	ours	93.84

Table 4. mAP results on the TT100K dataset.

lower accuracy of 97.53%, more advanced models such as VGG-16¹⁵ (99.21%) and CNN³ (99.66%) are also outperformed by the proposed approach. It also more validates the advantages and features that our proposed method has on the GTSRB dataset.

Table 3 presents a comparison with other approaches on the TSRD dataset. Some of the data in this table is sourced from⁴⁹. When compared to the Prototypes method⁴⁸, the proposed method yields identical results, but it is slightly outperformed by the CNN proposed by⁴⁴ (99.16%) in the improved CNN-based method. However, our method (99.13%) achieves a higher accuracy than the CNN-based method proposed by⁴⁵ (95.00%) and the combination of MLP, DWT, and DCT⁴⁶ (94.90%). Additionally, it significantly surpasses methods like PCA, HOG, LBP, and SVM⁴⁷ (84.44%), demonstrating its competitive performance in traffic sign recognition. The results suggested that our method provides a balanced solution, achieving high accuracy with 99.13%.

Table 4 provides a comparison with other approaches on the TT100K dataset, evaluated based on mAP. The data presented in this table is sourced from⁴². The proposed method (93.84%) outperforms several leading techniques, including CCSPNet⁵⁰ (90.9%), Faster R-CNN⁵¹ (89.5%), and FCOS⁵² (88.1%). Notably, our method also exceeds the performance of FII-CenterNet⁵³ (89.4%) and is competitive with CAMF⁴² (92.7%). These findings highlight the effectiveness of our approach in achieving a high mAP on the TT100K dataset. Compared to other methods, our model demonstrates a clear advantage in terms of mAP. It's suggested that it not only achieves high accuracy but also maintains superior precision and recall balance across different classes. This reinforces the robustness of our method in multi-class traffic sign recognition tasks. In addition, our method can achieve an accuracy of 99.07% on the TT100K dataset.

Figure 6 illustrates the convergence curves for both training and validation accuracy, as well as the loss values. As can be seen, on the GTSRB dataset, the training accuracy gradually increased from an initial 49.89% to 99.83%, and the validation accuracy also steadily increased from 51.99% to 99.84%, while the loss value gradually decreased from 0.69 to 0.06. The convergence patterns of accuracy and loss validate the performance of the Siamese-Mamba&VGG-16 network. The model achieves high accuracy on both the training set and the verification set and maintains a low loss value. This indicates that the model possesses strong learning capabilities and can effectively generalize across different scenarios, thus making it highly suitable for TSR tasks.

The Siamese-Mamba&VGG-16 network can achieve a high accuracy of over 99%, which is closely related to the HN-FR module. Figure 7 illustrates the input and output feature maps generated by the HN-FR module, showcasing its advantages in feature processing.

In SNN, the detection outcomes for traffic signs are typically determined by the similarity between sample pairs. After normalization, the similarity value generally ranges from 0 to 1, where values closer to 1 indicate a positive sample pair, and values closer to 0 correspond to a negative sample pair. It can be found from Table 5 that the similarity between two different types of traffic signs with similar colors and shapes is still very low. However, the similarity between two traffic signs of the same category with problems such as fuzzy occlusion can be as high as 0.95.

As shown in Fig. 8, although the proposed Siamese-Mamba & VGG-16 network achieves high accuracy across all datasets, it occasionally misclassifies traffic signs under extreme visual degradation, such as severe occlusion, motion blur, or low illumination. In these cases, features become less distinguishable, reducing similarity scores between true positive pairs. Low-resolution or heavily weathered signs may also cause confusion between visually similar categories. Analysis of misclassified samples from GTSRB, TSRD, and TT100K shows that only about 0.6% of test images were misclassified, mostly involving partially occluded or blurred signs (Fig. 8). According to the statistics of misclassified data, samples under blurry, occluded, and low-light conditions account for a

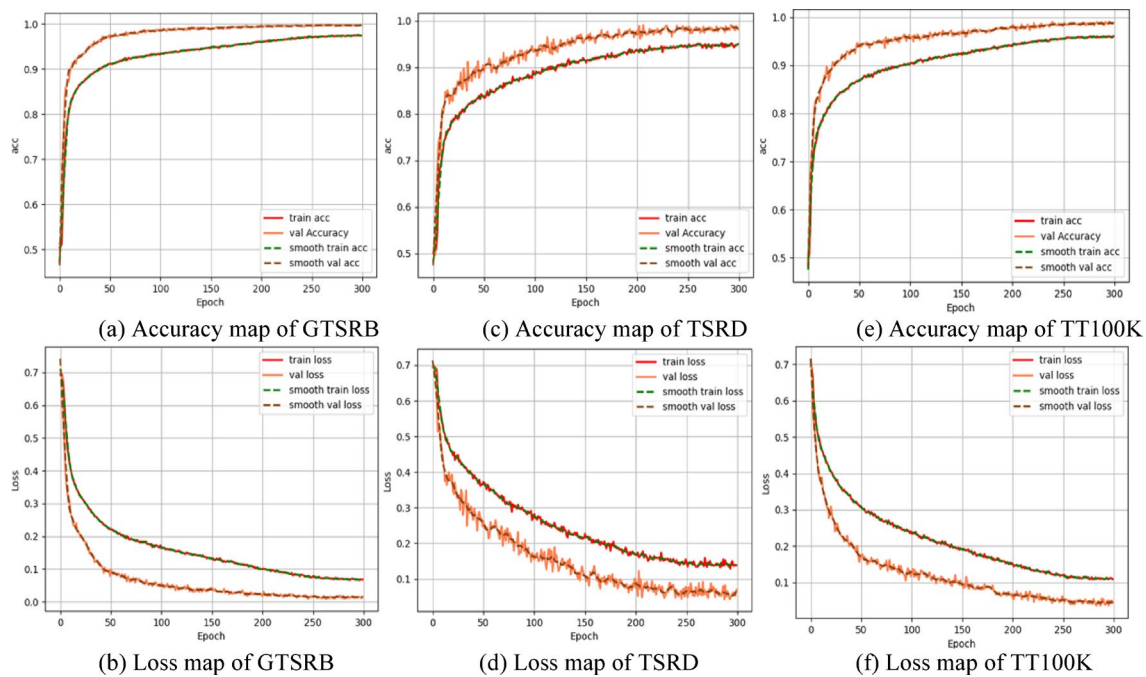


Fig. 6. Raw values of training loss and accuracy, along with their smoothed counterparts. (a) Accuracy map of GTSRB, (c) Accuracy map of TSRD, (e) Accuracy map of TT100K, (b) Loss map of GTSRB, (d) Loss map of TSRD, (f) Loss map of TT100K.

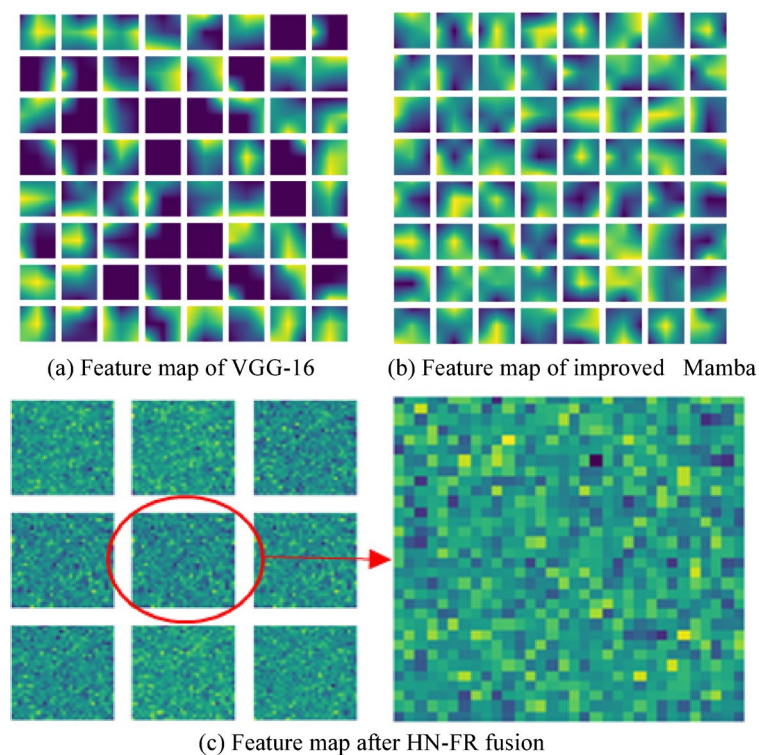


Fig. 7. Feature maps before and after fusion. (a) Feature map of VGG-16, (b) Feature map of improved Mamba, (c) Feature map after HN-FR fusion.









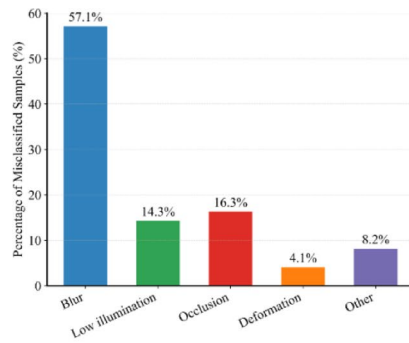
Traffic sign _{s(1)}	Traffic sign _{s(2)}	Basic information	
		Image pair	Positive sample
		similarity	0.9897
		Image pair	Negative sample
		similarity	0.1598
		Image pair	Negative sample
		similarity	0.0003
		Image pair	Positive sample
		similarity	0.9547

Table 5. Object classification result of traffic signs.



(a) Misclassified samples by error type



(b) Blur-Induced Error Image Pair



(c) Occlusion Error Image Pair



(d) Low-Resolution Error Image Pair

Fig. 8. Misclassified Image Pair. (a) Misclassified samples by error type, (b) Blur-Induced Error Image Pair, (c) Occlusion Error Image Pair, (d) Low-Resolution Error Image Pair.

large proportion, with blur accounting for 57.1% of the total misclassifications. Despite these limitations, the network still correctly recognizes many challenging signs, demonstrating that it maintains strong discriminative ability under difficult conditions. Overall, the method is robust in most real-world scenarios, achieving high recognition rates under illumination variations and background interference.

The designed fusion framework allows flexible use of different backbone networks. Table 6 presents a comparison of the performance on three public traffic sign datasets (TT100K, GTSRB, and TSRD) using four backbone networks (MobileNetV2, VGG-16, ResNet50, and AlexNet). The experimental results show

network	Dataset	Accuracy	Accuracy improvement	Average improvement
MobileNetV2 vs. Proposed method	TT100k	86.74% vs. 98.85%	12.11%	9.69%
	GTSRB	85.00% vs. 99.94%	14.94%	
	TSRD	96.90% vs. 98.91%	2.01%	
VGG-16 vs. Proposed method	TT100k	94.80% vs. 99.07%	4.27%	7.78%
	GTSRB	98.88% vs. 99.83%	0.95%	
	TSRD	81.01% vs. 99.13%	18.12%	
ResNet50 vs. Proposed method	TT100k	95.43% vs. 98.91%	3.48%	2.35%
	GTSRB	98.44% vs. 99.86%	1.42%	
	TSRD	97.24% vs. 99.39%	2.15%	
AlexNet vs. Proposed method	TT100k	62.50% vs. 98.70%	36.2%	23.40%
	GTSRB	85.77% vs. 99.84%	14.07%	
	TSRD	78.97% vs. 98.91%	19.94%	

Table 6. Performance comparison and improvement of different networks on multiple Datasets.

Method	Memory usage (MB)	Inference time (ms)	Hardware platform
propose method with MobileNetV2	202.92	21.58	RTX 4070 Ti
	29.87	250.13	Jetson Orin Nono
propose method with VGG-16	261.16	22.70	RTX 4070 Ti
	32.02	229.49	Jetson Orin Nono
propose method with ResNet50	200.99	22.71	RTX 4070 Ti
	31.42	263.20	Jetson Orin Nono
propose method with AlexNet	201.22	16.39	RTX 4070 Ti
	29.34	211.30	Jetson Orin Nono

Table 7. Performance comparison and improvement of different networks on multiple Datasets.

Model configuration	Accuracy(%)	Inference time (ms)	Parameters (M)
Full Model	99.13	22.70	161.63
Without DS Block	99.03	24.10	168.57
Without HN-FR Module	98.79	22.09	154.74
Without the improved Mamba	98.55	21.28	152.97

Table 8. Results of the ablation experiment.

that the proposed fusion framework significantly improves recognition accuracy across all networks and datasets. Specifically, compared with the four networks, SM&MobileNetV2, SM&VGG-16, SM&ResNet50, and SM&AlexNet achieve average accuracy improvements of 9.69%, 7.78%, 2.35%, and 23.40% on the three datasets, respectively, confirming the effectiveness of the proposed HN-FR module and the improved Mamba structure in feature fusion and information enhancement. Notably, for lightweight networks such as MobileNetV2, the fused models exhibit larger gains in accuracy, indicating that the proposed framework can effectively enhance discriminative capability. Overall, the proposed method demonstrates good generalization and robustness across different network architectures and datasets.

Table 7 compares the memory usage and inference speed of our four proposed networks on a PC (RTX 4070 Ti) and an embedded device (Jetson Orin Nano, 67 TOPS, TensorRT-accelerated). The results show that GPU memory usage on the embedded device is significantly reduced (by approximately 7–8 times), while the inference speed is slightly slower. As mentioned before, the proposed method follows the strategy that maximizes recognition performance under sufficient computational resources. Although the accuracy is better than existing methods in comparative experiments, the proposed method cannot run in real-time on resource limited platforms. In future work, inference time can be further reduced through lightweight design, such as model pruning or dimensionality reduction of key features. Moreover, with the continuous improvement of in-vehicle and embedded computing capabilities, our method has potential for deployment on embedded systems.

Ablation experiment

To evaluate the effect of the introduced modules, we carried out a series of ablation tests based on the TSRD dataset, and the specific data are shown in Table 8.

The ablation study results clearly demonstrate the critical role of the improved Mamba, the DS-Block, and the HN-FR Module in enhancing overall model performance. Removal of the DS-Block leads to a slight reduction in accuracy (by 0.1%) but results in a notable increase in inference time, indicating its primary contribution to improving model efficiency. In contrast, the removal of the HN-FR Module causes a more substantial accuracy drop (by 0.34%), underscoring its primary role in enhancing model precision. In addition, after removing the Mamba module, the overall accuracy shows a certain decrease (by 0.58), which proves the important role of the improved Mamba network in the overall network architecture.

Although the removal of these modules impacts inference time and parameter count, the most significant effect is the degradation in accuracy. These findings validate the effectiveness of the DS-Block in accelerating model inference and the HN-FR Module in boosting model accuracy, highlighting their respective roles in the overall system architecture.

Conclusion

In this paper, we propose to use the SNN framework to fuse the improved Mamba network and the mainstream CNN architectures. The proposed framework is designed to allow the CNN component to be easily replaced or upgraded, providing adaptability to different network architectures without affecting the core feature fusion mechanism. The exploitation of the DS Block structure can improve algorithmic efficiency by reducing computational parameters, which results in a decrease in computational complexity and an enhancement in computational efficiency. Specifically, the Heterogeneous Network Feature Reconstruction module is proposed to reconstruct the convolutional features. With the core objective of effectively merging and dimension-reducing feature vectors from different networks, the proposed neural network enhances performance in classification tasks. Experimental results indicate that the proposed SM&CNN outperforms existing traditional methods and other deep learning models on multiple public traffic sign datasets (such as GTSRB, TSRD, and TT100K). The proposed framework provides a promising solution for traffic sign recognition. In future work, we aim to integrate additional sources of information to further improve the accuracy and robustness of our framework under challenging conditions, such as severe occlusion, motion blur, and adverse weather. Moreover, the proposed strategy can be readily extended to a wider range of autonomous driving perception tasks, including multi-sensor fusion, complex scene understanding, and Vision–Language–Action models. For the issue of efficiency feasibility, we plan to conduct lightweight design, such as model pruning or dimensionality reduction of key features, to adapt to real-time operation on resource constrained platforms.

Data availability

All datasets used in this study are publicly available traffic sign image datasets: GTSRB (German Traffic Sign Recognition Benchmark), hosted by the Institute for Neural Information Processing, Ruhr-University Bochum (https://benchmark.ini.rub.de/gtsrb_dataset.html) TSRD (Traffic Sign Recognition Dataset), provided by the Natural Language Processing and Social Computing Research Center, Chinese Academy of Sciences (<https://nlpr.ia.ac.cn/pal/trafficdata/recognition.html>) TT100K (Tsinghua-Tencent 100 K), provided by the Computer Graphics Laboratory, Tsinghua University (<https://cg.cs.tsinghua.edu.cn/traffic-sign/>) These datasets are maintained by the respective institutions and are freely accessible for research purposes.

Received: 31 August 2025; Accepted: 30 October 2025

Published online: 28 November 2025

References

- Gamez Serna, C. & Ruichek, Y. Traffic Signs Detection and Classification for European Urban Environments, *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4388–4399, <https://doi.org/10.1109/TITS.2019.2941081> (2020).
- Zhao, R., Tang, S. H., Shen, J., Supeni, E. E. B. & Rahim, S. A. Enhancing autonomous driving safety: a robust traffic sign detection and recognition model TSD-YOLO. *Sig. Process.* **225**, 109619 (2024).
- Li, J. & Wang, Z. Real-Time Traffic Sign Recognition Based on Efficient CNNs in the Wild, *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 975–984, (2019). <https://doi.org/10.1109/TITS.2018.2843815>
- Abdel-Salam, R., Mostafa, R. & Abdel-Gawad, A. H. RIECNN: real-time image enhanced CNN for traffic sign recognition, *Neural Comput. Appl.*, vol. 34, no. 8, pp. 6085–6096, <https://doi.org/10.1007/s00521-021-06762-5> (2022).
- Lee, H. S. & Kim, K. Simultaneous traffic sign detection and boundary Estimation using convolutional neural network. *IEEE Trans. Intell. Transp. Syst.* **19** (5), 1652–1663. <https://doi.org/10.1109/TITS.2018.2801560> (2018).
- Wong, A., Shafiee, M. J. & Jules, M. S. MicronNet: A highly compact deep convolutional neural network architecture for Real-Time embedded traffic sign classification. *IEEE Access.* **6**, 59803–59810. <https://doi.org/10.1109/ACCESS.2018.2873948> (2018).
- Yue, Y. & Li, Z. MedMamba: Vision Mamba for Medical Image Classification, Sep. 29, arXiv: arXiv:2403.03849. <https://doi.org/10.48550/arXiv.2403.03849> (2024).
- Attarmoghaddam, N. & Li, K. F. An Area-Efficient FPGA Implementation of a Real-Time Multi-Class Classifier for Binary Images, *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 69, no. 4, pp. 2306–2310, <https://doi.org/10.1109/TCSII.2022.3148228> (2022).
- Kassani, P. H. & Teoh, A. B. J. A new sparse model for traffic sign classification using soft histogram of oriented gradients. *Appl. Soft Comput.* **52**, 231–246. <https://doi.org/10.1016/j.asoc.2016.12.037> (2017).
- Xie, H., He, J., Lu, Z. & Hu, J. Two-Level Feature-Fusion Ship Recognition Strategy Combining HOG Features with Dual-Polarized Data in SAR Images, *Remote Sens.*, **15**, 18, 4393, doi: <https://doi.org/10.3390/rs15184393>. (2023).
- Satti, S. K., Maddula, P. & Ravipati, N. V. Unified approach for detecting traffic signs and potholes on Indian roads. *J. King Saud University-Computer Inform. Sci.* **34** (10), 9745–9756 (2022).
- Wu, H. & Zhou, J. Privacy leakage of SIFT features via deep generative model based image reconstruction. *IEEE Trans. Inf. Forensics Secur.* **16**, 2973–2985. <https://doi.org/10.1109/TIFS.2021.3070427> (2021).
- Medhat, S., Abdel-Galil, H., Aboutabl, A. E. & Saleh, H. Iterative magnitude pruning-based light-version of AlexNet for skin cancer classification. *Neural Comput. Appl.* **36** (3), 1413–1428. <https://doi.org/10.1007/s00521-023-09111-w> (2024).
- An, Y., Yang, C. & Zhang, S. A lightweight network architecture for traffic sign recognition based on enhanced LeNet-5 network. *Front. Neurosci.* **18**, 1431033. <https://doi.org/10.3389/fnins.2024.1431033> (2024).

15. Bi, Z., Yu, L., Gao, H., Zhou, P. & Yao, H. Improved VGG model-based efficient traffic sign recognition for safe driving in 5G scenarios, *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 11, pp. 3069–3080, <https://doi.org/10.1007/s13042-020-01185-5> (2021).
16. Yu, J., Ye, X. & Tu, Q. Traffic Sign Detection and Recognition in Multiimages Using a Fusion Model With YOLO and VGG Network, *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16632–16642, <https://doi.org/10.1109/TITS.2022.3170354> (2022).
17. Dewi, C., Chen, R. C., Yu, H. & Jiang, X. Robust detection method for improving small traffic sign recognition based on spatial pyramid pooling, *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 7, pp. 8135–8152, <https://doi.org/10.1007/s12652-021-03584-0> (2023).
18. Yan, Y., Deng, C., Ma, J., Wang, Y. & Li, Y. A traffic sign recognition method under complex illumination conditions, *IEEE Access*, **11**, 39185–39196, <https://doi.org/10.1109/ACCESS.2023.3266825> (2023).
19. Latif, G. et al. Deep learning in transportation: optimized driven deep residual networks for Arabic traffic sign recognition, *Alexandria Eng. J.* **80**, 134–143 (2023).
20. Castruita Rodriguez, R. et al. Mexican traffic sign detection and classification using deep learning, *Expert Syst. Appl.*, vol. 202, p. 117247, <https://doi.org/10.1016/j.eswa.2022.117247> (2022).
21. Saha, S., Amit Kamran, S. & Shihab Sabbir, A. Total Recall: Understanding Traffic Signs Using Deep Convolutional Neural Network, in 2018 21st International Conference of Computer and Information Technology (ICCICT), pp. 1–6, <https://doi.org/10.1109/ICCICTECHN.2018.8631925> (2018).
22. Zatroch, F., Lehoczky, P., Masarykova, N. & Bencel, R. Evaluating a Siamese Network for Traffic Sign Recognition Using Synthetic Datasets, in 2024 International Symposium ELMAR, Sep. pp. 1–4, <https://doi.org/10.1109/ELMAR62909.2024.10694109> (2024).
23. Zhang, L. et al. TSD-DETR: A lightweight real-time detection transformer of traffic sign detection for long-range perception of autonomous driving, *Eng. Appl. Artif. Intell.* **139**, 109536, <https://doi.org/10.1016/j.engappai.2024.109536> (2025).
24. Lin, Z. et al. YOLO-LLTS: Real-Time Low-Light traffic sign detection via Prior-Guided enhancement and multibranch feature interaction, *IEEE Trans. Instrum. Meas.* **74**, 1–18, <https://doi.org/10.1109/TIM.2025.3604925> (2025).
25. Shen, L., You, L., Peng, B. & Zhang, C. Group multi-scale attention pyramid network for traffic sign detection, *Neurocomputing* **452**, 1–14, <https://doi.org/10.1016/j.neucom.2021.04.083> (2021).
26. Qu, S., Yang, X., Zhou, H. & Xie, Y. Improved YOLOv5-based for small traffic sign detection under complex weather, *Sci. Rep.* **13** (1), 16219, <https://doi.org/10.1038/s41598-023-42753-3> (2023).
27. Li, H. et al. A defense method based on attention mechanism against traffic sign adversarial samples, *Inform. Fusion*, **76**, 55–65, <https://doi.org/10.1016/j.inffus.2021.05.005> (2021).
28. Chen, T. et al. Mim-istd: Mamba-in-mamba for efficient infrared small target detection, *IEEE Transactions on Geoscience and Remote Sensing*, Accessed: 07 Apr 2025, <https://ieeexplore.ieee.org/abstract/document/10740056/>. (2024).
29. Li, G., Huang, Q., Wang, W. & Liu, L. Selective and multi-scale fusion Mamba for medical image segmentation, *Expert Syst. Appl.* **261**, 125518 (2025).
30. Zhang, R., Li, G., Qu, S., Wang, J. & Peng, J. Mamba-GIE: A visual state space models-based generalized image extrapolation method via dual-level adaptive feature fusion, *Expert Syst. Appl.* **264**, 125961 (2025).
31. Liu, Y., Liang, H. & Zhao, S. A lightweight visual Mamba network for image recognition under resource-limited environments, *Appl. Soft Comput.* **167**, 112294 (2024).
32. Li, Z. et al. Aug., Toward Effective Traffic Sign Detection via Two-Stage Fusion Neural Networks, *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 8, pp. 8283–8294, <https://doi.org/10.1109/TITS.2024.3373793> (2024).
33. He, H., Zhang, R., Zhang, Y. & Ren, J. Attentional aggregative interaction network for deep learning based recommender systems, *Neurocomputing* **547**, 126374, <https://doi.org/10.1016/j.neucom.2023.126374> (2023).
34. Jin, Y. et al. Multi-Feature fusion and enhancement single shot detector for traffic sign recognition, *IEEE Access*, **8**, 38931–38940, <https://doi.org/10.1109/ACCESS.2020.2975828> (2020).
35. Yu, B. et al. YOLO-MPAM: efficient real-time neural networks based on multi-channel feature fusion, *Expert Syst. Appl.* **252**, 124282, <https://doi.org/10.1016/j.eswa.2024.124282> (2024).
36. Huang, G., Xu, N., Lu, L., Ping, Z. & Hu, Z. Building Visual Maps From Two-Camera Views for Intelligent Vehicle Localization in Underground Parking Lots, *IEEE Trans. Intell. Veh.*, vol. 9, no. 2, pp. 3906–3916, <https://doi.org/10.1109/TIV.2023.3346193> (2024).
37. Que, Y. et al. Automatic classification of asphalt pavement cracks using a novel integrated generative adversarial networks and improved VGG model, *Eng. Struct.* **277**, 115406 (2023).
38. Wu, Z., Shen, C. & Van Den Hengel, A. Wider or deeper: revisiting the Resnet model for visual recognition, *Pattern Recogn.* **90**, 119–133 (2019).
39. Nan, Y., Ju, J., Hua, Q., Zhang, H. & Wang, B. A-MobileNet: An approach of facial expression recognition, *Alex. Eng. J.*, vol. 61, no. 6, pp. 4435–4444, (2022), <https://doi.org/10.1016/j.aej.2021.09.066>
40. Singh, I., Goyal, G. & Chandel, A. AlexNet architecture based convolutional neural network for toxic comments classification, *J. King Saud Univ. - Comput. Inf. Sci.* **34** (9), 7547–7558, <https://doi.org/10.1016/j.jksuci.2022.06.007> (2022).
41. Wan, Y., Shao, M., Cheng, Y. & Zuo, W. Image all-in-one adverse weather removal via dynamic model weights generation, *Knowl. Based Syst.* **302**, 112324 (2024).
42. An, F., Wang, J. & Liu, R. Road Traffic Sign Recognition Algorithm Based on Cascade Attention-Modulation Fusion Mechanism, *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 11, pp. 17841–17851, <https://doi.org/10.1109/TITS.2024.3439699> (2024).
43. Haque, W. A., Arefin, S., Shihavuddin, A. S. M. & Hasan, M. A. DeepThin: A novel lightweight CNN architecture for traffic sign recognition without GPU requirements, *Expert Syst. Appl.* **168**, 114481, <https://doi.org/10.1016/j.eswa.2020.114481> (2021).
44. Jain, A., Mishra, A., Shukla, A. & Tiwari, R. A Novel Genetically Optimized Convolutional Neural Network for Traffic Sign Recognition: A New Benchmark on Belgium and Chinese Traffic Sign Datasets, *Neural Process. Lett.*, vol. 50, no. 3, pp. 3019–3043, <https://doi.org/10.1007/s11063-019-09991-x> (2019).
45. Tabernik, D. & Skočaj, D. Deep Learning for Large-Scale Traffic-Sign Detection and Recognition, *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1427–1440, <https://doi.org/10.1109/TITS.2019.2913588> (2020).
46. Sapijaszko, G., Alobaidi, T. & Mikhael, W. B. Traffic Sign Recognition Based on Multilayer Perceptron Using DWT and DCT, in 2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 440–443, <https://doi.org/10.1109/MWSCAS.2019.8884897> (2019).
47. Soni, D., Chaurasiya, R. K. & Agrawal, S. Improving the Classification Accuracy of Accurate Traffic Sign Detection and Recognition System Using HOG and LBP Features and PCA-Based Dimension Reduction, Feb. 24, 2019, Social Science Research Network, Rochester, NY: 3358756, <https://doi.org/10.2139/ssrn.3358756>
48. Fu, H. & Wang, H. Traffic Sign Classification Based on Prototypes, in 2021 16th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Nov. pp. 7–10, <https://doi.org/10.1109/ISKE54062.2021.9755432> (2021).
49. Vashisht, M. & Kumar, B. Effective implementation of machine learning algorithms using 3D colour texture feature for traffic sign detection for smart cities, *Expert Syst.*, vol. 39, no. 5, p. e12781, <https://doi.org/10.1111/exsy.12781> (2022).
50. Hong, H., Zhou, Y., Shu, X. & Hu, X. CCSPNet-Joint: Efficient Joint Training Method for Traffic Sign Detection Under Extreme Conditions, in 2024 International Joint Conference on Neural Networks (IJCNN), pp. 1–8, <https://doi.org/10.1109/IJCNN60899.2024.10651346> (2024).
51. Yang, L. et al. Jul., An Improving Faster-RCNN With Multi-Attention ResNet for Small Target Detection in Intelligent Autonomous Transport With 6G, *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 7, pp. 7717–7725, <https://doi.org/10.1109/TITS.2022.3193909> (2023).

52. Tian, Z., Shen, C., Chen, H. & He, T. A simple and strong Anchor-free object detector. *IEEE Trans. Pattern Anal. Mach. Intell.* 1–1. <https://doi.org/10.1109/TPAMI.2020.3032166> (2020).
53. Fan, S. et al. Jan., FII-CenterNet: An Anchor-Free Detector with Foreground Attention for Traffic Object Detection, *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 121–132, <https://doi.org/10.1109/TVT.2021.3049805> (2021).

Author contributions

All authors contributed valuable input that significantly influenced the research, analysis, and development of the paper. **Gang Huang: ** Writing-review & editing, Methodology, validation, Supervision. **Huilin Cao: ** Writing-original draft, editing, Methodology, experiment planning and verification, investigation. **Jiayue Sun: ** Supervision. **Zhenling Chen: ** Supervision. **Zhe Zhou: ** Supervision.

Funding

National Natural Science Foundation of China (No. 52402494). Funding: Natural Science Foundation of Hubei Province (No. 2024AFB273 and 2024AFB139). Funding: Hubei Province Major Science and Technology Innovation Program (No. 2024BAA011). Funding: Hubei Key Laboratory of Power System Design and Test for Electrical Vehicle (No. ZDSYS202504).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Z.Z.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025