

# Road Surface Recognition Method of Automatic Driving System Based on Improved ResNet Model

Lixia Niu, Binhe Sun, *Member, IAENG*, Shize Liu, Sijia Liang

**Abstract**—Accurate road surface recognition is crucial for autonomous driving, as complex real-world environments impose higher demands on classification accuracy and generalization ability. This study proposes an improved ResNet model, integrating transfer learning with Domain Adversarial Neural Network (DaNN) and Efficient Channel Attention (ECA) to enhance model performance, optimize feature representation, and improve domain adaptation. The AdamW optimizer with a learning rate decay strategy is employed to mitigate overfitting and improve model robustness. A dataset comprising six distinct road surface types was constructed, and the performance of various ResNet variants is systematically evaluated. Experimental results demonstrate that the improved ResNet-152 model achieves 94.68% classification accuracy, a Micro-F1 score of 0.949, a reduced loss value of 0.3532, and an inference time of just 6.4954 milliseconds, making it suitable for real-time applications. Benefiting from transfer learning and computational optimizations, despite the increased depth and complexity of ResNet-152, its training time is reduced to 16.27 hours, outperforming other mainstream deep models. Comparisons with VGG, DenseNet, and Inception-ResNet-v2 architectures reveal that the improved ResNet-152 achieves an effective balance between classification accuracy and computational efficiency, providing strong support for high-precision, low-latency road surface recognition in autonomous driving.

**Index Terms**—Road surface type recognition, Transfer learning, Domain adversarial neural network, ECA attention mechanism, Automatic driving

## I. INTRODUCTION

With the rapid growth of autonomous driving technology, autonomous systems still confront various hurdles when navigating complicated road situations. In poor weather conditions such as ice, snow, slick, and water accumulation road surfaces diminish friction between the vehicle and the road, decreasing braking performance and increasing the likelihood of traffic accidents [1-2]. One of the most pressing difficulties that autonomous driving

technology must handle is the precise detection of road surface types under changing environmental circumstances. As a result, accurate road surface condition detection is critical for lowering traffic accident risks, increasing road safety, and greatly developing intelligent driving technology.

Traditional methods for road surface type recognition are grounded in tire dynamics principles, utilizing onboard vehicle sensors to measure road adhesion coefficients. [3-5] These approaches have stringent criteria for specific sensors and vehicle settings, and their applicability and accuracy are limited across weather situations. Computer vision has improved its image recognition performance as artificial intelligence technology has advanced. Computer vision-based road surface recognition methods can be broadly categorized into traditional machine learning and modern deep learning approaches. The literature [6] used LiDAR data from roads and a Support Vector Machine (SVM) to identify and distinguish five types of surfaces: asphalt, wood, marble, grass, and concrete. Traditional machine learning algorithms, however, struggle to produce good results in real-world traffic systems due to weather fluctuations, diverse road conditions, and variations in lighting. These algorithms also have long training times and limited adaptability. To address these challenges, researchers explored deep learning-based approaches. The literature [7] used RCNet (Regional Center for Nuclear Education & Training) to train models, allowing for the categorization and recognition of curved, dry, icy, rocky, and wet roads, demonstrating the effectiveness of deep learning in handling complex and diverse road conditions. The literature [8] employed visible light and thermal imagery with Convolutional Neural Networks (CNN) to classify icy, snowy, wet, and muddy road surfaces, achieving high accuracy on a dataset of 4,244 pedestrian crossing images. These studies highlight the potential of deep learning methods in overcoming the adaptability and robustness limitations of traditional machine learning approaches.

Deep learning-based road surface recognition systems have emerged as a focal point of research in autonomous driving technologies. Nonetheless, enhancing the precision and efficacy of such recognition continues to encounter several hurdles [9], including: 1) Variations in image acquisition conditions result in inconsistent datasets, and alterations in illumination intensity impact the recognition accuracy of identical road surfaces. 2) The categorization of road surface types is frequently inadequately detailed. The literature [10] indicates a substantial disparity in adhesion coefficients between snow and semi-melted snow, signifying they cannot be categorized as the same type. 3) Using a

Manuscript received February 10, 2025; revised April 26, 2025.

This work was funded by the National Natural Science Foundation of China under the 52174184.

Lixia Niu is a professor of Liaoning Technical University, Huludao 125000, China. (e-mail: 18342986199@163.com).

Binhe Sun is a postgraduate student of Liaoning Technical University, Huludao 125000, China. (Corresponding Author, e-mail: 2562289225@qq.com).

Shize Liu is a postgraduate student of Liaoning Technical University, Huludao 125000, China. (e-mail: 2578792840@qq.com).

Sijia Liang is a postgraduate student of Liaoning Technical University, Huludao 125000, China. (e-mail: 2561943726@qq.com).

dataset with a limited number of samples makes it challenging to comprehensively capture data features and ensures diversity, thereby restricting the model's generalization ability. 4) Significant variations in texture, color, and lighting intensity can occur even among identical road surface types. Training techniques for models necessitate considerable computational resources to manage extensive datasets; however, the computational capabilities of cars are constrained.

To address the aforementioned issues, this paper proposes a recognition method for six different road surface types. To enhance data diversity, a large-scale dataset is constructed, and the range of road surface categories will be expanded, thereby addressing issues related to limited sample size and insufficiently defined classification types. Additionally, the dataset includes samples of the same road surface under identical acquisition conditions but with varying illumination intensities, textures, and colors. It also classifies snow and semi-melted snow as separate categories, enabling the model to learn road surface features more accurately under different environmental conditions, thereby enhancing recognition robustness. Meanwhile, the optimized model architecture improves training efficiency and reduces computational costs. The ResNet model is enhanced with transfer learning and Domain-Adversarial Neural Networks (DaNN) to increase the effective dataset size without raising training costs. This method minimizes inconsistencies across various data distributions, improves the model's generalization capability across diverse domains, and facilitates feature extraction and classification of distinct road surface types with less iterations. The incorporation of the Efficient Channel Attention (ECA) mechanism enhances the model's focus on critical features within road surface images under complex conditions, thereby improving its ability to accurately recognize challenging road surface states such as ice, rain, snow, and semi-melted snow. This also diminishes the utilization of computer resources, rendering the strategy more feasible for practical applications.

## II. RESEARCH METHOD

### A. ResNet

In research on road surface type recognition, the dataset typically comprises a substantial number of samples characterized by high complexity. To improve the model's recognition capabilities, it is essential to augment the network depth. However, as the number of layers increases, problems such as gradient vanishing [11-13], and gradient explosion [14-15] emerge. Gradient vanishing transpires when the error gradient value falls below 1 during model training. In backpropagation, each gradient computation is multiplied by a factor less than one, resulting in the gradient ultimately converging to zero. This leads to negligible weight adjustments and diminishes the model's learning efficacy. Conversely, gradient explosion occurs when the error gradient value surpasses 1. In backpropagation, each gradient computation is multiplied by a value exceeding 1, resulting in an exponential escalation of the gradient, potentially culminating in infinity. Both gradient vanishing and gradient explosion impair the model's training performance, hindering the effective training of deep networks.

ResNet (Residual Network) is a deep neural network architecture, extensively utilized in computer vision tasks including image classification, object detection, and semantic segmentation [16-19]. ResNet features an exceptionally deep network architecture, with the number of layers reaching up to eight times that of conventional classification models such as VGG [20] and Inception [21], while maintaining relatively low computational complexity.

ResNet utilizes residual blocks to enable skip connections. This design effectively mitigates problems such as gradient vanishing or gradient explosion, even with substantial increases in network depth, thus preventing model degradation. The residual function of a residual block can be articulated as:

$$F(x) = H(x) - x \quad (1)$$

$$H(x) = F(x, \{W_i\}) + x \quad (2)$$

$$H(x) = x + W_2\sigma(W_1x + b_1) + b_2 \quad (3)$$

where,  $x$  denotes the input, while  $H(x)$  signifies the output of the residual block. The  $F(x, \{W_i\})$  residual mapping function comprises multiple convolutional, activation, and normalization operations, with  $W_i$  denoting the weight coefficients of these operations.  $W_1$  and  $W_2$  represent the weight matrices of the convolutional layers, while  $b_1$  and  $b_2$  are the bias terms. The function  $\sigma$  denotes the activation function, such as ReLU. Fig 1 illustrates the principle of the residual block. This configuration facilitates the direct propagation of the gradient through the shortcut connection, guaranteeing efficient gradient transmission in exceedingly deep networks. This alleviates the vanishing gradient issue and markedly enhances deep networks' training efficiency and performance.

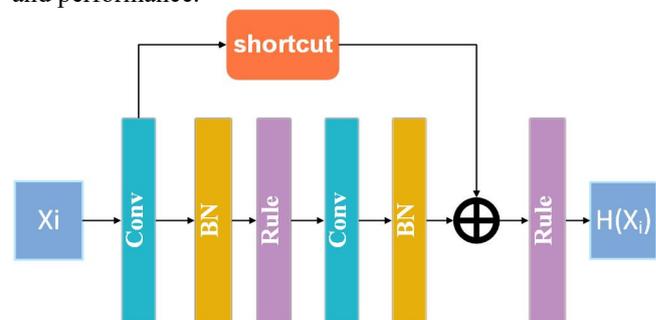


Fig. 1. Residual block structure

### B. Transfer learning

Transfer learning has emerged as a pivotal methodology in deep learning domains [22]. When training models on large-scale datasets, increasing their depth and complexity is often necessary to improve performance. Training a sophisticated convolutional neural network requires significant computational resources and extended iterative procedures. Transfer learning mitigates this issue by transferring pre-trained model parameters from one task to a new model, facilitating the training of the latter. This method obviates the necessity of retraining all parameters from the beginning on the new dataset, thereby enhancing training efficiency and markedly reducing computational resource utilization and training duration.

Transfer learning encompasses transductive transfer learning [23], inductive transfer learning [24], and

unsupervised transfer learning [25]. The fundamental concept of transfer learning revolves around domains consisting of data features and their distributions. The dataset utilized for pre-training the model is referred to as the source domain, whereas the new dataset for subsequent training is termed the target domain, as depicted in Fig.2.

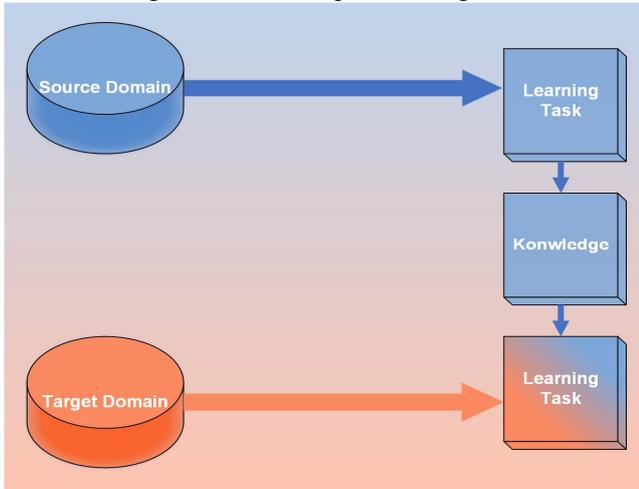


Fig. 2. Transfer learning flowchart

Assume the source domain dataset is  $D_S = \{(x_i^S, y_i^S)\}_{i=1}^{N_S}$  and the target domain dataset is  $D_T = \{(x_i^T, y_i^T)\}_{i=1}^{N_T}$ , where  $x_i^S$  and  $x_i^T$  represent the input data of the source and target domains, respectively,  $y_i^S$  and  $y_i^T$  are the corresponding labels. During transfer learning, the model is initially trained on the source domain by optimizing the objective function:

$$L_S = -\frac{1}{N_S} \sum_{i=1}^{N_S} y_i^S \log(p(y_i^S | x_i^S; \theta)) \quad (4)$$

where,  $p(y_i^S | x_i^S; \theta)$  represents the class probabilities given by the model, and  $\theta$  denotes the model parameters.

$$L_T = -\frac{1}{N_T} \sum_{i=1}^{N_T} y_i^T \log(p(y_i^T | x_i^T; \theta')) \quad (5)$$

where,  $\theta$  represents the initial weight obtained from pre-training on the source domain, and they are further adjusted on the target domain using optimization methods such as gradient descent.

Inductive transfer learning is utilized when tasks in the source domain and target domain differ yet possess analogous features. This method employs ResNet model that has been pre-trained on an extensive image dataset, incorporating pre-existing parameters. Only the output layer is modified to satisfy the particular demands of the new task. By training solely the altered output layer and constraining the parameters in the other neural layers, the model preserves the training knowledge from the extensive image dataset while markedly reducing computational expenses and time.

### C. Domain-Adversarial Neural Network

DaNN is a effective domain adaptation approach that employs adversarial training methods to enable the model's acquisition of domain-invariant feature representations, thus enhancing its generalization capacity for tasks in the target domain [26-27]. The fundamental concept of DaNN is to

acquire a feature representation that is both discriminative for classification tasks and invariant to the originating domain of the data. Domain adaptation is a critical challenge in transfer learning, especially when there is a discrepancy between the data distributions of the source and target domains. In transfer learning contexts where the data distributions of the source and target domains diverge while the tasks remain identical, the primary objective of domain adaptation is to mitigate the distributional discrepancies between the two domains, thereby enhancing knowledge transfer.

The DaNN architecture comprises three components: the feature extraction network  $G_f$ , the domain classification network  $G_d$ , and the label classification network  $G_y$ , as illustrated in Fig.3. The loss  $L_D$  denotes the loss from the domain classification network, while the loss  $L_Y$  signifies the loss from the label classification network. The Gradient Reversal Layer (GRL) functions as the gradient reversal layer. The feature extraction network extracts feature representations from the input data, whereas the domain classification network predicts the input data's domain, denoted as  $f = G_f(x)$ . The domain classification network identifies domain characteristics by assessing whether the input features originate from the source or target domain.  $y = G_y(f)$  represent the domain label, and the loss function is articulated as:

$$L_D = -\frac{1}{N} \sum_{i=1}^N [d_i \log D(G_f(x_i)) + (1-d_i) \log(1-D(G_f(x_i)))] \quad (6)$$

In forward propagation, the Gradient Reversal Layer (GRL) passes the input features without alteration. During backpropagation, the gradient direction is inverted, establishing an antagonistic relationship between the feature extraction network and the domain classification network. This configuration improves the feature extraction network's capability to extract label-specific information and bolsters the label classification network's ability to recognize labels. Consequently, the feature extraction network acquires domain-invariant characteristics, facilitating precise identification of sample labels.

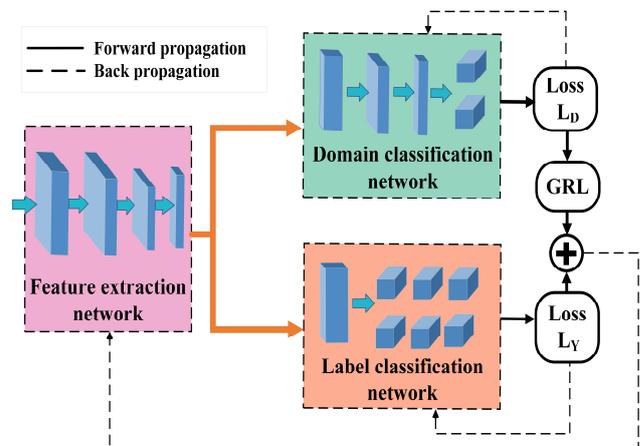


Fig. 3. Domain adversarial neural networks

### D. ECA Attention Mechanism

The attention mechanism is extensively utilized in convolutional neural networks to enhance model

performance and feature representation abilities [28]. ECA mechanism seeks to mitigate the computational redundancy inherent in conventional channel attention mechanisms. It produces channel attention weights through global average pooling on the channel dimension, subsequently applying these weights to each channel of the feature map, thereby modulating the feature intensity of each channel accordingly. Fig.4 illustrates the architecture of the ECA attention mechanism.

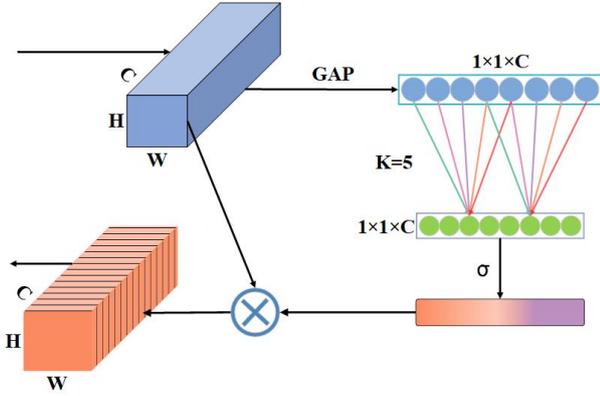


Fig. 4. ECA attention mechanism

The ECA attention mechanism, in contrast to conventional channel attention mechanisms, does not depend on intricate multi-layer perceptron architectures [29]. Instead, it utilizes efficient convolution operations that diminish computational expenses. The primary procedure of the ECA attention mechanism is as follows:

Initially, execute global average pooling on the input feature map to derive the global semantic representation for each channel:

$$y_c = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W x_c(i, j) \quad (7)$$

where  $x_c(i, j)$  represents the value of the  $C$  channel of the feature map at position  $(i, j)$ , and  $H$  and  $W$  denote the height and width of the feature map, respectively.

ECA employs a one-dimensional convolution operation to capture local dependencies among channels, with the kernel size adaptively modified according to the number of channels.

$$Z_c = \sigma(\text{Conv1D}(y)) \quad (8)$$

where  $\sigma$  represents the Sigmoid activation function, and  $Z_c$  denotes the channel weights obtained through the convolution operation.

The kernel size  $k$  is adaptively determined by the number of channels  $C$  as follows:

$$k = g(C) = \lceil \log_2(C) \gamma \rceil_{\text{odd}} \quad (9)$$

where:  $g(C)$  is the adaptive kernel size calculation function based on channel number  $C$ ;  $\gamma$  is a hyperparameter used to adjust the kernel size;  $\lceil \cdot \rceil_{\text{odd}}$  represents taking the nearest odd number to ensure the kernel size is odd.

This approach allows the kernel size to be automatically adjusted according to the input feature channels.

The derived channel weights are applied to the original feature map to effect weighted modulation across the channels:

$$\hat{x}_c = Z_c \bullet x_c \quad (10)$$

The ECA attention mechanism is an innovative channel attention method that substitutes the fully connected layers in the original SENET with  $1 \times 1$  convolutional kernels. This alteration diminishes the model parameters and renders it more lightweight. The ECA attention mechanism improves network performance by efficiently recalibrating channel features. It can significantly enhance model efficacy, especially in image classification and object detection endeavors. Fig. 5 illustrates the network workflow that integrates the ECA attention mechanism.

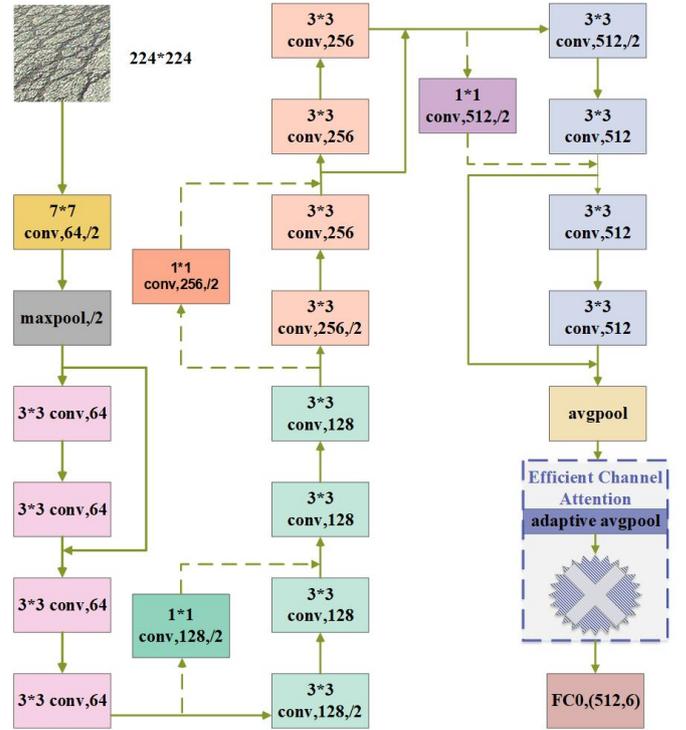


Fig. 5. ECA attention mechanism architecture

### E. AdamW Optimization and Learning Rate Decay

The Adam (Adaptive Moment Estimation) optimizer is among the most prevalent optimization algorithms utilized in deep learning. The SGD-M optimizer enhances the Stochastic Gradient Descent algorithm by integrating first-order momentum, whereas the AdaGrad optimizer incorporates second-order momentum into SGD. The Adam optimizer incorporates first-order and second-order momentum within the SGD framework [30], facilitating the efficient training of neural networks on extensive datasets. The integration of L2 regularization into the gradient update mechanism of the Adam optimizer may result in unstable weight adjustments.

The AdamW optimizer is an enhanced variant of Adam [31]. It mitigates the instability of weight updates by decoupling weight decay from the gradient update process, applying weight decay directly to the weight adjustments instead of incorporating it into the loss function. The revised formula for the bias-corrected momentum estimation subsequent to the application of AdamW is as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (11)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (12)$$

The bias correction is given by:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (13)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (14)$$

The parameter update is:

$$\theta = \theta_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} - \eta \lambda \theta_{t-1} \quad (15)$$

where  $\theta_t$  represents the parameters,  $g_t$  denotes the gradient,  $\eta$  is the learning rate,  $\lambda$  is the weight decay rate,  $m_t$  and  $v_t$  refer to the first and second moment estimates, respectively.  $\beta_1$  and  $\beta_2$  are the exponential decay rates for the moment terms. After the AdamW optimization, weight decay is decoupled from gradient computation, leading to more stable weight updates and effectively preventing model overfitting, particularly in large-scale neural networks.

Learning rate decay is a technique employed during model training to regulate the learning rate, enhancing the convergence speed and stability of the model. The learning rate is a crucial hyperparameter in deep learning algorithms, as its value directly affects training efficacy. Should the learning rate remain invariant during the training process, various complications may emerge: 1. Insufficient Learning Rate: This leads to sluggish gradient descent, markedly prolonging training duration; 2. Excessive Learning Rate: Although it can expedite training, it may induce substantial fluctuations in the loss function value during later phases, hindering the model's attainment of a global optimum and causing oscillation around the minimum value [32]. The learning rate decay strategy progressively diminishes the model's learning rate throughout training, regulating the extent of parameter updates following each iteration. This study utilizes a periodic decay approach, wherein the learning rate is diminished to 0.1 times its existing value in the subsequent iteration. An elevated learning rate during initial phases accelerates convergence, whereas a progressively diminishing learning rate enhances stability as training advances.

### III. MODEL ESTABLISHMENT

#### A. Dataset

Training a network to identify road surface types necessitates a substantial dataset to comprehend diverse surface characteristics. An imbalance in the sample quantity for a single category, whether excessive or insufficient, can substantially impair the model's recognition and classification efficacy. This research develops the dataset utilizing the publicly accessible road image dataset [33] from Tsinghua University.

Most images are captured on urban asphalt roads, with significant variation in sample quantity across categories. To prevent dataset imbalance and minimize the influence of road material characteristics on the model, 204,273 pertinent images from the road surface image dataset are selected to establish a more balanced surface type dataset. The surface types are classified as dry, ice, snow, semi-melted snow, water, and wet, with each classification featuring images of diverse materials captured under varying lighting conditions, including direct sunlight, overcast skies and low-light environments. Due to the road surface dataset images being sized at 240×360, which is incompatible with the input

dimensions required by the ResNet model, transformations were employed. Utilize the Center Crop function to crop images to a dimension of 224×224. The cropped images were subsequently standardized and normalized, with select sample images from the dataset illustrated in Fig.6.

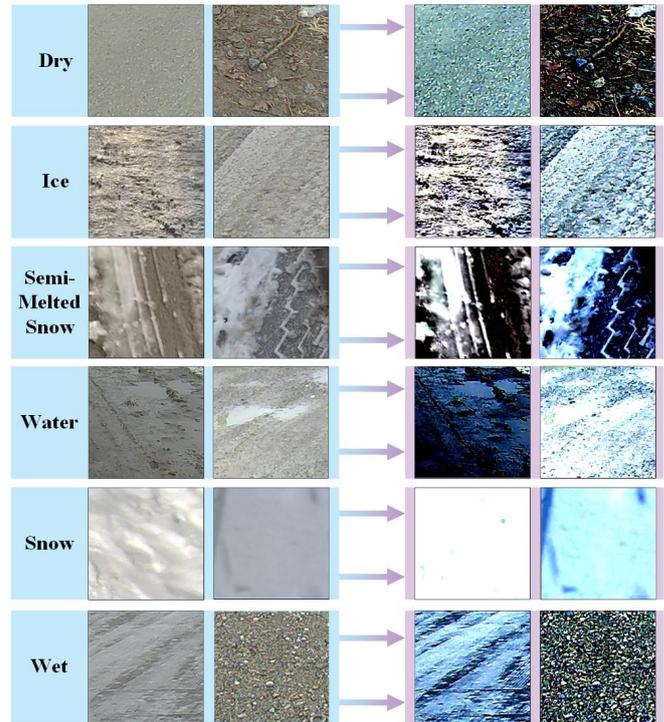


Fig. 6. Partial dataset

#### B. Hyperparameter Optimization

This study systematically optimizes and adjusts key hyperparameters, specifically the initial learning rate and batch size [34], to ensure model convergence and enhance performance during training. The initial learning rate is a pivotal hyperparameter that affects the convergence rate and ultimate accuracy of the model during deep learning training. To ascertain a suitable initial learning rate, the loss values and accuracy of the ResNet-152 model are evaluated across various initial learning rates, as illustrated in Table I. With a learning rate of 0.001, the loss value is 0.6243, and the accuracy is 0.7553. A learning rate of 0.001 yields comparatively lower loss and higher accuracy than other learning rates. While the accuracies for learning rates of 0.0005 and 0.0001 are marginally superior, they necessitate considerably extended training durations. Consequently, a learning rate of 0.001 attains an optimal equilibrium between accuracy and training duration, exhibiting enhanced training efficacy, and is selected as the ideal parameter.

TABLE I  
Comparison of model accuracy under different learning rates

Learning Rate	Loss	Accuracy
0.01	0.7098	0.6813
0.005	0.6639	0.7199
0.001	0.6243	0.7553
0.0005	0.6211	0.7562
0.0001	0.6204	0.7563

The batch size directly influences the training efficiency of the model and the stability of gradient updates. Choosing the batch size necessitates a thorough evaluation of training velocity and the model's convergence efficacy. Various batch sizes were tested in the experiments, and the results are presented in Table II. With a batch size of 32, the loss is 0.5973, and the accuracy is 0.7551. Compared to smaller batch sizes, the loss value experiences a slight increase, yet it remains relatively low while the accuracy nears its maximum potential. The performance regarding loss and accuracy at a batch size of 32 surpasses that of larger batch sizes. Additionally, the reduced batch size facilitates expedited convergence. Consequently, a batch size of 32 achieves an optimal equilibrium among loss, accuracy, and training duration, yielding more favorable convergence results.

TABLE II  
Comparison of model accuracy under different batch sizes

Batch-Size	Loss	Accuracy
16	0.5966	0.7559
32	0.5973	0.7551
64	0.6119	0.7434
128	0.6322	0.7398
256	0.6503	0.7302

C. Improved ResNet-152 Network Architecture

The research aims to optimize the ResNet-152 architecture to improve efficacy in classification and domain adaptation tasks. The improved model incorporates transfer learning, ECA module, and DaNN to enhance feature extraction and effectively tackle cross-domain issues.

ResNet-152 is a deep convolutional neural network that acquires more abstract feature representations by incorporating multiple Residual Block [35]. The conventional ResNet may exhibit constraints in the selection of global feature channels, which the enhanced network rectifies by integrating ECA) module. This ECA module is situated post the final residual block and adaptively assigns weights to various channels, augmenting the network's capacity to discern critical channel features like ice and snow, thus enhancing its representational efficacy.

DaNN, utilizing the Gradient Reversal Layer (GRL) [36] is introduced to enhance the model's generalization capability and mitigate discrepancies in data distribution across diverse domains. This network functions concurrently with the ResNet-152 feature extraction network, categorizing the input data features to diminish the domain disparity. The GRL inverts the gradient signs, reducing the domain classifier's capacity to identify domain-specific information as the model acquires feature representations, thus improving the domain invariance of these representations. The domain classification network consists of several fully connected layers and is trained with a cross-entropy loss function [37].

The backbone network is founded on the core architecture of ResNet-152, employing transfer learning. Subsequent to the final average pooling, the feature maps are fed into both the label classification network and the domain classification network. The label classifier generates class predictions, whereas the domain classifier forecasts domain labels. To

facilitate effective joint training, both classification loss and domain classification loss are optimized in a weighted fashion. An adaptive learning rate strategy is utilized during model training, commencing with an initial learning rate of 1e-3, which is progressively diminished throughout the training process. The proposed network architecture, through these enhancements, demonstrates enhanced feature extraction capabilities and effectively mitigates domain discrepancies, surpassing traditional ResNet networks in cross-domain tasks. The model's overall architecture is depicted in Fig.7.

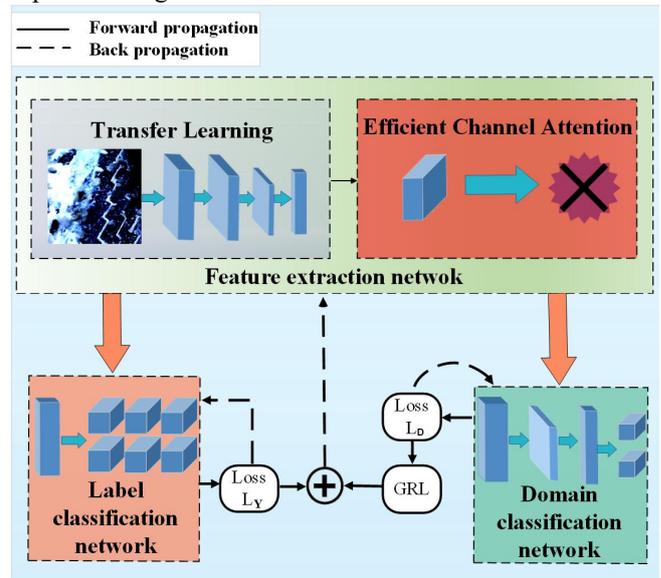


Fig. 7. Improved ResNet architecture

D. Ablation Study

To further evaluate the contributions of key components in the improved ResNet architecture, an ablation study was conducted. The experiment involved systematically removing specific modules, such as the ECA attention mechanism and DaNN, to examine their individual impacts on model performance. The models were trained under identical conditions, and their classification accuracy and loss values were compared. All models were trained under identical conditions to ensure a fair comparison.

Table III presents the results of the ablation study, where "Baseline" represents the original ResNet architecture without modifications. "ResNet + ECA" refers to the model incorporating the ECA attention mechanism, while "ResNet + DaNN" includes the domain adaptation module. The full model, "ResNet + ECA + DaNN," integrates both enhancements.

TABLE III  
Ablation study

Model Variant	Accuracy (%)	Loss
Baseline (ResNet-18)	68.5	0.812
ResNet + ECA	70.2	0.759
ResNet + DaNN	71.9	0.726
ResNet + ECA + DaNN	73.8	0.693

The results indicate that incorporating ECA improves feature representation by refining channel-wise attention,

leading to a performance gain. The addition of DaNN further enhances domain adaptation, reducing the distribution discrepancy and improving generalization. The combination of both modules achieves the highest accuracy and the lowest loss value, demonstrating their complementary effects.

#### IV. EXPERIMENTAL RESULTS ANALYSIS

The improved ResNet architecture was executed on a system equipped with an Intel i7-13620H CPU, RTX 4050 GPU, and 16 GB of RAM. The model was constructed utilizing the open-source PyTorch framework, with a batch size of 32, an initial learning rate of 0.01, and the AdamW optimizer. To accelerate training, CUDA was employed for GPU computation, along with cuDNN to optimize deep learning operations. Additionally, Automatic Mixed Precision (AMP) was utilized to enable lower-precision computations reducing memory consumption and improving computational efficiency. The training consisted of 20 epochs, utilizing the Negative Log Likelihood Loss (NLLoss) [38] as the loss function for the multi-class classification task, where a loss value approaching zero signifies superior model performance. Given the multi-class classification nature of the task, Micro-F1 score [39] was chosen as an additional evaluation metric due to its ability to account for both precision and recall across all classes, providing a more balanced view of model performance, especially in cases where class distribution is imbalanced.

$$L_{NLL} = -\sum_{i=1}^C y_i \log p(y_i) \quad (16)$$

where  $C$  represents the total number of classes,  $y_i$  denotes the one-hot encoded representation of the true class label, and  $p(y_i)$  refers to the predicted probability of class  $i$ , typically obtained from the Softmax output.

$$\text{Micro Precision} = \frac{\sum TP_i}{\sum TP_i + \sum FP_i} \quad (17)$$

$$\text{Micro Recall} = \frac{\sum TP_i}{\sum TP_i + \sum FN_i} \quad (18)$$

$$\text{Micro-F1} = 2 \times \frac{\text{Micro Precision} \times \text{Micro Recall}}{\text{Micro Precision} + \text{Micro Recall}} \quad (19)$$

Where:

- 1)  $TP_i$  (True Positives): The number of samples correctly predicted as class  $i$ .
- 2)  $FP_i$  (False Positives): The number of samples incorrectly predicted as class  $i$ .
- 3)  $FN_i$  (False Negatives): The number of samples that belong to class  $i$  but were not correctly predicted.

To evaluate the optimization effect of models, minimizing the loss and improving the accuracy were taken as the target of the model optimization [39]. Experiments utilized the identical validation set and model parameters, with the accuracy of each model compared as illustrated in Fig.8.

The initial accuracy of the ResNet-18 model in the experiments was 0.442. Following the integration of transfer learning with the ECA mechanism and DaNN, accuracy increased to 0.7366, indicating a notable improvement in feature extraction and cross-domain generalization abilities. After implementing transfer learning, the ECA mechanism, and DaNN, ResNet-50 attained an accuracy of 0.8099, signifying a substantial enhancement in effective transfer and feature extraction across varying data distributions. The deeper ResNet-101 and ResNet-152 models demonstrated enhanced performance, with ResNet-152 achieving an accuracy of 0.9468, thereby corroborating the efficacy of these improvements in deep networks.

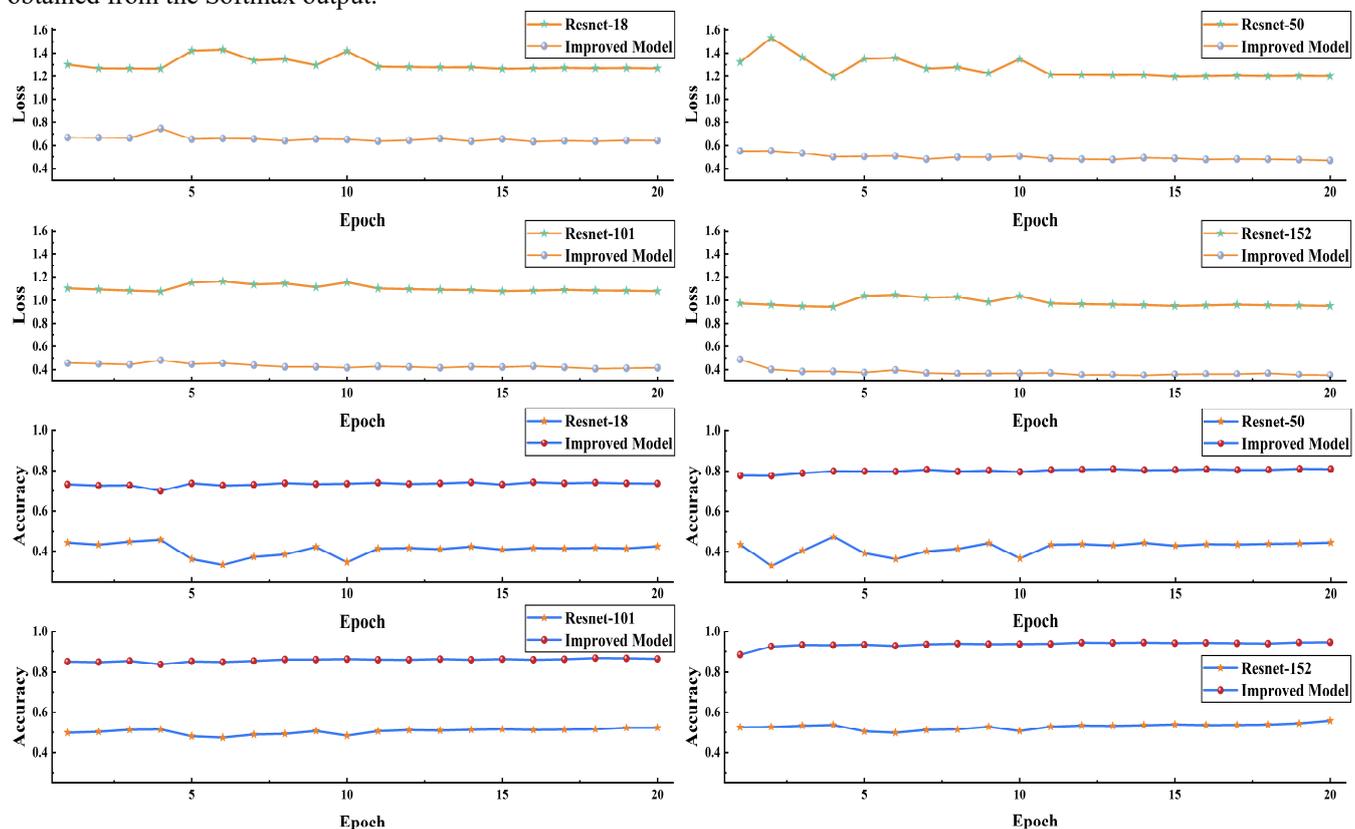


Fig. 8. Comparison of model before and after improvement

The initial loss for ResNet-18 was 1.301, which diminished to 1.2695 following 20 iterations of training the improved model. For ResNet-50, the initial loss was 0.724, decreasing to 0.4441, whereas ResNet-101 and ResNet-152 converged from 1.1032 and 0.9732 to 0.4149 and 0.3532, respectively. The ECA mechanism significantly improves feature extraction, especially in deep models like ResNet-50 and beyond, by balancing feature weights across channels and enhancing discriminative ability. When combined with transfer learning and DaNN, it accelerates convergence and increases accuracy. The DaNN module improves robustness, particularly in handling data from diverse domains, thus boosting performance in transfer learning tasks. In ResNet-152, DaNN enhances adaptability to target domain data, improving predictive accuracy. Together, the ECA and DaNN modules enhance the model's generalization and adaptability, reducing loss and increasing accuracy.

The recognition accuracy for various improved models on the same validation set, shown in Fig.9, demonstrates the efficacy of these algorithms in road surface type recognition. The residual connection in ResNet addresses gradient issues, preventing model degradation, while the improved model shows a steady increase in accuracy with deeper networks, significantly enhancing recognition capabilities.

The network exhibits enhanced performance when integrated with transfer learning, the ECA module, and DaNN, particularly highlighting exceptional generalization capabilities in cross-domain tasks. These enhancements guarantee that the model exhibits increased reliability in practical applications, offering substantial support for road surface type recognition in intricate scenarios.

The improved ResNet-152 model demonstrates excellent performance across key evaluation metrics, achieving a recognition accuracy of 94.68% and a loss value of 0.3532. Compared to other models, ResNet-152 shows a clear

advantage in accuracy, effectively distinguishing between different road surface types. The improved ResNet-18 model offers faster processing speed.; however, its lower accuracy results in weaker overall performance.

In the road surface recognition task within autonomous driving scenarios, Table IV summarizes the best recognition accuracy, loss value, Micro-F1 score, training time and average inference time for each model. The improved ResNet-152 achieves the best overall performance, with an accuracy of 94.68%, a loss value of 0.3532, a Micro-F1 score of 0.949, and the shortest inference time of 6.4954 milliseconds. This demonstrates the model's balanced performance in terms of both high accuracy and low latency.

In comparison, the DenseNet series models also show competitive performances. DenseNet-121 achieves an accuracy of 90.35%, with a loss value of 0.4124, a Micro-F1 score of 0.904 and an inference time of 8.1321 milliseconds. DenseNet-169 slightly improves accuracy to 91.87%, but with a longer inference time of 9.0564 milliseconds, indicating increased computational resource demands.

The Inception-ResNet-v2 model achieves an accuracy of 92.45% and a loss value of 0.3786, but its inference time is relatively long at 13.562 milliseconds. While this model performs well in complex scenarios, its higher computational complexity may limit its applicability in real-time systems that require faster responses.

The VGG models perform less favorably in this task. VGG-16 achieves an accuracy of 85.22% with an inference time of 15.234 milliseconds, while VGG-19 improves accuracy slightly to 86.70% but increases inference time to 17.012 milliseconds. Although the VGG models remain competitive in certain tasks, their low parameter efficiency and longer inference times make them unsuitable for real-time applications in autonomous driving.

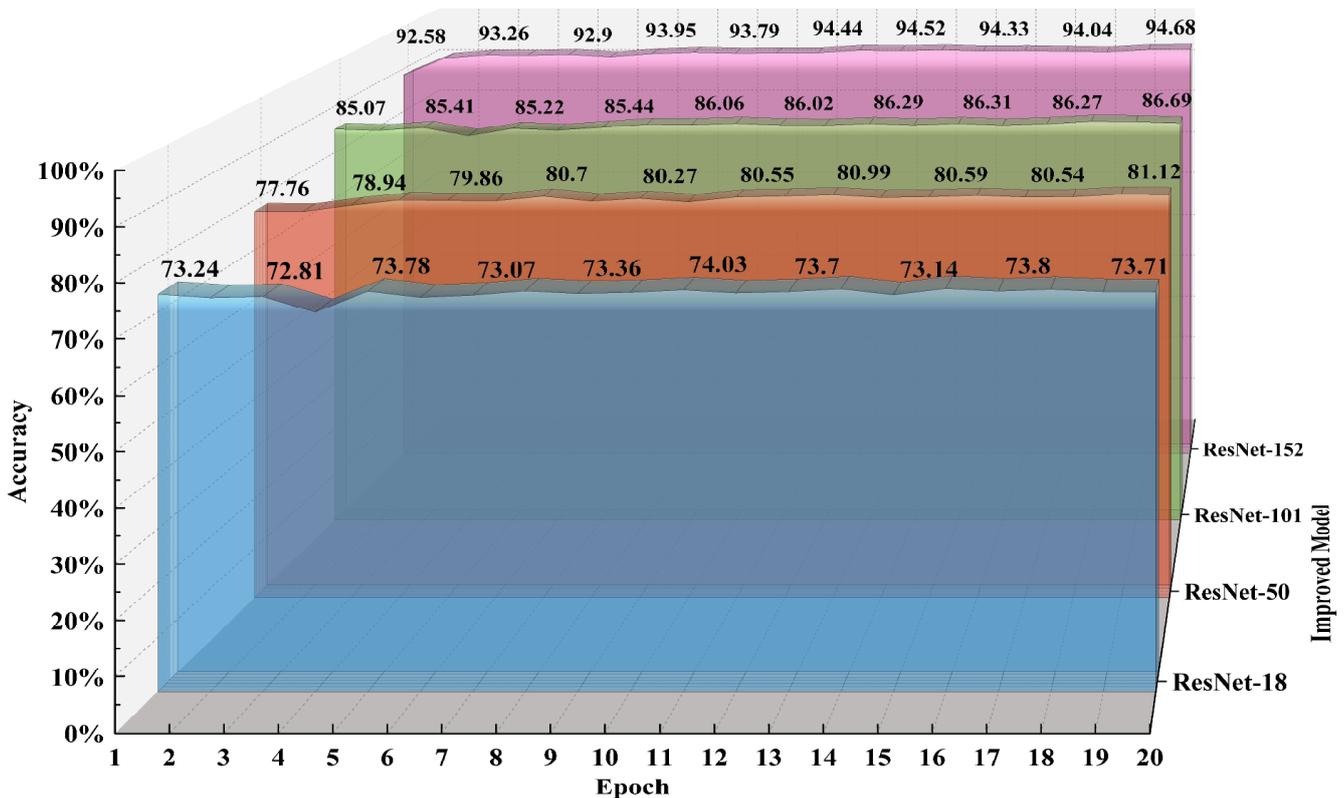


Fig. 9. Comparison of model accuracy values

TABLE IV  
Comparison of model accuracy and inference time

Model	The Best Accuracy (%)	NNLLoss	Mirco-F1 Score	Training Time (hours)	Average Inference Time (ms)
VGG-16	85.22	0.5648	0.846	18.2	15.234
VGG-19	86.70	0.5239	0.865	21.5	17.012
Inception-ResNet-v2	92.45	0.3786	0.927	36.8	13.562
DenseNet-121	90.35	0.4124	0.904	24.3	8.1321
DenseNet-169	91.87	0.3897	0.919	27.1	9.0564
<b>ResNet-152(Improved)</b>	<b>94.68</b>	<b>0.3532</b>	<b>0.949</b>	<b>16.27</b>	<b>6.4954</b>

Although ResNet-152 has a deeper architecture and more parameters, its training time is significantly shorter than DenseNet-121, DenseNet-169, and Inception-ResNet-v2. This efficiency is largely due to transfer learning, which allows the model to leverage pre-trained weights for faster convergence. Additionally, the use of CUDA, cuDNN, and AMP further optimizes GPU computation, accelerating training without sacrificing accuracy.

The improved ResNet-152 model demonstrates superior performance across all evaluated metrics, making it the optimal choice for road surface recognition in autonomous driving scenarios. The DenseNet series, with their combination of high accuracy and parameter efficiency, are well-suited for resource-limited environments. Inception-ResNet-v2 performs well in complex tasks but ranks lower due to its longer inference time. The VGG models, hindered by long inference times, are the least suited for applications requiring real-time performance. The improved ResNet-152 model, with its high accuracy and low latency, provides autonomous vehicles with fast, precise decision support, significantly improving overall system stability and safety.

## V. CONCLUSION

This paper presents an improved model for road surface type recognition using the ResNet algorithm, achieving notable reductions in computational resources and training time. By integrating transfer learning with domain adversarial neural networks, the ECA attention mechanism, the AdamW optimizer, and a learning rate decay strategy, the model enables rapid convergence in early training stages and addresses overfitting in later stages. The model's residual structures mitigate challenges like gradient vanishing and explosion as network depth and complexity increase. With the ECA attention mechanism boosting model accuracy and the domain adversarial neural network addressing data distribution differences in transfer learning, the model effectively facilitates knowledge transfer between domains.

Compared to both the original ResNet model and other widely used architectures such as VGG, Inception-ResNet, and DenseNet, the enhanced model achieves significantly lower loss values and higher accuracy, even with the same iteration count. The improved ResNet-152 model demonstrates excellent accuracy, reaching 94.68%, with an inference time of 6.4954 milliseconds, striking a balance between precision and efficiency. Moreover, the model achieves a high F1-score of 0.949, indicating excellent performance in terms of both precision and recall, and

ensuring reliable classification across different road surface categories. The model also demonstrates strong robustness and generalization across diverse road surface conditions, particularly through the incorporation of semi-melted snow into the recognition process. This enhancement highlights the model's adaptability to complex road surface scenarios, while maintaining high recognition accuracy across various real-world applications. Its high accuracy and low latency make the model well-suited to the real-time and precision demands of road surface type recognition in autonomous driving, providing a solid foundation for advancements in autonomous driving technologies.

## REFERENCES

- [1] Yi-Qiu TAN, Ji-Lu Li, and Hui-Ning XU, "Review on Friction Characteristics of Icy and Snowy Roads and Operational Risk Management," *China Journal of Highway and Transport*, vol. 35, no.7, pp1-17, 2022.
- [2] Sohini Roychowdhury, Min-Ming Zhao, Andreas Wallin, Niklas Ohlsson, and Mats Jonasson, "Machine Learning Models for Road Surface and Friction Estimation Using Front-camera Images," in *Proc. 2018 Int. Joint Conf. Neural Networks*, 8–13 July, 2018, Rio de Janeiro, Brazil, pp1–8.
- [3] Zhi-Wei Xu, Yong-Jie, Na Chen, and Yin-Feng Han, "Integrated Adhesion Coefficient Estimation of 3D Road Surfaces Based on Dimensionless Data-driven Tire Model," *Machines*, vol. 11, no.2, pp189, 2023.
- [4] Miao Yu, Bo Xiao, Zhan-Ping You., et al., "Dynamic Friction Coefficient between Tire and Compacted Asphalt Mixtures Using Tire-pavement Dynamic Friction Analyzer," *Construction and Building Materials*, vol. 258, 2020.
- [5] Xiao-Qiang Sun, Hou-Zhong Zhang, Ying-Fang Cai, Shao-Hua Wang, and Long Chen, "Hybrid Modeling and Predictive Control of Intelligent Vehicle Longitudinal Velocity Considering Nonlinear Tire Dynamics," *Nonlinear Dynamics*, vol. 97, no.2, pp1051–1066, 2019.
- [6] Aliakbar Matkan, Mohammad Hajeb, and S. Sadeghian, "Road Extraction from Lidar Data Using Support Vector Machine Classification," *Photogrammetric Engineering & Remote Sensing*, vol. 80, no.5, pp409-422, 2014.
- [7] Deepak Kumar Dewangan and Satya Prakash Sahu, "RCNet: Road Classification Convolutional Neural Networks for Intelligent Vehicle System," *Intelligent Service Robotics*, vol. 14, no.2, pp199-214, 2021.
- [8] Ce Zhang, Ehsan Nateghinia, Luis F. Miranda-Moreno, and Li-jun Sun, "Winter Road Surface Condition Classification Using Convolutional Neural Network (CNN): Visible Light and Thermal Image Fusion," *Canadian Journal of Civil Engineering*, vol. 49, no.4, pp569-578, 2022.
- [9] Marcus Nolte, Nikita Kister, and Markus Maurer, "Assessment of Deep Convolutional Neural Networks for Road Surface Classification," in *Proc. 21st Int. Conf. Intelligent Transportation Systems (ITSC)*, 16–19 October, 2018, Maui, HI, USA, pp. 381–386.
- [10] Adila Paerhati., "Safety driving Speed Test and Analysis of Semi-melted Road Surface with Snow," M.S. thesis, Dept. Civil Eng., Xinjiang Univ., Urumqi, China, 2019.
- [11] Zheng Hu, Jiao-Jiao Zhang, and Yun Ge, "Handling Vanishing Gradient Problem Using Artificial Derivative," *IEEE Access*, vol. 9, pp22371-22377, 2021.

- [12] Yao Lu, Stephen Gould, Thalajasingam Ajanthan, "Bidirectionally Self-Normalizing Neural Networks," *Neural Networks*, vol. 167, pp283-291, 2023.
- [13] Yu-Min Chen, Xiao Zhang, Ying Zhuang, Bing-Yu Yao, and Bin Lin, "Granular Neural Networks with a Reference Frame," *Knowledge-Based Systems*, vol. 260, 2023.
- [14] Zhao-Dong Chen, Lei Deng, Bang-Yan Wang, Guo-Qi Li, and Yuan Xie, "A Comprehensive and Modularized Statistical Framework for Gradient Norm Equality in Deep Neural Networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp13-31, 2022.
- [15] Qian Kang, Deng-Xiu Yu, Kang Hao Cheong, and Zhen Wang, "Deterministic Convergence Analysis for Regularized Long Short-Term Memory and Its Application to Regression and Multi-Class Classification Problems," *Engineering Applications of Artificial Intelligence*, vol. 133, 2024.
- [16] Kai-Ming He, Xiang-Yu Zhang; Shao-Qing Ren; Jian Sun, "Deep Residual Learning for Image Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 27–30 June, 2016, Las Vegas, NV, USA, pp770–778.
- [17] Long Wen, Xin-Yu Li, and Liang Gao, "A Transfer Convolutional Neural Network for Fault Diagnosis Based on ResNet-50," *Neural Computing and Applications*, vol. 32, no. 10, pp6111-6124, 2020.
- [18] B Kanaka Durga, V. Rajesh, "A ResNet Deep Learning-Based Facial Recognition Design for Future Multimedia Applications," *Computers and Electrical Engineering*, vol. 104, 2022.
- [19] Mahdi Jampour, Saeid Abbaasi, Malihe Javidi, "CapsNet Regularization and Its Conjugation with ResNet for Signature Identification," *Pattern Recognition*, vol. 120, 2021.
- [20] Que Yun, Dai Yi, Ji Xue, et al., "Automatic Classification of Asphalt Pavement Cracks Using a Novel Integrated Generative Adversarial Networks and Improved VGG Model," *Engineering Structures*, vol. 277, 2023.
- [21] Peng-Jie Tang, Han-Li Wang, and Sam Kwong, "G-MS2F: GoogLeNet Based Multi-Stage Feature Fusion of Deep CNN for Scene Recognition," *Neurocomputing*, vol. 225, pp188-197, 2017.
- [22] Ji-Xu Chen, and Xiao-Ming Liu, "Transfer Learning with One-Class Data," *Pattern Recognition Letters*, vol. 37, pp32-40, 2014.
- [23] Moreo Alejandro, Esuli Andrea, and Sebastiani Fabrizio, "Lost in Transduction: Transductive Transfer Learning in Text Classification," *ACM Transactions on Knowledge Discovery from Data*, vol. 16, no. 1, pp1-21, 2021.
- [24] Zhao-Hong Deng, Yi-Zhang Jiang, Hisao Ishibuchi, Kup-Sze Choi, and Shi-Tong Wang, "Enhanced Knowledge-Leverage-Based TSK Fuzzy System Modeling for Inductive Transfer Learning," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no.1, pp1-21, 2016.
- [25] Hong-Wei Yang, Hui He, Tao Li, Ya-Wen Bai, Wei-Zhe Zhang, "Multi - Metric Domain Adaptation for Unsupervised Transfer Learning," *IET Image Processing*, vol. 14, no.12, pp2780-2790, 2020.
- [26] Guan-Nan Li, Yu-Bei Wu, Cheng-Chu Yan, Xi Fang, Tao Li, et al, "An Improved Transfer Learning Strategy for Short-Term Cross-Building Energy Prediction Using Data Incremental." *Building Simulation*, vol. 17, no.1, pp165-183, 2024.
- [27] Sicilia Anthony, Xing-Chen Zhao, and Hwang Seong Jae, "Domain Adversarial Neural Networks for Domain Generalization: When It Works and How to Improve," *Machine Learning*, vol. 112, no.7, pp2685-2721, 2023.
- [28] Zhen-Xi Sun, Jin Zhang, Zi-Yi Chen, Lu Hong, Rui Zhang, Wei-Shi Li, et al, "Image Super-Resolution Reconstruction Using Swin Transformer with Efficient Channel Attention Networks," *Engineering Applications of Artificial Intelligence*, vol. 136, 2024.
- [29] Jin-Zheng Guang, and Zheng-Hao Xi, "ECAENet: Efficient Net with Efficient Channel Attention for Plant Species Recognition," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no.4, pp4023-4035, 2022.
- [30] Shankar M. Gowri, Babu C. Ganesh, and Rajaguru Harikumar, "Classification of Cardiac Diseases from ECG Signals Through Bio Inspired Classifiers with Adam and R-Adam Approaches for Hyperparameters Updation," *Measurement*, vol. 194, pp111048, 2022.
- [31] Daniela Lopez-Betancur, Efrén González-Ramírez, Carlos Guerrero-Mendez, et al., "Evaluation of Optimization Algorithms for Measurement of Suspended Solids," *Water*, vol. 16, no.13, 2024.
- [32] Chang-Yong Yu, Xin Qi, Hai-Tao Ma, et al., "LLR: Learning Rates by LSTM for Training Neural Networks," *Neurocomputing*, vol. 394, pp41-50, 2020.
- [33] Zhao Tong, Yin-tao Wei, "A Road Surface Image Dataset with Detailed Annotations for Driving Assistance Applications," *Data in brief*, vol. 43, 2022.
- [34] Mikolaj Wojciuk, Zaneta Swiderska-Chadaj, Krzysztof Siwek, Arkadiusz Gertych., "Improving Classification Accuracy of Fine-Tuned CNN Models: Impact of Hyperparameter Optimization," *Heliyon*, vol. 10, no.5, 2024.
- [35] Xin-Wei Wang, Shi-Zhen Su, Jun-Yao Xu, et al., "Automatic Identification of Seismic Faults Via Integrating Residual Network-50 Residual Blocks and Convolutional Block Attention Modules," *Applied Geophysics*, vol. 20, no.1, pp20-35, 2023.
- [36] Gaurav. Prasad, Ankur. Pandey and Sudhir. Kumar, "Domain Adaptation for Localization Using Combined Autoencoder and Gradient Reversal Layer in Dynamic IoT Environment," in *IEEE Transactions on Network Science and Engineering*, vol. 11, no. 1, pp685-695, 2024.
- [37] Kai Hu, Zhen-Zhen Zhang, Xiao-Rui Niu, et al., "Retinal Vessel Segmentation of Color Fundus Images Using Multiscale Convolutional Neural Network with an Improved Cross-Entropy Loss Function," *Neurocomputing*, vol. 309, pp179-191, 2018.
- [38] Mei-Ling Liu, Xiang-Nan Liu, Ling Wu, et al., "Hybrid Spatiotemporal Graph Convolutional Network for Detecting Landscape Pattern Evolution from Long-term Remote Sensing Images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp1-16, 2022.
- [39] Yuan-Bin Wang, Yu Duan, Yuan-Yuan Li, Hua-Ying Wu, "Smoke Recognition based on Dictionary and BP Neural Network," *Engineering Letters*, vol. 31, no.2, pp554-561, 2023.
- [40] Mai Hau T., Lee Seunghye, Kang Joowon, Lee Jaehong, "A Damage-Informed Neural Network Framework for Structural Damage Identification," *Computers & Structures*, vol. 292, 2024.

**Lixia NIU:** Lixia NIU was born in LvLiang, Shanxi Province, China in 1983. Professor, Ph.D. supervisor. In 2013, she graduated from Liaoning Technical University with a doctorate degree in management science and engineering. From 2013 to 2016, she was engaged in postdoctoral research in business administration at Beijing Jiaotong University. In 2019, she was a visiting scholar at the Institute of Advanced Analytics in Sydney, Australia. Her research interests are system modeling and optimization.