



ARTICLE

Cost-Sensitive Dual-Stream Residual Networks for Imbalanced Classification

Congcong Ma^{1,2}, Jiaqi Mi¹, Wanlin Gao^{1,2} and Sha Tao^{1,2,*}

¹College of Information and Electrical Engineering, China Agricultural University, Beijing, 100083, China

²Key Laboratory of Agricultural Informatization Standardization, Ministry of Agriculture and Rural Affairs, China Agricultural University, Beijing, 100083, China

*Corresponding Author: Sha Tao. Email: taosha20070608@163.com

Received: 30 May 2024 Accepted: 07 August 2024 Published: 12 September 2024

ABSTRACT

Imbalanced data classification is the task of classifying datasets where there is a significant disparity in the number of samples between different classes. This task is prevalent in practical scenarios such as industrial fault diagnosis, network intrusion detection, cancer detection, etc. In imbalanced classification tasks, the focus is typically on achieving high recognition accuracy for the minority class. However, due to the challenges presented by imbalanced multi-class datasets, such as the scarcity of samples in minority classes and complex inter-class relationships with overlapping boundaries, existing methods often do not perform well in multi-class imbalanced data classification tasks, particularly in terms of recognizing minority classes with high accuracy. Therefore, this paper proposes a multi-class imbalanced data classification method called CSDSResNet, which is based on a cost-sensitive dual-stream residual network. Firstly, to address the issue of limited samples in the minority class within imbalanced datasets, a dual-stream residual network backbone structure is designed to enhance the model's feature extraction capability. Next, considering the complexities arising from imbalanced inter-class sample quantities and imbalanced inter-class overlapping boundaries in multi-class imbalanced datasets, a unique cost-sensitive loss function is devised. This loss function places more emphasis on the minority class and the challenging classes with high inter-class similarity, thereby improving the model's classification ability. Finally, the effectiveness and generalization of the proposed method, CSDSResNet, are evaluated on two datasets: 'DryBeans' and 'Electric Motor Defects'. The experimental results demonstrate that CSDSResNet achieves the best performance on imbalanced datasets, with macro_F1-score values improving by 2.9% and 1.9% on the two datasets compared to current state-of-the-art classification methods, respectively. Furthermore, it achieves the highest precision in single-class recognition tasks for the minority class.

KEYWORDS

Deep learning; imbalanced data classification; fault diagnosis; cost-sensitivity

1 Introduction

Classification is a critical task in data mining with substantial research significance. In practical applications, class imbalance is a common issue, prevalent in domains such as disease detection, intrusion detection, and industrial fault diagnosis. These scenarios often emphasize identifying



minority samples, such as faults and anomalies, amidst a vast majority of normal data samples. The substantial differences in sample sizes and the complexity of data distributions make accurately identifying minority class samples particularly challenging.

Currently, extensive research is focused on multi-class imbalanced data classification. Solutions typically fall into two categories: data-level methods, such as resampling and feature selection, and algorithm-level methods, including multi-class decomposition techniques, ensemble learning, and cost-sensitive learning. Data-resampling methods involve either undersampling majority class samples or oversampling minority class samples before training a classifier. Hybrid resampling methods can also be used to balance class distributions. For instance, Li et al. [1] proposed an entropy-based undersampling method that balances datasets using a new class imbalance metric called entropy-based imbalance degree (EID), which mitigates information loss associated with basic undersampling methods [2]. Similarly, Li et al. [3] introduced an undersampling method based on minority class neighborhood distance, addressing data imbalance in sentiment classification by removing majority text within minority text neighborhoods. Features selection, on the other hand, involves filtering out redundant features while retaining relevant ones to improve classifier performance. Fu et al. [4], for example, proposed a feature selection algorithm based on Hellinger distance, suitable for high-dimensional imbalanced classification. Maldonado et al. [5] explored embedded feature selection methods in support vector machine classification. While undersampling and feature selection methods can enhance classification performance, they often involve removing data samples or features, which can lead to varying degrees of information loss.

Oversampling techniques, on the other hand, involve creating new samples for the minority class to balance the dataset. Notable methods include the Synthetic Minority Over-sampling Technique (SMOTE) and Generative Adversarial Network (GAN)-based approaches for generating minority class samples. Asniar et al. [6] proposed SMOTE-LOF method to synthesize minority data by the Synthetic Minority Oversampling Technique (SMOTE) which has added local outliers' factors and achieved a better effect on dataset with a large number of data examples and a smaller imbalance ratio. Joloudari et al. [7] proposed an effective method for class-imbalanced learning based on Synthetic Minority Over-sampling Technique and Convolutional Neural Networks (SMOTE-CNN), which combines the SMOTE oversampling technique with a CNN classifier to effectively address imbalanced data. Lee et al. [8] proposed an intrusion detection system based on generative adversarial networks, which uses GAN networks to oversample minority class samples and improve the accuracy of classification models on imbalanced datasets. However, oversampling methods inevitably introduce noise while balancing the dataset, and may even lead to severe overfitting issues.

Algorithm-level methods focus on developing new algorithms or improving existing ones, such as Logistic Regression (LR), Support Vector Machines (SVM), and ResNet [9], to enhance performance in imbalanced multiclass problems. These methods do not alter the dataset by adding or removing samples, thereby maintaining the original data distribution, making them well-suited for complexly distributed imbalanced classification challenges. Multiclass decomposition techniques, on the other hand, employ divide-and-conquer strategies to break down multiclass problems into simpler binary subproblems, thereby simplifying the overall problem. For example, Gao et al. [10] proposed a differential partition sampling ensemble method (DPSE), which splits a multiclass dataset into multiple binary datasets using the One-vs.-All (OVA) strategy for model training, achieving good performance in imbalanced learning. Mohammed et al. [11] combined the OVA decomposition strategy with ensemble learning, introducing an adaptive window adjustment method based on the imbalance ratio to reduce uncertainty during imbalanced data stream learning. Such methods can reduce the burden on each classifier and exhibit better robustness to imbalanced data. However, multiclass decomposition

techniques may not capture more complex relationships between multiple classes, potentially reducing the recognition rate for certain classes.

Ensemble learning involves combining multiple weak base classifiers with modest classification abilities to create a strong classifier with improved performance. For example, AdaBoost [12] typically uses decision trees as its base classifier. AdaBoost is an adaptive ensemble algorithm, where each new model is adjusted based on the performance of the previous model, leading to highly accurate classification results. XGBoost [13] is an optimization algorithm based on gradient boosted decision tree, adding a new decision tree in each iteration to fit the residual between the prediction result of the previous decision tree and the true value, and the final integrated classifier has high classification accuracy. RUSBoost [2] first utilizes resampling techniques to process imbalanced datasets into balanced datasets, and then utilizes ensemble methods to classify them. Random Forest is a highly flexible machine learning algorithm that integrates multiple decision trees to learn and classify input samples. The decision trees that make up the random forest have no association with one another. This bagging-based approach increases prediction accuracy and is resistant to data that is not balanced. Ensemble learning often requires more time and computational resources because multiple classifiers need to be trained. In some cases, these techniques can help balance class distributions and improve the performance of classifiers affected by class imbalance. However, they may struggle to achieve high classification accuracy when there are several minority groups in the imbalanced data and the boundaries between classes overlap.

The above-described data imbalance is not an isolated phenomenon; it also occurs in a number of industries and agricultural sectors, for example, in the identification of similar motor defects and close-source biological classification. In such circumstances, it is essential to recognize minority classes effectively and accurately. However, the four previously mentioned methods may not meet these demands, as they lack targeted improvement measures for these specific cases. Cost-sensitive learning addresses this by assigning different costs to the misclassification of various classes, thereby enhancing the performance of general classifiers. This is typically achieved by improving the loss function in the baseline model. Specifically, a cost-sensitive loss function sets a greater misclassification cost for the positive class compared to the negative class, making the model more attentive to the accurate classification of minority classes. Existing research on improving loss functions can be broadly categorized into three types: weighted cross-entropy loss, weighted support vector machine loss, and improved hybrid loss functions. Weighted cross-entropy loss assigns different weights to the cross-entropy loss terms of different classes to reduce the dominance of the majority class in the loss function, thereby improving the classification accuracy of minority classes. For example, Focal Loss [14] reduces the weight of easily classified samples and increases the weight of hard-to-classify samples (usually minority classes) to diminish the influence of easily classified samples on the loss function. Class-Balanced Loss [15] adjusts the contribution of each class to the loss function based on the number of samples in each class, giving higher weight to samples from minority classes. Cost-Sensitive Convolutional Neural Network (CSCNN) [16] is an improved algorithm based on Convolutional Neural Networks (CNNs) that incorporates weighted cross-entropy loss to differentiate the misclassification costs of imbalanced classes. In the task of imbalanced encrypted traffic classification, CSCNN outperformed general machine learning and deep learning classification algorithms, achieving an F1-score exceeding 96%. ECG-CNN [17] utilizes a cost-sensitive CNN model to address the imbalanced data issue in ECG rhythm detection, achieving higher classification performance. Cost-Sensitive Residual Network (CS-ResNet) [18] is an enhanced algorithm based on Residual Neural Networks (ResNet). It strengthens the standard ResNet by incorporating weighted cross-entropy loss. Specifically, it sets cost-sensitive factors according to the degree of imbalance between classes, assigning larger weights to classes with

fewer samples. By combining ResNet's excellent feature extraction capabilities with the balancing ability of the cost-sensitive layer, CS-ResNet ultimately improves the accuracy of printed circuit board defect detection, achieving a maximum sensitivity of 0.89. Weighted support vector machine loss enhances the handling of class imbalance by assigning different weights to the support vector machine loss function terms. For example, Cost-Sensitive Support Vector Machine (CS-SVM) [19] adjusts the SVM loss function according to a cost matrix, focusing more on the classification of minority classes. CS-SVM has demonstrated superior classification accuracy across more than ten datasets, indicating good generalization performance in experiments. FCSSVM (fine cost-sensitive support vector machine classifier) [20] addresses imbalanced data classification by using a fined cost-sensitive support vector machine, which improves classification performance by finely tuning class cost weights. Improved hybrid loss functions combine different types of loss terms and apply weighting based on class imbalance to enhance model performance in imbalanced data classification tasks. For instance, Dice Loss [21] combines cross-entropy loss and Dice coefficient loss, balancing their proportions to improve model performance in handling minority classes. Cost-sensitive learning offers advantages over other methods by directly addressing the issue of imbalanced data. It prioritizes minority classes, thereby improving their classification accuracy while reducing unnecessary computational costs. This makes cost-sensitive learning a potential approach for accurately classifying imbalanced data with multiple minority classes and overlapping class boundaries.

Existing cost-sensitive learning methods have focused on addressing the issue of imbalanced class sizes, which has improved the recognition accuracy of minority classes to some extent. However, they still struggle with overlapping imbalanced class distributions. Therefore, this paper introduces an imbalanced data classification method named CSDSResNet, based on a cost-sensitive dual-stream residual neural network. It utilizes an optimized residual network as the base classifier and designs a dual-stream backbone network structure to enhance the classifier's feature extraction capability. Additionally, it incorporates cost-sensitive functions that account for class imbalance and inter-class similarity, directing the model's attention to minority classes and highly similar inter-class categories. This approach aims to achieve accurate classification for imbalanced data scenarios involving multiple minority classes with imbalanced overlapping inter-class sample distributions, providing a solution for real-world problems characterized by such imbalanced data.

The main contributions and innovations of this paper are as follows:

- Proposed a classifier based on a dual-stream residual network model. Designed the dual-stream residual backbone structure to provide the model with different scales of receptive fields, enhancing its feature extraction capabilities. This allows the classifier to quickly learn from multi-class imbalanced data.
- Designed a cost-sensitive function based on inter-class sample imbalance and inter-class similarity. This function directs the model's attention towards minority classes and the challenging task of distinguishing highly similar classes. This approach aims to address the complex classification issues arising from imbalanced sample sizes and imbalanced sample distribution overlap in multi-class imbalanced data.
- Addressed practical tasks in multi-class imbalanced data classification, specifically focusing on dry beans classification and motor defect detection: evaluated the effectiveness of the proposed CSDSResNet method on the "DryBeans" and "Sensorless_drive_diagnosis" datasets. These datasets exhibit characteristics of imbalanced sample sizes and imbalanced inter-class distribution overlaps due to the presence of biologically closely related bean categories in "DryBeans" and multiple similar yet distinct defects in "Sensorless_drive_diagnosis".

The remaining parts of the paper are organized as follows: [Section 2](#) demonstrates the design of the proposed method CSDSResNet in detail. Experimental setting is stated in [Section 3](#). In [Section 4](#), we introduce the detailed experimental results. Finally, [Section 5](#) provides conclusions and future works.

2 Methods

To address the issue of imbalanced inter-class sample quantities and overlapping sample distributions in multi-class imbalanced datasets, this paper proposes an improved residual network model called CSDSResNet. This is an improved algorithm based on the residual network concept, taking into account the excellent feature extraction capability of residual networks. Therefore, residual units are selected as the fundamental building blocks of the model. The overall structure of CSDSResNet is shown in [Fig. 1](#).

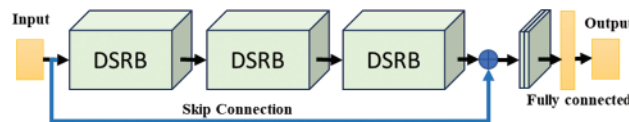


Figure 1: Overview of the CSDSResNet

2.1 Overall Structure of CSDSResNet

The CSDSResNet model is designed with a dual-stream backbone network to acquire different receptive fields, thereby further optimizing the model's fundamental feature extraction capability. It couples the cost-sensitive factor with the model's loss function, enabling the model to gain a special focus on minority classes during the backward propagation learning process. The overall structure of the model is illustrated in [Fig. 1](#). The backbone network consists of stacked dual-stream residual blocks, skip connections, two convolutional layers, and one fully connected layer. More details about the main innovations of the proposed method will be elaborated in [Sections 2.2](#) and [2.3](#), respectively.

2.2 The Dual-Stream Residual Block

To enhance the model's feature extraction capability, we designed a backbone network composed of stacked dual-stream residual blocks, as shown in [Fig. 2](#). The introduction of residual structures reduces information loss in the lower layers of the network compared to the upper layers during the convolution process, and residual connections can prevent network degradation [9].

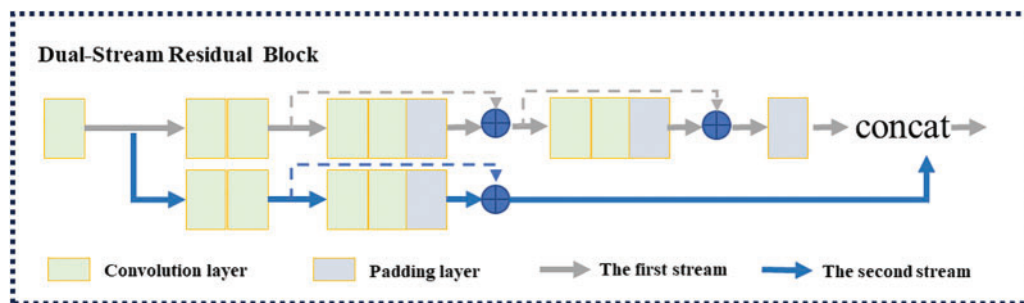


Figure 2: The structure of the dual-stream residual block

The gray line path represented in Fig. 2 is Path 1 with a feature extraction stride of 2. Due to the small number of samples in the minority class in the imbalanced dataset, the single-path network structure cannot adequately extract features. Therefore, we added an additional feature extraction path, represented by the blue Path 2 in Fig. 2, with a feature extraction stride of 3. We designed convolutional kernels with odd-even differences in size. This is because the size of convolutional kernels determines the receptive field of the model, and this design allows the model to extract data features from different dimensions. Compared to simply increasing the depth of the network to enlarge the receptive field, the dual-stream network structure is evidently more suitable for dealing with the limited sample size of minority classes in multi-class imbalanced datasets.

2.3 Cost-Sensitive Loss Function

The cross-entropy loss function is frequently utilized with the softmax classifier. The latter calculates cross-entropy using the one-hot form of the real category to derive the associated loss, which is suited for multi-classification learning. Cross entropy loss considers misclassification of different categories to be equally relevant, which can only be used to minimize misclassification of all samples. However, because of the tiny proportion of minority samples in imbalanced data, it has little influence on the overall sample classification accuracy, and the recognition rate of the majority of samples is crucial in the overall sample recognition rate. As a result, the misclassification of minority class samples is ignored to some extent, causing the classifier to tend to improve the accurate recognition of the majority class, resulting in minority class misclassification.

There are two main reasons for the difficulty of classifying imbalanced datasets: on the one hand, the sample number of the minority class is small, which differs greatly from the majority classes. The model can obtain considerable performance by biasing the classification results to the majority classes; On the other hand, the high-dimensional features of the inter-class samples have overlapping parts, the imbalance in the overlap of inter-class distributions increases the classification difficulty. While this kind of imbalance also exists in balanced data distributions, having an abundance of training data allows the model to eventually learn classifier-friendly features and recognize and classify them correctly. In the case of imbalanced multi-class tasks, due to the lack of training data for minority classes, it's necessary to strengthen the learning capability of the classifier. Therefore, we design cost-sensitive functions to help the classifier quickly learn more discriminative and classifier-friendly features in imbalanced classification tasks where minority class sample distribution is uneven. Considering the above two points, we will define inter-class imbalance cost-sensitive factor CS_I and inter-class similarity cost-sensitive factor CS_D to optimize the loss function, which affects the optimization of model parameters in the backward pass of model training. The new loss function is defined as follows:

$$Loss = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C (y_{i,c} \log \hat{y}_{i,c} \cdot CS_I(\tau, c) \cdot CS_D(\tau, c)) \quad (1)$$

In this study, we enhance the conventional multi-class cross-entropy calculation by integrating the inter-class imbalance cost-sensitive matrix CS_I and the inter-class similarity cost-sensitive matrix CS_D into the classification error computation. Specifically, N denotes the total number of samples in the dataset, C represents the number of classes, $y_{i,c}$ is the true label (0 or 1) of sample i for class c , and $\hat{y}_{i,c}$ is the predicted probability of the model assigning sample i to class c . τ represents the true class of sample i . From the Eq. (1), it is evident that the values of the inter-class imbalance cost-sensitive factor $CS_I(\tau, c)$ and the inter-class similarity cost-sensitive factor $CS_D(\tau, c)$ depend solely on the true class τ of sample i and the predicted class c . These values are precomputed before the model training

begins. The matrix CS_I has dimensions $C \times C$, with the sensitivity factor in the τ -th row and c -th column being:

$$CS_I(\tau, c) = \sqrt{\frac{Num_\tau}{Num_c}} \quad (2)$$

Num_τ represents the total number of samples in class τ , while Num_c represents the total number of samples in class c . Thus, Num_c and Num_τ are independent of each other and also independent of other classes. The benefit of taking the square root of the ratio of sample quantities is that it reduces the magnitude of factor variation, making the final loss value changes smoother, which aids in the stable training of the model. The inter-class similarity cost-sensitive factor CS_D also has dimensions of $C \times C$, with the sensitivity factor in the τ -th row and c -th column being:

$$CS_D(\tau, c) = (1 + \exp(\varphi(\tau, c)))^{-1} + 1 \quad (3)$$

$$\varphi_{ij} = \sqrt{\sum_{k=1}^K (\chi_{\tau,k} - \chi_{c,k})^2} \quad (4)$$

Here, K denotes the number of features of the samples, and $\chi_{\tau,k}$ represents the k -th feature value of the central feature vector within the 95% confidence space of class τ . It is important to note that before calculating the central feature vector, the sample set has been standardized. The central feature vector for a class is obtained by averaging the corresponding dimensions of all samples in that class. $\varphi(\tau, c)$ represents the Euclidean distance between the central vectors of classes τ and c . When the distributions of the two classes are close, φ approaches 0, and the cost-sensitive factor $CS_D(\tau, c)$ approaches 1.5. Conversely, when the distributions of the two classes are significantly different, $CS_D(\tau, c)$ approaches 1. Therefore, this sensitivity factor effectively increases the penalty for misclassification between classes with similar distributions.

The following pseudocode details the computation process for the inter-class similarity cost-sensitive factor (Algorithm 1):

Algorithm 1: Interclass Similarity Cost-Sensitive Algorithm

Input: Initialize original training data X , total number of categories C , The number of samples for each category is Num . Initialize training feature dimension K and Confidence threshold T , default value $T = 1.96$.

Output: the matrix of Interclass Similarity Cost-Sensitive CS_D .

1. Standardize the training data X , get X' ;
// Step 1. Remove samples outside 95% confidence interval for each class.
 2. **for** c in $\{1, 2, \dots, C\}$ **do**
 3. X'_c represents the sample set with category c ;
 4. X'_c has Num_c samples and forms a sample sequence $(x_{c1}, x_{c2}, \dots, x_{cNum_c})$, Each vector is represented as: $x_{ci} = (x_{ci1}, x_{ci2}, \dots, x_{ciK})$;
 5. $\mu_{cj} = \frac{1}{Num_c} \sum_{i=1}^{Num_c} x_{cij}$ // Calculate the mean for each dimension.
 6. $\sigma_{cj} = \frac{1}{Num_c} \sum_{i=1}^{Num_c} (x_{cij} - \mu_{cj})^2$
// Calculate the total z-score for each sample.
 7. **for** i in $\{1, 2, \dots, Num_c\}$ **do**
-

(Continued)

Algorithm 1 (continued)

-
8. $z_{ci} = \sqrt{\sum_{j=1}^K \left(\frac{(x_{cij} - \mu_{cj})}{\sigma_{cj}} \right)^2}$
 9. **if** $z_{ci} \leq T$ **then**
 10. drop off sample x_{ci} ;
 11. **end if**
 12. Update category sample size from Num_c to Num'_c ;
 13. $\chi_c = \left(\frac{1}{Num'_c} \sum_{t=1}^{Num'_c} x_{ct1}, \frac{1}{Num'_c} \sum_{t=1}^{Num'_c} x_{ct2}, \dots, \frac{1}{Num'_c} \sum_{t=1}^{Num'_c} x_{ctK} \right)$
 // Step 2. Calculate Cost-Sensitive value between two categories.
 14. **for** i in $\{1, 2, \dots, C\}$ **do**
 15. **for** j in $\{1, 2, \dots, C\}$ **do**
 16. $\varphi(i, j) = \sqrt{\sum_{k=1}^K (\chi_{ik} - \chi_{jk})^2}$
 17. $CS_D(i, j) = \frac{1}{1 + e^{\varphi(i, j)}} + 1$
-

3 Experiment Setting**3.1 Experimental Environment**

Our experiments were conducted on a computer with an i7-12700K processor and 32 GB memory, running Windows 10. The system is equipped with NVIDIA GeForce RTX 3060Ti GPUs featuring 8 GB memory, a 14 GHz memory clock, and a 256-bit width. The model was implemented using TensorFlow 2.0, with PyCharm as the IDE. Key toolkits included numpy, random, glob, imageio, math, time, and os, with Python 3.7 as the primary programming language.

3.2 Datasets

We conducted our research on the “DryBean” [22] and “Sensorless_drive_diagnosis” datasets. In imbalanced classification tasks, the category with a large number of samples is typically referred to as the negative class, while the category with a small number of samples is known as the positive class. The class imbalance ratio (IR) is defined as the ratio of the number of negative class samples to the number of positive class samples. Generally, the larger the IR, the more challenging the classification task becomes. t-SNE [23] is a non-linear dimensionality reduction algorithm particularly suited for reducing high-dimensional data to 2D or 3D, while preserving the similarity in the joint probability distribution between the low-dimensional and original data.

The DryBean dataset, includes size and shape features of seven different dry beans, with 16 attributes like Area and Perimeter. The t-SNE visualization (Fig. 3a) shows distinct class distributions and varying degrees of overlap, especially among the minority classes Bombay, Sira, and Horoz. To validate the model’s generalization, we used the Sensorless_drive_diagnosis dataset, containing 11 classes with 48 attributes each, and displayed t-SNE results for Classes 6 to 10 (Fig. 3b). Both datasets were randomly sampled to create different class imbalance ratios. Specifically, we employed stratified sampling, where samples were randomly drawn from each class according to the imbalance ratio (IR). The sampled datasets are described in Tables 1 and 2. In our experiments, we also divided

the datasets into training and test sets using an 8:2 ratio within each class, ensuring that the IRs remained consistent.

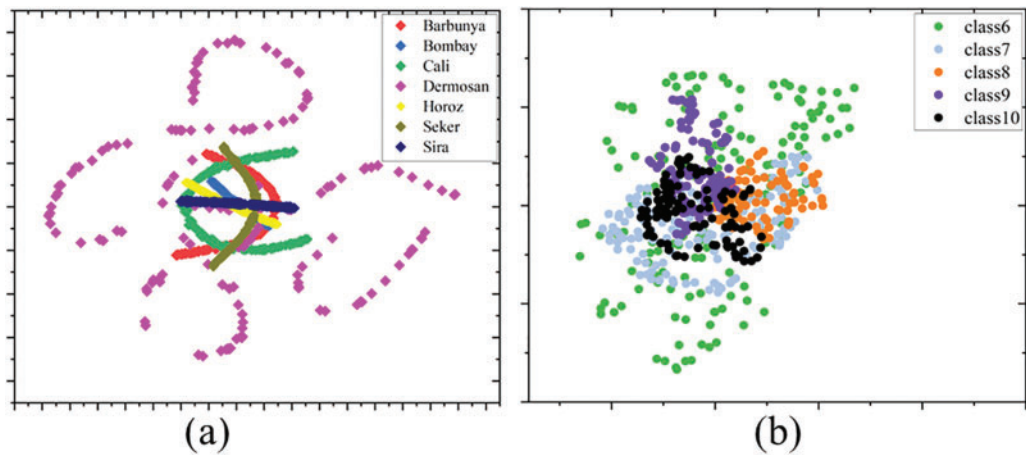


Figure 3: T-SNE visualization results

Table 1: Description of DryBean dataset

Category	Dried bean species	Number of attributes	Sample size	IR
1	Derosan	16	3536	1
2	SEKER	16	304	11.63
3	Sira	16	211	16.76
4	Horoz	16	193	18.32
5	Cali	16	489	7.23
6	Barbunya	16	397	8.91
7	Bombay	16	157	22.52

Table 2: Description of Sensorless_drive_diagnosis dataset

Class	Number of attributes	Sample size	IR
0	48	2660	1
1	48	2530	1.05
2	48	2340	1.14
3	48	2050	1.30
4	48	2100	1.27
5	48	2370	1.12
6	48	2500	1.06
7	48	1410	1.89
8	48	850	3.13

(Continued)

Table 2 (continued)

Class	Number of attributes	Sample size	IR
9	48	930	2.86
10	48	1010	2.63

3.3 Evaluation Metric

The performance of the proposed method is evaluated using accuracy, precision, recall, and F1-score. The macro average of these metrics effectively assesses classifier performance in multi-class tasks, emphasizing the importance of accurately classifying minority classes.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

$$F_1 - score = \frac{2TP}{2TP + FN + FP} \quad (8)$$

$$macro_S = \frac{1}{n} \sum_{i=1}^n S_i \quad (9)$$

TP, FP, TN and FN represent the samples belong to True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) of the category, respectively.

3.4 Parameter Configuration

We compare the performance of the model at a learning rate of 0.001, 0.005, and 0.01, respectively. The experimental results are shown in Fig. 4. The subfigure (a–c) shows that when the learning rate is 0.01, the loss-training epoch curve declines the fastest, reaching roughly 1.3 at 10 epoch and 1.2 at 60 epoch before becoming stable. When the learning rate is 0.005, it falls to around 1.3 at 20 epoch, reaches approximately 1.2 at 60 epoch, and then remains stable. When the learning rate is 0.001, the curve decreases the slowest, and gradually be stable at 120 epochs. The ultimate accuracy of the model is at its maximum when the learning rate is 0.005, as can be seen in subfigure (d). We finally chose a learning rate of 0.005, because it has the highest training accuracy and is more stable.

In order to improve training efficiency, we set a comparison experiment as shown in Fig. 5 to determine the most appropriate mini batch size. According to subfigure (a–c), it can be shown that the loss curve falls below 1.3 at 20 epochs when the mini batch size is 128 and 64. The fastest and most stable model convergence occurs when the mini batch size is 128. The subfigure (d) shows that the model training accuracy increases the fastest when the mini batch size is 128 and that the ultimate accuracy attained by the three is essentially the same. Therefore, we set the mini batch size to 128.

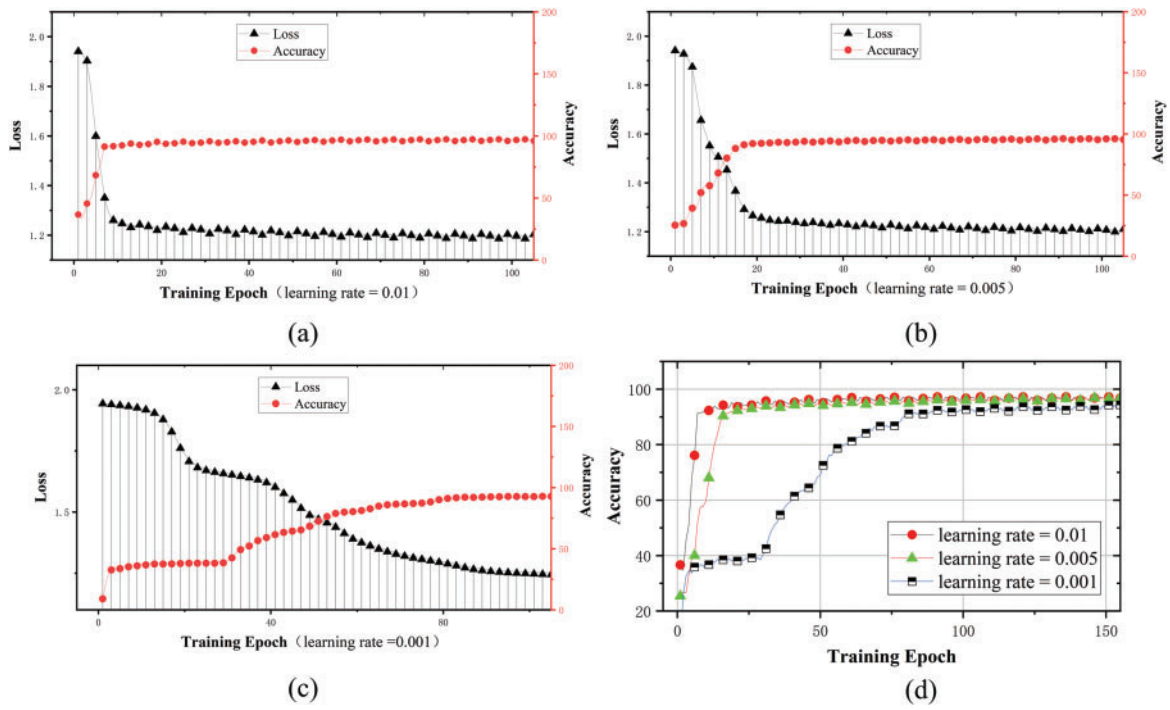


Figure 4: Model training performance at different learning rates

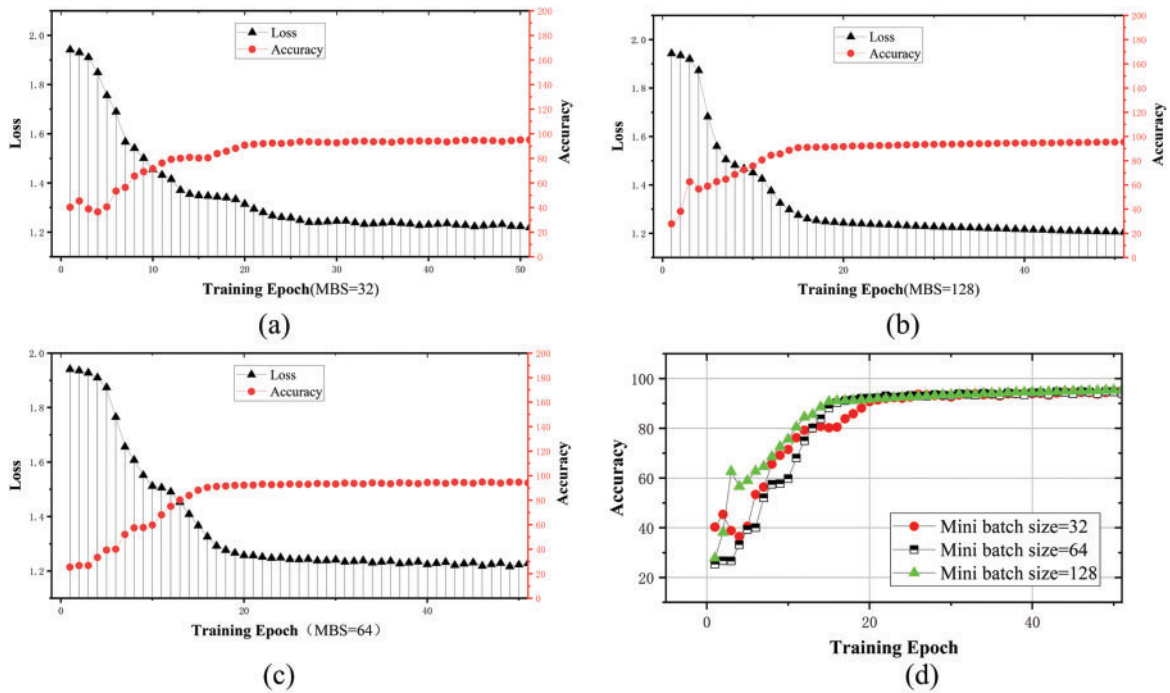


Figure 5: Model training performance at mini batch sizes

4 Results and Discussion

In this section, a significant variety of experiments are carried out to completely demonstrate the effectiveness of the proposed method for imbalanced classification tasks.

4.1 Ablation Study

Ablation experiments involve removing certain enhancement elements from the final model to assess their necessity. In this section, we evaluate the effectiveness of the proposed cost-sensitive loss function and dual-stream residual block for imbalanced classification.

To further validate the effectiveness of the cost-sensitive factors CS_I and CS_D in addressing class imbalance, we designed an ablation experiment. The results, shown in Fig. 6, compared with the methods without cost-sensitive factors CS_I and CS_D , the accuracy of CSDSResNet is improved by 1.3% and 1.3%, respectively. Macro_P is improved by 5% and 4.4%, respectively. Macro_F1 improved by 3.3% and 1.8%, respectively, indicate that the inclusion of these cost-sensitive factors enhances the classifier's attention to and accuracy in identifying minority classes. The introduction of the cost-sensitive factor resulted in a slight decrease in Macro_Recall, which we attribute to the classifier becoming more conservative. It tends to lean towards predicting negatives to avoid false positives. This inclination enhances precision and accuracy but sacrifices recall. This observation aligns with our experimental findings.

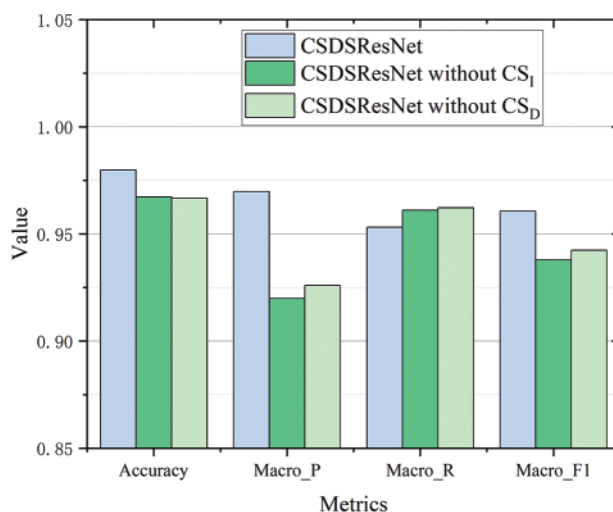


Figure 6: Ablation study results of cost-sensitive loss functions

Table 3 presents the ablation study results of dual-stream residual blocks. We retained only a single path as the model's main backbone for feature extraction, effectively degrading the model to a standard residual neural network while keeping all other components unchanged. The best results are highlighted in bold. It can be observed that the dual-stream structure achieves the best performance, with the model's Accuracy and Macro_F1-score improving by 1.39% and 1.89%, respectively, compared to the model with only Path 1. Similarly, compared to the model with only Path 2, the Accuracy and Macro_F1-score improved by 2.02% and 2.94%, respectively. The ablation study results demonstrate that dual-stream residual blocks effectively extract and fuse high-dimensional features, enhancing the model's learning capability.

Table 3: Ablation study results of dual-stream residual blocks

Path 1	Path 2	Accuracy	Macro_precision	Macro_Recall	Macro_F1-score
✓		0.9660	0.9242	0.9527	0.9417
	✓	0.9597	0.9098	0.9507	0.9312
✓	✓	0.9799	0.9697	0.9531	0.9606

4.2 Model Comparison

This subsection compares the performance of our proposed method, CSDSResNet, with several classical methods for imbalanced classification, including CSCNN, XGBoost, SVM, RUSBoost, Random Forest, Logistic Regression, SMOTE-CNN, FCSSCM, ECG-CNN and AdaBoost. Table 4 presents the performance metrics of various methods on the “DryBean” dataset. Our proposed method, CSDSResNet, outperforms all other methods across accuracy, macro precision, and macro_F1-score. For example, compared to the second-best method, CSDSResNet shows a 1.64% improvement in accuracy, a 2.48% increase in macro precision, and a 2.72% enhancement in macro F1-score. To understand which class recognition improvements most impact CSDSResNet’s overall performance, we visualized the recognition performance for each class, as shown in Fig. 7.

Table 4: Results of several methods on “DryBean”

Methods	Accuracy	Macro_precision	Macro_Recall	Macro_F1-score
CSCNN	0.9242	0.9356	0.9287	0.9321
XGBoost	<u>0.9635</u>	0.9306	0.9125	0.9203
SVM	0.9598	<u>0.9449</u>	0.9061	0.9195
RUSBoost	0.8956	0.8754	0.6811	0.7073
RF	0.9623	0.9389	0.9126	0.9251
LR	0.9591	0.9250	0.8983	0.9080
SMOTE-CNN	0.9509	0.9057	0.9521	0.9238
FCSSVM	0.9415	0.8867	0.9428	0.9075
ECG-CNN	0.9610	0.9442	0.9579	<u>0.9334</u>
AdaBoost	0.6467	0.5393	0.5223	0.4777
CSDSResnet	0.9799	0.9697	<u>0.9531</u>	0.9606

CSDSResNet and ECG-CNN achieved the top two F1-scores in single-class recognition. Notably, CSDSResNet demonstrated the best single-class recognition ability for the minority classes Seker and Sira, with improvements of 6.6% and 13.7%, respectively, compared to ECG-CNN. These results indicate that the overall performance enhancement of CSDSResNet is primarily due to its superior recognition ability for minority classes, demonstrating its effectiveness in handling highly imbalanced classification tasks.

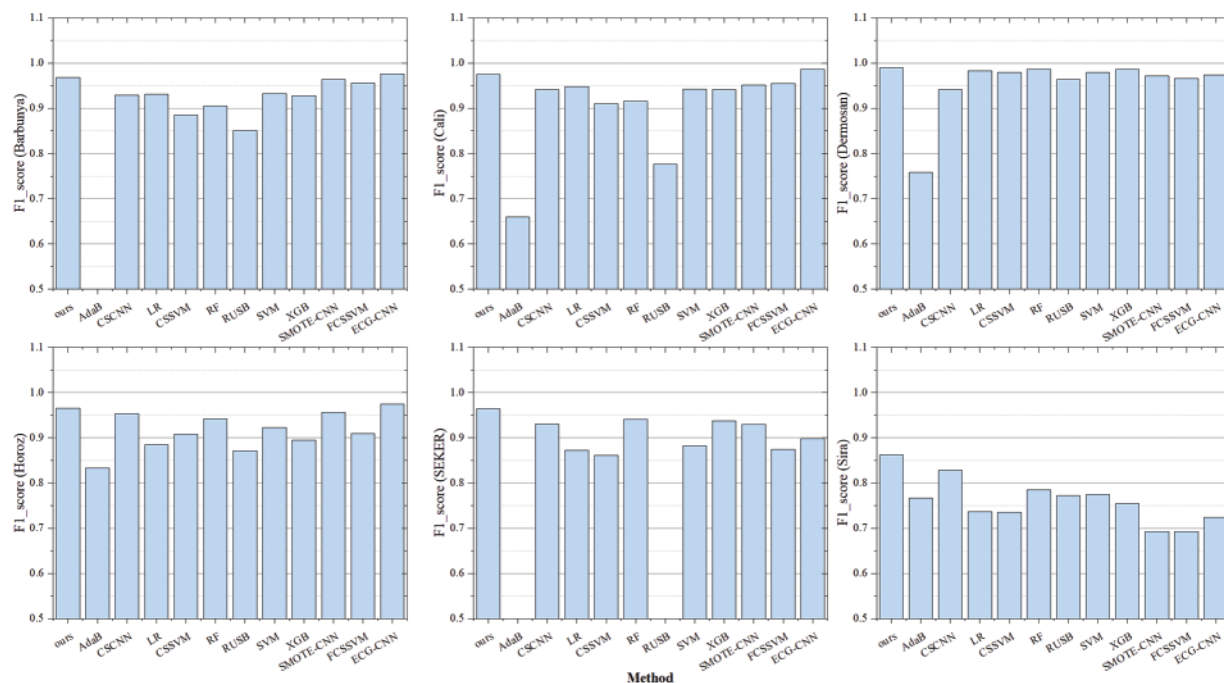


Figure 7: Single-class identification results on “DryBean”

4.3 Model Transfer

To validate the generalization performance of the proposed model, experiments were conducted on the Sensorless_drive_diagnosis dataset. Table 5 compares the performance of 11 classification algorithms. CSDSResNet outperforms all other methods across all metrics, achieving improvements of 1.4% in accuracy, 1.1% in macro precision, 0.7% in macro recall, and 1.4% in macro F1-score. ECG-CNN and XGBoost ranked second and third, respectively.

Table 5: Results of several methods on Sensorless_drive_diagnosis

Methods	Accuracy	Macro_precision	Macro_Recall	Macro_F1-score
CSCNN	0.923	0.905	0.908	0.908
XGBoost	<u>0.937</u>	0.945	0.921	0.929
SVM	0.910	0.905	0.889	0.895
RUSBoost	0.882	0.878	0.863	0.869
RF	0.936	<u>0.947</u>	0.911	0.923
LR	0.879	0.902	0.847	0.861
SMOTE-CNN	0.919	0.905	0.899	0.901
FCSSVM	0.928	0.893	0.897	0.926
ECG-CNN	0.934	0.935	<u>0.934</u>	<u>0.934</u>
AdaBoost	0.907	0.919	0.887	0.898
CSDSResNet	0.951	0.958	0.941	0.948

Due to the high number of classes in the Sensorless_drive_diagnosis dataset, single-class recognition results for several minority classes are shown in Fig. 8. The data’s high similarity among classes makes single-class recognition challenging. CSDSResNet excels in recognizing minority Classes 6, 7, 8, and 9, with a 3.7% improvement in F1-score for Class 8. CSDSResNet, along with XGBoost and ECG-CNN, ranks among the top three in single-class recognition, consistent with Table 5. These results highlight CSDSResNet’s effective classification of imbalanced data with overlapping distributions.

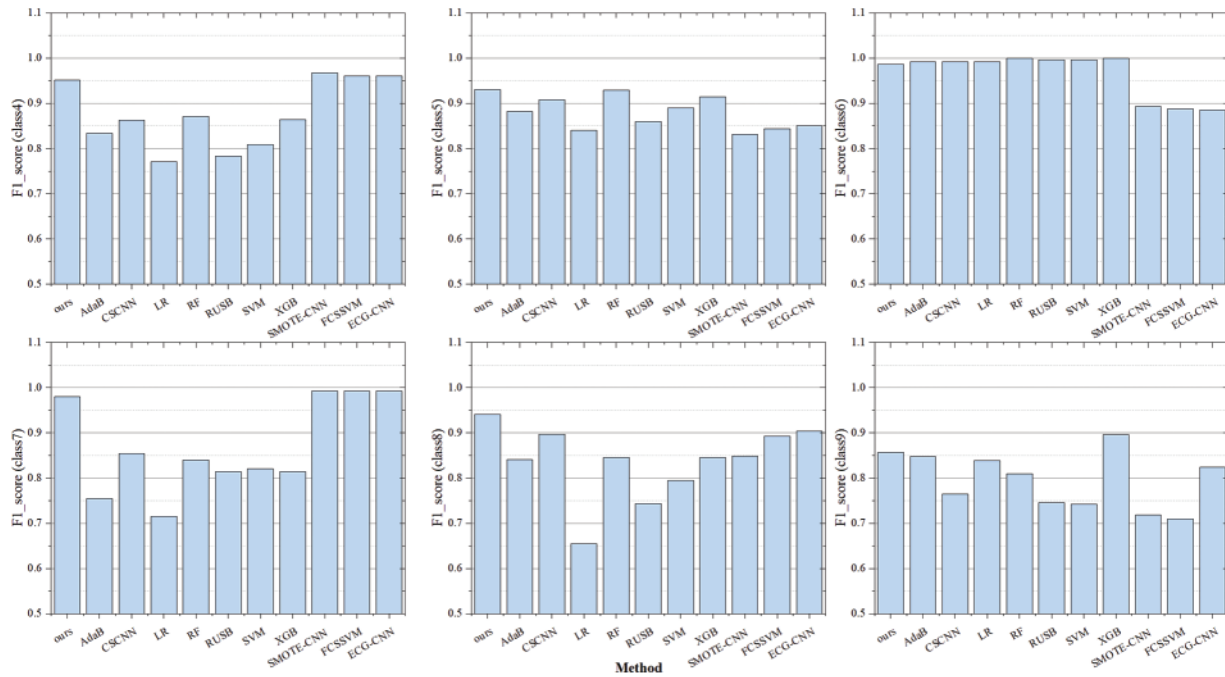


Figure 8: Single-class identification results on Sensorless_drive_diagnosis

Given that the Sensorless_drive_diagnosis dataset is relatively large, several cost-sensitive deep learning methods have shown promising results in comparison experiments. We are also curious about the performance of these methods on smaller datasets. To further evaluate the performance of the proposed CSDSResNet in imbalanced classification tasks, we performed a 10-fold random undersampling for each class of the Sensorless_drive_diagnosis dataset. Specifically, Classes 4, 5, and 6 were undersampled by a factor of 100 to further increase the imbalance ratio and classification difficulty. The description of the adjusted dataset is shown in Table 6.

Table 6: Description of the Small-Sized Sensorless_drive_diagnosis dataset

Class	Number of attributes	Sample size	IR
0	48	266	1
1	48	253	1.05
2	48	234	1.14
3	48	205	1.30
4	48	21	12.67

(Continued)

Table 6 (continued)

Class	Number of attributes	Sample size	IR
5	48	23	11.57
6	48	25	10.64
7	48	141	1.89
8	48	85	3.13
9	48	93	2.86
10	48	101	2.63

Table 7 shows the results for the Small-Sized Sensorless_drive_diagnosis dataset. While reducing the training set size decreased performance across all methods, CSDSResNet still performed the best. It improved accuracy by 1.9%, macro precision by 4%, macro recall by 6.2%, and macro F1-score by 5.2% compared to ECG-CNN. These results highlight that CSDSResNet excels in handling highly imbalanced tasks with limited minority class samples and overlapping distributions, demonstrating its robustness in challenging classification scenarios.

Table 7: Results of several methods on the Small-Sized Sensorless_drive_diagnosis

Methods	Accuracy	Macro_precision	Macro_Recall	Macro_F1-score
CSCNN	0.903	<u>0.870</u>	0.786	0.803
XGBoost	0.926	0.759	0.794	0.791
SVM	0.899	0.773	0.753	0.759
RUSBoost	0.812	0.703	0.675	0.667
RF	<u>0.954</u>	0.781	0.795	0.787
LR	0.839	0.713	0.718	0.706
SMOTE-CNN	0.908	0.813	0.778	0.786
FCSSVM	0.917	0.834	0.814	0.814
ECG-CNN	0.944	0.864	<u>0.848</u>	<u>0.849</u>
CSDSResNet	0.963	0.904	0.910	0.901

Fig. 9 shows the single-class recognition results on this dataset. CSDSResNet achieved the best results for Classes 4, 6, 7, and 9, with Classes 4 and 6 being high IR minority classes. From the subplots of Classes 4 and 5 in Fig. 9, it is evident that the sharp reduction in training set size had a devastating impact on some methods. For instance, FCSSVM failed to recognize minority Class 4, and ECG-CNN failed to recognize minority Class 5. In contrast, our proposed CSDSResNet maintained good performance, which is noteworthy.

Although CSDSResNet performs exceptionally well across various multi-class imbalanced classification tasks, it improves the accuracy of minority classes but has a limited effect on the overall dataset accuracy, partly because minority class samples contribute less to the total accuracy. Additionally, the method does not specifically focus on learning features from majority class samples.

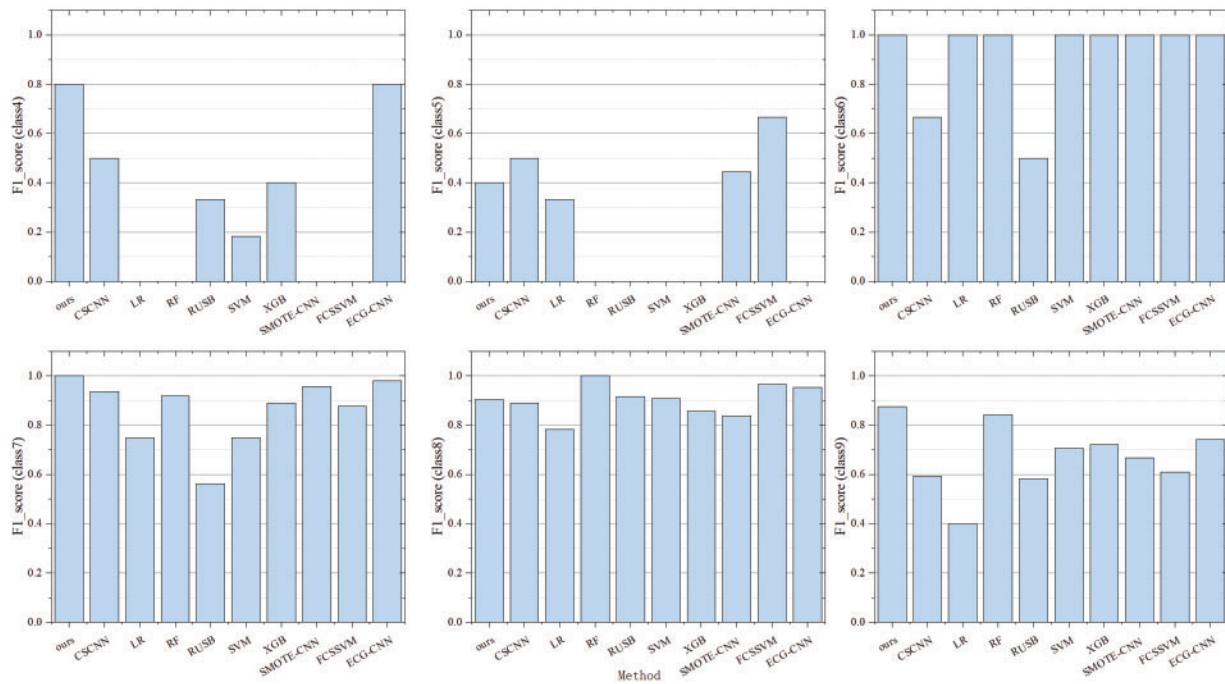


Figure 9: Single-class identification results on the Small-Sized Sensorless_drive_diagnosis

5 Conclusion

To tackle the challenges of multi-class imbalanced data with multiple minority classes and overlapping distributions, this paper proposes the cost-sensitive dual-stream residual network model (CSDSResNet). CSDSResNet employs a dual-stream residual backbone network and enhances feature extraction using convolutional kernels with odd-even differences to expand the receptive field. It also addresses the problem of attention shift due to class imbalance and performance degradation caused by overlapping sample distributions among minority classes through a cost-sensitive loss function based on sample imbalance and inter-class similarity. This function increases the model’s sensitivity to misclassifications of minority classes and strengthens its ability to distinguish among them.

In experiments on the “DryBean” and “Sensorless_drive_diagnosis” datasets, CSDSResNet outperformed existing techniques, showing notable improvements in macro F1-score of 2.72% and 1.4%, respectively. It excelled in single-class recognition, boosting precision by 13.7% for the “Sira” class and achieving a 3.7% increase in F1-score for Class 8 compared to the second-best method. CSDSResNet also demonstrated strong generalization for imbalanced tasks with smaller sample sizes, highlighting its effectiveness in complex classification scenarios. Future work will aim to further improve CSDSResNet to address a broader range of classification tasks.

Acknowledgement: This work was supported by Beijing Municipal Science and Technology Project (No. Z221100007122003). All of the mentioned support is gratefully acknowledged. In addition, thanks for all the help of the teachers and students of the related universities.

Funding Statement: Not applicable.

Author Contributions: Congcong Ma: Conceptualization, Methodology, Validation, Formal analysis, Writing–Original Draft, Visualization. Jiaqi Mi: Software, Investigation. Wanlin Gao: Resources. Sha Tao: Writing–Review & Editing. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The DryBean dataset that support the findings of this study are available at <https://doi.org/10.1016/j.compag.2020.105507> (accessed on 10 February 2024), reference number [7]. The Sensorless_drive_diagnosis dataset is derived from the following resources available in the public domain: The UCI Machine Learning Repository and URL: https://archive.ics.uci.edu/datasets?search=Sensorless_drive_diagnosis (accessed on 10 February 2024).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] L. Li, H. He, and J. Li, “Entropy-based sampling approaches for multi-class imbalanced problems,” *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 11, pp. 2159–2170, 2019. doi: [10.1109/TKDE.2019.2913859](https://doi.org/10.1109/TKDE.2019.2913859).
- [2] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, “RUSBoost: A hybrid approach to alleviating class imbalance,” *IEEE Trans. Syst., Man, Cybern. A: Syst. Humans*, vol. 40, no. 1, pp. 185–197, 2009. doi: [10.1109/TSMCA.2009.2029559](https://doi.org/10.1109/TSMCA.2009.2029559).
- [3] Y. Li, J. Wang, S. Wang, J. Liang, and J. Li, “Local dense mixed region cutting+ global rebalancing: A method for imbalanced text sentiment classification,” *Int. J. Mach. Learn. Cybern.*, vol. 10, no. 7, pp. 1805–1820, 2019. doi: [10.1007/s13042-018-0858-x](https://doi.org/10.1007/s13042-018-0858-x).
- [4] G. Fu, Y. Wu, M. Zong, and J. Pan, “Hellinger distance-based stable sparse feature selection for high-dimensional class-imbalanced data,” *BMC Bioinform.*, vol. 21, no. 1, pp. 1–14, 2020. doi: [10.1186/s12859-020-3411-3](https://doi.org/10.1186/s12859-020-3411-3).
- [5] S. A. N. Maldonado and J. L. O. Pez, “Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification,” *Appl. Soft Comput.*, vol. 67, pp. 94–105, 2018. doi: [10.1016/j.asoc.2018.02.051](https://doi.org/10.1016/j.asoc.2018.02.051).
- [6] N. Asniar, U. Maulidevi, and K. Surendro, “SMOTE-LOF for noise identification in imbalanced data classification,” *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3413–3423, 2022. doi: [10.1016/j.jksuci.2021.01.014](https://doi.org/10.1016/j.jksuci.2021.01.014).
- [7] J. H. Joloudari, A. Marefat, M. A. Nematollahi, S. S. Oyelere, and S. Hussain, “Effective class-imbalance learning based on SMOTE and convolutional neural networks,” *Appl. Sci.*, vol. 13, no. 6, 2023, Art. no. 4006. doi: [10.3390/app13064006](https://doi.org/10.3390/app13064006).
- [8] J. Lee and K. Park, “GAN-based imbalanced data intrusion detection system,” *Pers. Ubiquit. Comput.*, vol. 25, no. 1, pp. 121–128, 2021. doi: [10.1007/s00779-019-01332-y](https://doi.org/10.1007/s00779-019-01332-y).
- [9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” presented at the IEEE Conf. Comput. Vis. Pattern Recognit., Las Vegas, NV, USA, Jun. 27–30, 2016.
- [10] X. Gao *et al.*, “A multiclass classification using one-versus-all approach with the differential partition sampling ensemble,” *Eng. Appl. Artif. Intell.*, vol. 97, no. 3, 2021, Art. no. 104034. doi: [10.1016/j.engappai.2020.104034](https://doi.org/10.1016/j.engappai.2020.104034).
- [11] R. A. Mohammed, K. W. Wong, M. F. Shiratuddin, and X. Wang, “Classification of multi-class imbalanced data streams using a dynamic data-balancing technique,” in *27th Int. Conf. Neural Inf. Process.*, Bangkok, Thailand, Nov. 18–22, 2020.
- [12] Y. Freund and R. E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997. doi: [10.1006/jcss.1997.1504](https://doi.org/10.1006/jcss.1997.1504).

- [13] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, San Francisco, CA, USA, Aug. 13–17, 2016.
- [14] T. -Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," presented at the IEEE Int. Conf. Comput. Vis., Venice, Italy, Oct. 22–29, 2017.
- [15] Y. Cui, M. Jia, T. -Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," presented at the IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Long Beach, CA, USA, Jun. 16–20, 2019.
- [16] S. Soleymanpour, H. Sadr, and M. N. Soleimandarabi, "CSCNN: Cost-sensitive convolutional neural network for encrypted traffic classification," *Neural Process. Lett.*, vol. 53, no. 5, pp. 3497–3523, 2021. doi: [10.1007/s11063-021-10534-6](https://doi.org/10.1007/s11063-021-10534-6).
- [17] M. Zubair and C. Yoon, "Cost-sensitive learning for anomaly detection in imbalanced ECG data using convolutional neural networks," *Sensors*, vol. 22, no. 11, 2022, Art. no. 4075. doi: [10.3390/s22114075](https://doi.org/10.3390/s22114075).
- [18] H. Zhang, L. Jiang, and C. Li, "CS-ResNet: Cost-sensitive residual convolutional neural network for PCB cosmetic defect detection," *Expert. Syst. Appl.*, vol. 185, no. 1, 2021, Art. no. 115673. doi: [10.1016/j.eswa.2021.115673](https://doi.org/10.1016/j.eswa.2021.115673).
- [19] A. Iranmehr, H. Masnadi-Shirazi, and N. Vasconcelos, "Cost-sensitive support vector machines," *Neuro-computing*, vol. 343, no. 6, pp. 50–64, 2019. doi: [10.1016/j.neucom.2018.11.099](https://doi.org/10.1016/j.neucom.2018.11.099).
- [20] B. Zhu, X. Jing, L. Qiu, and R. Li, "An imbalanced data classification method based on hybrid resampling and fine cost sensitive support vector machine.," *Comput. Mater. Contin.*, vol. 79, no. 3, pp. 3977–3999, 2024. doi: [10.32604/cmc.2024.048062](https://doi.org/10.32604/cmc.2024.048062).
- [21] F. Milletari, N. Navab, and S. -A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *4th Int. Conf. 3D Vis. (3DV)*, Stanford, CA, USA, Oct. 25–28, 2016.
- [22] M. Koklu and I. A. Ozkan, "Multiclass classification of dry beans using computer vision and machine learning techniques," *Comput. Electron. Agric.*, vol. 174, 2020, Art. no. 105507. doi: [10.1016/j.compag.2020.105507](https://doi.org/10.1016/j.compag.2020.105507).
- [23] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.