Pattern Recognition 144 (2023) 109827

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Feature fusion network for long-tailed visual recognition

Xuesong Zhou, Junhai Zhai *, Yang Cao

Hebei Key Laboratory of Machine Learning and Computational Intelligence, College of Mathematics and Information Science, Hebei University, Baoding, 071002, Hebei, China

ARTICLE INFO

Keywords:
Long-tailed learning
Head and tail classes
Feature representations
Feature fusion network

ABSTRACT

Deep learning has achieved remarkable success in recent years; however, deep learning methods face significant challenges on long-tailed datasets, which are prevalent in real-world scenarios. In a long-tailed dataset, there are many more samples in the head classes than in the tail classes, and this class imbalance makes it difficult to learn a good feature representation for both head and tail classes simultaneously, particularly when using a single-stage method. Although the existing two-stage methods can alleviate the problem of single-stage methods not performing well on the tail classes by classifier retraining in the second stage, this does not resolve the problem of insufficient learning of head and tail features. Thus, in this paper, we propose a two-stage feature fusion network (FFN). The proposed FFN addresses this issue using one network for the head classes and another network for the tail classes, each of which is trained with a different loss function. This allows the feature learning module to effectively distinguish between the head and tail classes in the embedding space. The classifier learning module fuses the features obtained from the feature learning module, and the classifier is fine-tuned to classify the input images. Different from traditional two-stage methods, the proposed utilizes different loss functions for the head and tail classes; thus, the classifier can achieve balanced results between the head and tail classes. We conduct extensive experiments on three benchmark datasets comparing the proposed FFN with six state-of-the-art methods including two baseline methods, the experimental results demonstrate that the FFN achieves significant improvement on all three benchmark datasets. The code is publicly available at https://github.com/zxsong999/Feature-Fusion-Network.pytorch.

1. Introduction

Many traditional deep learning algorithms rely on manually collected and constructed balanced datasets, e.g., ImageNet ILSVRC [1], MS COCO [2], and Places Database [3]. However, in real-world applications, the data distribution is typically not balanced [4,5], and a long-tailed distribution (see Fig. 1) can pose significant challenges for traditional deep learning algorithms because the model may become biased toward head classes during training. This can lead to poor performance on tail classes, which are often the classes of interest in many applications, such as mechanical fault diagnosis and identification of rare animals. Long-tail visual recognition has various applications. For instance, in the field of security surveillance, it can optimize the detection and identification of uncommon security incidents and criminal behaviors. Similarly, in medical image analysis, it can be utilized to identify infrequent cases or subtypes of diseases, significantly aiding doctors in accurate diagnoses and treatment planning. Researchers have proposed various techniques to address the class imbalance issue, e.g., re-sampling, re-weighting, and feature transfer, to balance the class distribution and improve the model's performance on tail classes. These techniques attempt to increase the sample size of the tail classes during training and make the learned features more discriminative for all classes regardless of their frequency.

Motivation. Class rebalancing is a common strategy to address the long-tail distribution problem in deep learning. Here, the class distribution in the training data is modified such that head and tail classes have equal representation. This can be realized by assigning higher weights to samples from the tail classes during training, which effectively rebalances the loss and ensures that the model learns to assign equal importance to all classes. However, this strategy compromises the ability to represent the entire dataset. Although this method can improve the accuracy of the tail classes significantly, it is based on sacrificing the accuracy of the head classes [6]. For traditional deep learning methods, head classes are dominant and thus contribute more than the tail classes in updating the network parameters, resulting in bias towards head classes. Moreover, traditional methods normally underfit the tail classes and show poor performances on the tail classes in the test set. This motivated us to develop a method to balance head and tail features to address the limitations of traditional methods.

* Corresponding author.
E-mail address: mczjh@hbu.edu.cn (J. Zhai).

https://doi.org/10.1016/j.patcog.2023.109827
Received 11 March 2023; Received in revised form 18 June 2023; Accepted 21 July 2023
Available online 27 July 2023

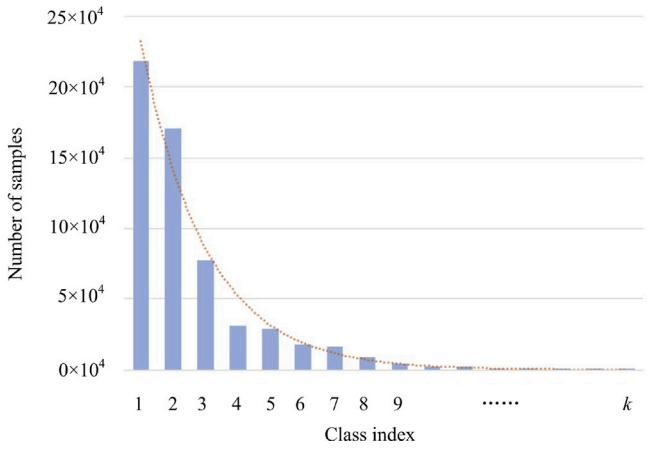

Fig. 1. Diagram of long tail distribution.

To achieve this, we proposed a multi-branch feature fusion approach. Specifically, we employ two structurally identical backbone networks to learn features using different losses. One network distinguishes head classes in the embedding space, and the other network distinguishes tail classes in the embedding space. Finally, we perform fusion on the features extracted in the first stage and retrain the classifier.

Contribution. We propose a method to address the long-tailed distribution in recognition tasks. Our contributions include: (1) The accuracy of head and tail classes is balanced by utilizing two feature extraction networks biased toward head or tail classes. Then, the extracted features are fused and used to retrain the classifier. (2) The proposed method differs from the traditional two-stage training methods, which utilizes a different loss function in the classifier training stage than the first stage. (3) The results of extensive experiments conducted on three popular long-tail datasets comparing the FFN with six state-of-the-art methods demonstrate that the proposed method outperforms the six comparative methods.

2. Related work

Long-tail image classification is a challenging research topic. To address this issue, various methods have been proposed [7]. We classify these methods into the following categories:

Data augmentation. The data augmentation method typically employs generative models, e.g., generative adversarial networks, variational autoencoders, diffusion models, and their variants, to generate new samples to increase the number of samples for tail classes. A previous study [8] employed one or more data augmentation methods to improve the low performance of traditional methods caused by an insufficient number of samples in unbalanced datasets. Feature augmentation and sampling adaptation (FASA) proposed in [9] solves the insufficient sample size problem by enhancing the feature space (especially for tail classes). Implicit semantic data augmentation (ISDA) proposed in [10] augments the number of samples for tail classes. Meta-semantic augmentation proposed in [11] is a variant of ISDA.

Class-Level re-weighting. Balanced meta-softmax proposed in [12] is a resampling strategy for long-tail learning, which explicitly considers the change of label distribution in the meta-optimization process. To make the decision boundary more favorable for tail classes, a long-tailed object detector with classification equilibrium [13] is proposed, which employs a fractionally guided equilibrium loss and a special sampler to adjust the decision boundary of the tail class features in the embedding space. To improve the deviation of the classification head in the head and the tail classes, Wang et al. [14] proposed a novel sampling scheme SimCal that initially uses image-level sampling and then uses instance-level sampling.

Metric learning. The metric learning method solves the target problem by changing the distance between the extracted features and the model classifier. For example, label-distribution-aware margin (LDAM) [15] improves the existing soft margin loss [16]. By assigning larger margins to the features of tail classes, a previous study [17] made it easier to distinguish tail classes from the head classes in the embedding space, thereby improving the accuracy of the model's tail class recognition. The method proposed in [18] is based on label frequency enforcing class-dependent margins and encouraging tail classes to have larger margins. Different from the methods in [17,18], RoBal [19] adds an extra margin to each head class to avoid the problem where the margins of tail classes are too large. The approach presented in [20] shifts the decision boundary to eliminate the a priori gap and representation gap via post-threshold processing.

Decoupling methods. Typically, decoupling methods decouple the learning process into feature learning and classifier learning processes. Kang et al. [21] proposed a two-stage long-tail learning method that utilizes a different sampling strategies in the feature learning phase and a one-stage feature extractor with fixed parameters in the classifier learning phase to train the classifier. Based on the work in [21], Kang et al. [22] employed a k-positive contrastive loss to improve the decoupled training method in order to learn a more efficient feature space and make each class easily distinguishable in the feature space. Another study [23] proposed a weight-guided class complementing framework.

Transfer learning. Liu et al. [24] found that head classes have large spatial spans, the intra-class features are diverse, the spatial spans of tail classes are much smaller than that of head classes, and there is a lack of diversity within class. To handle this problem, they proposed a solution to transfer the diversities of head classes to tail classes by transferring variances. Self-supervised pre-training [25] effectively improves imbalanced learning by utilizing unlabeled data for semi-supervised training. In addition, SSD [26] applies knowledge distillation to the long-tail recognition process. Here, the backbone network is trained on the original data through self-supervision and original label supervision, and then the balanced sampling method is employed to generate soft labels. Finally, the final classifier is combined with decoupling training.

Mixture-of-Experts. The self-supervised aggregation of diverse experts method [27] learns multi-expert solutions that specialize in different classes of distributions and combine multiple experts to alleviate the long-tail distribution problem. LFME [28] combines the advantages of multiple experts training multiple classification heads on a smaller subset, and it synthesizes all classification heads to make a judgment. Based on these experts, LFME employs multiple teacher experts for knowledge distillation to refine a unified student model. ACE [29] employs a multi-expert structure, where each expert is the most effective for the subset it is responsible for and is complementary to the other experts.

Super-class construction. A previous study [30] added a regularization term to the objective function and proposed a deep super-class learning model. The model learns both features and classifiers in an end-to-end process to obtain super-class structures. Ma et al. [31] designed a two-step training mode, where tail classes in the dataset are first aggregated into superclasses, and then the aggregated dataset is used to train a prototype model. Finally, the aggregated active super-classes are scattered to train the model to distinguish the tail classes.

This paper focuses on class-level re-weighting and decoupling method. The authors find that while class-level re-weighting can improve decision boundaries for tail classes, it often results in significantly sacrificing performance on head classes. On the other hand, the decoupling method partially solves the problem of classifier learning bias caused by insufficient samples of tail classes. To overcome these limitations, this paper proposes a Feature Fusion Network (FFN) that combines the advantages of both approaches. FFN aims to handle the problem of sacrificing performance on head classes while maintaining the excellent performance on tail classes. FFN provides a novel solution that combines the advantages of the two methods.

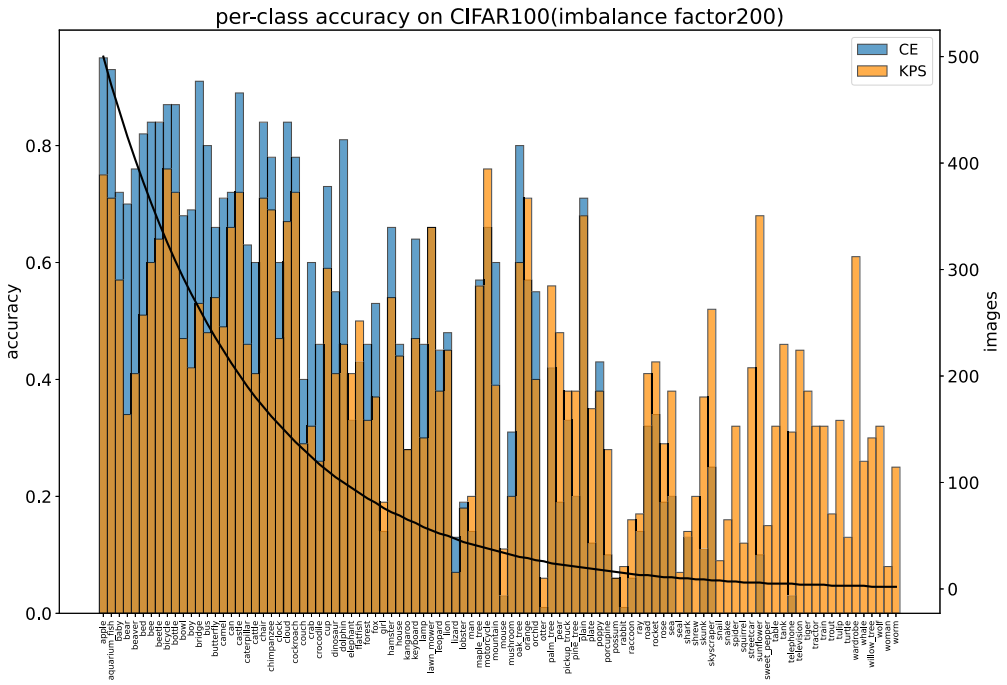

Fig. 2. Classification accuracy using KPS loss and CE loss on CIFAR100-LT (The black curve in the figure is the number of samples per class) with an imbalance factor 200. The blue part in the figure is CE Loss, and the orange part is KPS Loss. It can be seen that CE Loss is significantly better than KPS Loss on the head classes, and KPS Loss is significantly better than CE Loss on the tail classes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3. Methodology

It is observed that the traditional cross-entropy loss (CE loss) can achieve better results on head classes and Key Point Sensitive Loss (KPS Loss) can achieve better results on tail classes, and both of them have a significant drawback: CE Loss will underfit the tail classes, i.e. it performs poorly on the tail classes, and KPS Loss is to trade the accuracies of the head classes for the ones of the tail classes, as shown in Fig. 2. We want to find a way to retain its excellent tail performance without losing too much accuracies of head classes by fusing the feature extraction network of KPS with that of CE. To this end, we design a novel long-tail recognition framework called FFN.

3.1. Data augmentation methods used in the training process

To learn more effective features, we adopt mixup [32] in the feature extraction phase. Mixup can make feature learning more effective because (1) mixup can alleviate the overfitting of the head class by traditional methods; (2) mixup can improve the quality of representation learning but has little effect on classifier learning. Based on these observations, MiSLAS [33] recommends using mixup for enhancing representation learning in decoupled schemes. Please note that mixup is not used in the classifier training phase.

Mixup is a generic nearest-neighbor distribution of the original data distribution

$$\mu\left(\tilde{x}, \tilde{y} \mid x_i, y_i\right) = \frac{1}{n}\sum_{j}^{n} \mathbb{E}_{\lambda}\left[\delta\left(\tilde{x}, \tilde{y}\right)\right] \tag{1}$$

where $\lambda \sim \text{Beta}(\alpha, \alpha)$, for $\alpha \in (0, \infty)$. In a nutshell, sampling from the mixup vicinal distribution produces virtually feature-target vectors by

$$\tilde{x} = \lambda \cdot x_i + (1-\lambda)\cdot x_j \tag{2}$$

and

$$\tilde{y} = \lambda \cdot y_i + (1-\lambda)\cdot y_j \tag{3}$$

where (x_i, y_i) and (x_j, y_j) are two data points drawn randomly from the training data, and $\lambda \in (0, 1)$.

3.2. Overall framework

The overall framework of FFN is shown in Fig. 3, FFN has two key components: (1) a feature learning branch that uses two structurally identical backbone networks with different loss functions (feature learning phase), and (2) a classifier that fuses the features extracted from the two backbone networks in the feature learning phase and is fine-tuned to classify input images (classifier learning phase). Specifically, for all the networks in the feature learning phase and the classifier learning phase, a residual network with the same architecture but different weights is used. Let $(x, y) \in S$, x be the input training image, y is the image label corresponding to x, S is a dataset with k classes and n samples in total, and z_y is the predicted score of the class y. For the feature learning phase, we use two feature extraction networks, one of which uses Cross-entropy Loss.

$$L_{\text{CE}}(x, y) = -\log\left(\frac{e^{z_y}}{\sum_{j=1}^{k} e^{z_j}}\right) \tag{4}$$

The other uses KPS Loss [17]

$$L_{\text{KPS}}(x, y) = -\log\frac{e^{s\cdot\left(r_y\cos\theta_y - m_y\right)}}{e^{s\cdot\left(r_y\cos\theta_y - m_y\right)} + \sum_{j=1, j\neq y}^{k} e^{s\cdot r_j\cos\theta_j}} \tag{5}$$

where θ_y denotes the angle between the class anchor point vector w_i and the feature f, and $\cos\theta_y$ is given by

$$\cos\theta_y = \text{Norm}\left(z_y\right) = \frac{w_y^T f}{\left|w_y^T\right|_2 |f|_2} \tag{6}$$

In (5), r_y is a labeling correlation factor given by (7), and n_y is the number of samples in the class y and C_r is a constant.

$$r_y = \log n_y + C_r \tag{7}$$

In (5), m_y is the margin that is given by (8).

$$m_y = C_m - \log n_y \tag{8}$$

where $C_m > n_{max}$.

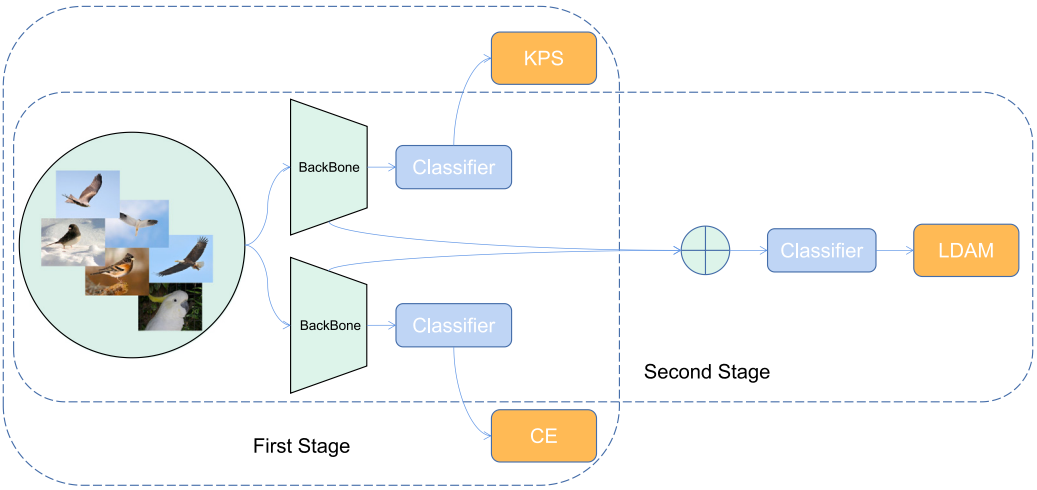

Fig. 3. The FFN is a long-tail recognition framework that addresses the problem of imbalanced class distribution in deep neural network training. The FFN framework consists of two key components: (1) Feature Learning Branch: The feature learning branch includes two structurally identical backbone networks, one biased towards the head classes and the other towards the tail classes. (2) Classifier: The classifier fuses the features extracted from the two backbone networks and fine-tuns to classify input images.

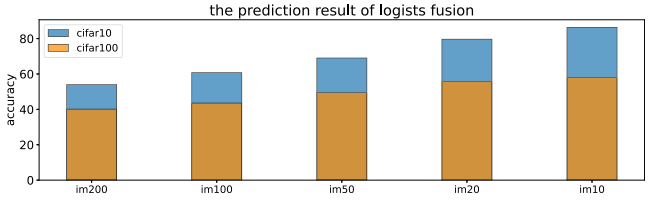

Fig. 4. The prediction result of logists fusion.

In the classifier learning phase, we use the following LDAM Loss [15]

$$L_{\mathrm{LDAM}}(x,y) = -\log \frac{e^{z_y - \Delta_y}}{e^{z_y - \Delta_y} + \sum_{j \neq y} e^{z_j}} \tag{9}$$

where the Δ_y is defined by

$$\Delta_y = \frac{C}{n_y^{1/4}} \tag{10}$$

3.3. Proposed feature fusion networks

Fusion method selection. In our method, we choose the feature fusion rather than logists fusion. The reason is that we find by experiments the prediction result of logists fusion given by the following formula is very poor, as shown in Fig. 4.

$$Logits_{\mathrm{fusion}} = w_1 \cdot Logits_{\mathrm{CE}} + w_2 \cdot Logits_{\mathrm{KPS}} \tag{11}$$

where $Logits_{CE}$ and $Logits_{KPS}$ are the softmax logits obtained by CE loss and KPS loss respectively.

After the fusion method is selected, we use the samples $(x, y) \in S$ to train the two features learning networks N_{CE} and N_{KPS} by the CE loss and KPS loss respectively, and obtained the features f_{CE} and f_{KPS} for fusing in the next phase. The f_{CE} and f_{KPS} are the embedding space features extracted by N_{CE} and N_{KPS} respectively, which are given by

$$f_{\mathrm{CE}} = N_{\mathrm{CE}}(x) \tag{12}$$

and

$$f_{\mathrm{KPS}} = N_{\mathrm{KPS}}(x) \tag{13}$$

Classifier Selection. In the classifier learning phase, f_{KPS} and f_{CE} obtained in the feature extraction phase are used as inputs for classifier training, and the output logits are expressed as

$$z = w_1 \cdot f_{\mathrm{KPS}} + w_2 \cdot f_{\mathrm{CE}} \tag{14}$$

where $z \in R^D$ is the predicted output, i.e., $[z_1, z_2, \ldots, z_D]^T$, w_1, w_2 are the fusion coefficients of f_{KPS} and f_{CE}, $w_1, w_2 \in [0, 1]$ and $w_1 + w_2 = 1$. Finally, LDAM Loss [15] is applied to calculate the Loss.

Table 1
The basic setup of the feature learning phase.

Dataset	CIFAR10-LT	CIFAR100-LT	ImageNet-LT
Backbone	ResNet-32	ResNet-32	ResNet-50
Initial l_r	0.1	0.1	0.1
$l_r$warm-up	Yes	Yes	No
Batch size	64	64	64
Weight decay	2×10^{-4}	2×10^{-4}	1×10^{-4}
Epochs	200	200	200
l_r decay ratio	0.01	0.01	0.1
l_r decay epochs	160, 180	160, 180	120, 160

Table 2
The basic settings of the classifier learning phase.

Dataset	CIFAR10-LT	CIFAR100-LT	ImageNet-LT
Backbone	ResNet-32	ResNet-32	ResNet-50
Initial l_r	0.1	0.1	0.1
Batch size	64	64	64
Weight decay	2×10^{-4}	2×10^{-4}	1×10^{-4}
Epochs	30	30	30
Fusion coefficient w_1	0.5	0.9	0.9
Fusion coefficient w_2	0.5	0.1	0.1

4. Experiments

4.1. Datasets

The proposed method was evaluated on three benchmark datasets i.e., the CIFAR10-LT, CIFAR100-LT, and ImageNet-LT datasets. Note that the original versions of CIFAR-10 [34], CIFAR-100 [34], and ImageNet-2012 [1] are all balanced datasets. In this evaluation, we employ the long-tailed versions of the CIFAR-10, CIFAR-100, and ImageNet-2012 datasets [35,36].

CIFAR-10/100 LT. The CIFAR-10/100 LT dataset is a modified version of the original CIFAR-10/100 dataset [35]. This dataset has an exponentially decreasing number of training samples for each class. By a factor $\mu \in (0, 1)$, the number of samples per class is decreased in a

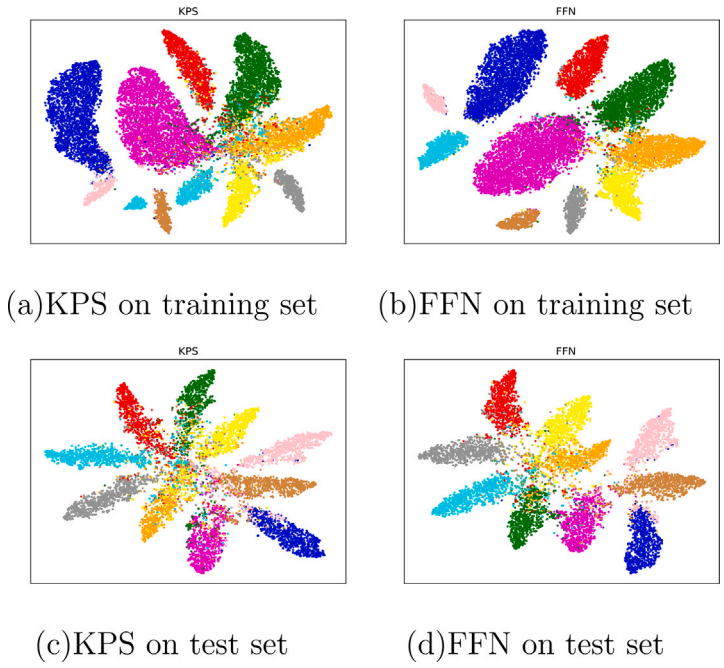

Fig. 5. Visualization of the embedding via t-SNE on CIFAR10-LT with $\gamma = 10$, where backbone network is ResNet-32.

controlled manner, where μ is a hyperparameter regulating the level of imbalance. Note that the test data are not altered. We conducted experiments using CIFAR-10 and CIFAR-100 LT datasets, which have imbalance factors of 10, 20, 50, 100, and 200, respectively.

ImageNet-LT. The original ImageNet-2012 is a large real-world dataset for image classification and object detection. We build on the work of Liu et al. by creating a long-tailed version of the dataset using the Pareto distribution [36] and power-valued samples $\alpha = 6$ from the original ImageNet-2012 dataset [1]. The test set has not been altered. With a maximum of 1280 images per category and a minimum of five images per category, ImageNet-LT has a total of 115.8 K images from 1000 categories.

4.2. Experimental settings

Basic settings of CIFAR10/100-LT. The CIFAR10/100-LT use the following augmentation strategy [37]. (1) randomly crop a 32 × 32 region from an image. (2) flip the image horizontally with probability 0.5. (3) pad four pixels on each side of the cropped region with the average of the pixels in the image. This strategy is utilized to increase the variability of the training data and prevent overfitting. Other basic settings are shown in Table 1.

Basic setup of ImageNet-LT. The basic setup is summarized as follows. (1) The shorter dimension of the image was scaled to 256. (2) a random 224 × 224 patch or its horizontal flip was cropped from the scaled image for data augmentation following the strategy presented in the literature [38]. Other basic settings, such as the optimizer and learning rate schedule are shown in Table 2.

4.3. Compared methods and evaluation metric

In these experiments,we compare the proposed method with two baseline methods: cross-entropy loss (CE loss) and class-balanced softmax CE Loss (CBL). In addition, we compared the FFN model with the four state-of-the-art methods KPS [17], LDAM-DRW [15], CE loss + mixup [32] and BBN [6], and the top-1 classification accuracy was used as the evaluation metric.

Table 3
Comparative results of top-1 classification accuracy on CIFAR-10/100-LT and ImageNet-LT using different fusion coefficients (%).

Fusion coefficient	CIFAR10-LT	CIFAR100-LT	ImageNet-LT
Imbalance factor	200	200	–
$w_1 = 0.0$ $w_2 = 1.0$	79.47	20.2	5.56
$w_1 = 0.1$ $w_2 = 0.9$	79.81	23.97	12.81
$w_1 = 0.2$ $w_2 = 0.8$	80.46	29.66	32.4
$w_1 = 0.3$ $w_2 = 0.7$	81.43	35.9	42.23
$w_1 = 0.4$ $w_2 = 0.6$	81.7	39.81	46.41
$w_1 = 0.5$ $w_2 = 0.5$	**82.65**	41.35	48.35
$w_1 = 0.6$ $w_2 = 0.4$	81.83	42.23	49.31
$w_1 = 0.7$ $w_2 = 0.3$	80.1	42.55	49.87
$w_1 = 0.8$ $w_2 = 0.2$	78.74	42.75	50.01
$w_1 = 0.9$ $w_2 = 0.1$	77.83	**43.02**	**50.54**
$w_1 = 1.0$ $w_2 = 0.0$	77.74	42.87	49.2

4.4. Ablation study

Ablation study on fusion coefficients. To more clearly show the impact of fusion coefficients w_1 and w_2 on the performance of FFN, we conduct ablation experiments on fusion coefficients. In our experiments, we use the highest imbalance factor of 200 in CIFAR10-LT and CIFAR100-LT, which can fully demonstrate the effectiveness of FFN on these three datasets. In Table 3 we investigate the impact of these coefficients on classification accuracy for the three datasets. Experimental results show that the fusion coefficients used in the Table 2 produce the most favorable results, but in CIFAR100-LT and ImageNet-LT, the performance varies greatly due to the difference in the imbalance factor, which is caused by inappropriate feature fusion.

Ablation study on data augmentation. To demonstrate the performance gain of FFN mainly comes from feature fusion rather than data augmentation, or in other words, the contribution of feature fusion is much more significant than that of data augmentation, we conducted ablation study on data augmentation, and the experimental results are listed in Table 4. The experimental results in Table 4 confirm the correctness of the above conclusions.

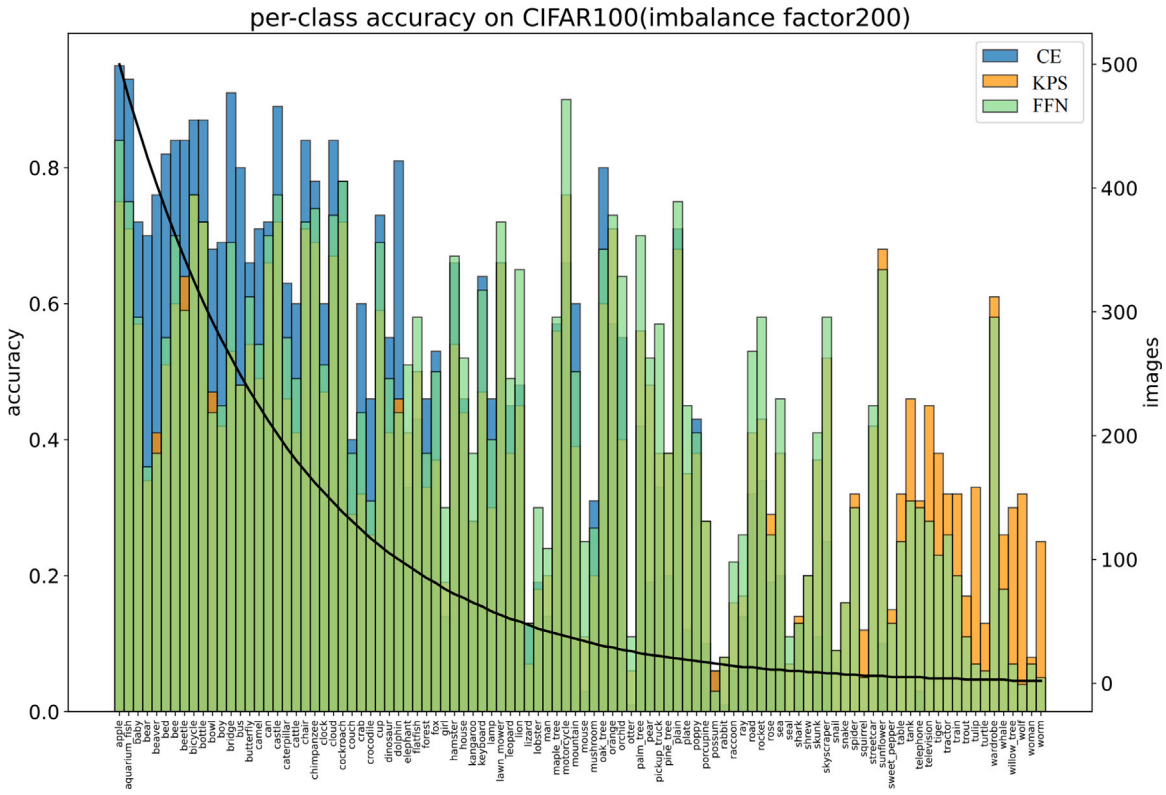

Fig. 6. Classification accuracy of CE, KPS and FFN on the CIFAR100 with imbalance factor 200(The black curve in the figure is the number of samples per class).

Table 4
Comparative results on CIFAR-10/100-LT, ImageNet-LT in top-1 classification accuracy (%).

Datasets	CIFAR10-LT					CIFAR100-LT					ImageNet-LT
Backbone	ResNet-32										ResNet-50
Imbalance factor	200	100	50	20	10	200	100	50	20	10	–
FFN	**82.65**	**85.24**	**87.67**	**89.54**	**91.18**	**43.02**	**47.51**	**50.01**	**56.34**	**59.3**	**50.54**
FFN - mixup	79.41	83.04	85.84	88.45	89.38	42.05	46.8	49.25	55.37	57.64	50.2

Table 5
Comparative results on CIFAR-10/100-LT, ImageNet-LT in top-1 classification accuracy (%).

Datasets	CIFAR10-LT					CIFAR100-LT					ImageNet-LT
Backbone	ResNet-32										ResNet-50
Imbalance factor	200	100	50	20	10	200	100	50	20	10	–
CE Loss	65.68	70.7	74.81	82.23	86.39	34.84	38.43	43.9	51.14	55.71	44.51
CBL [35]	68.99	73.82	80.25	84.92	88.2	36.23	39.6	45.32	52.59	57.99	–
CE loss + mixup [32]	65.84	72.96	79.48	–	–	35.84	40.01	45.16	–	–	45.66
LDAM-DRW [15]	73.52	77.03	81.03	–	–	38.91	42.04	47.62	–	–	48.80
KPS [17]	76.92	79.93	83.91	86.07	88.56	40.06	44.79	48.62	54.67	58.04	49.34
BBN [6]	73.47	79.82	81.18	–	–	37.21	42.56	47.02	–	–	44.70
FFN	**82.65**	**85.24**	**87.67**	**89.54**	**91.18**	**43.02**	**47.51**	**50.01**	**56.34**	**59.3**	**50.54**

4.5. Results

To investigate the effectiveness of the proposed FFN, we compare FFN with four state-of-the-art methods and two baseline methods, and the top-1 accuracies for all methods are reported in Table 5. Please note that all methods are trained on the imbalanced training set while tested on the unaltered and balanced test set.

CIFAR10/100-LT. It is observed that our proposed FFN showed superior performances in all imbalance factors, especially with the imbalance factor of 200. The proposed method exhibits significant improvements, achieving 82.65% and 43.02% in top-1 classification accuracy, outperforming the KPS method [17] by 5.73% and 2.96%, respectively.

ImageNet-LT. The proposed FFN method achieved a top-1 classification accuracy of 50.54% on this dataset, achieving a 6.03% improvement over the benchmark method.

Model validation and analysis. The t-SNE visualization results shown in Fig. 5 demonstrate that compared with the embedded features obtained by the KPS Loss method, the embedded features obtained by the proposed FFN method are more scattered between classes and more compact within classes with smaller blurred regions. These results demonstrate that the proposed FFN method is more effective in terms of capturing the underlying structure of the data as because it produces more distinct and separable embedded features. Fig. 6 shows the Top-1 accuracy results for the CIFAR100-LT dataset. As can be seen, the proposed FFN method outperformed the CE and KPS methods in terms of both head and tail classes. These results indicate that FFN can

combine the benefits of CE and KPS because it can retain the tail-class effects of KPS while absorbing the head-class effects of CE. Overall, the experimental results show that the proposed FFN method can address the class imbalance issue in the CIFAR100-LT dataset because it can produce more separable embedded features, differentiate between head classes, and perform better with tail classes.

5. Conclusion

In view of the shortcoming of existing two-stage long-tailed visual recognition methods, this paper proposed an method based on feature fusion network. The proposed approach has three advantages: (1) it can effectively improve the recognition accuracy of the samples in tail classes without deteriorating the recognition accuracy of samples in head classes, this merit distinguishes the proposed method from the existing two-stage methods. (2) The proposed feature fusion network can take into account both head classes and tail classes simultaneously. The features learned by the feature fusion network have good representation ability for samples of both head classes and tail classes. (3) The long-tailed recognition accuracy of the proposed approach is improved on all three datasets, CIFAR10-LT, CIFAR100-LT, and ImageNet-LT. There is room for improvement of the proposed approach on large scale long-tailed datasets. Future work of this study should focus on two core aspects. Inspired by multimodality learning, the feature fusion mechanism can be extended to fusing features from different modalities. In addition, feature fusion can be extended to more challenging problems, e.g., open long-tailed recognition tasks.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This research is supported by the key R&D program of science and technology foundation of Hebei Province (19210310D), and by the natural science foundation of Hebei Province (F2021201020).

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (2015) 211–252.

[2] T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer, 2014, pp. 740–755.

[3] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, Places: A 10 million image database for scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. 40 (6) (2017) 1452–1464.

[4] P. Davies, Kendall's advanced theory of statistics. Volume 1. Distribution theory, 1988.

[5] M. Li, H. Sun, C. Lin, C.G. Li, J. Guo, The devil in the tail: Cluster consolidation plus cluster adaptive balancing loss for unsupervised person re-identification, Pattern Recognit. 129 (2022) 108763.

[6] B. Zhou, Q. Cui, X.S. Wei, Z.M. Chen, Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9719–9728.

[7] L. Yang, H. Jiang, Q. Song, J. Guo, A survey on long-tailed visual recognition, Int. J. Comput. Vis. 130 (7) (2022) 1837–1872.

[8] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (1) (2019) 1–48.

[9] Y. Zang, C. Huang, C.C. Loy, Fasa: Feature augmentation and sampling adaptation for long-tailed instance segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3457–3466.

[10] Y. Wang, X. Pan, S. Song, H. Zhang, G. Huang, C. Wu, Implicit semantic data augmentation for deep networks, Adv. Neural Inf. Process. Syst. 32 (2019).

[11] S. Li, K. Gong, C.H. Liu, Y. Wang, F. Qiao, X. Cheng, Metasaug: Meta semantic augmentation for long-tailed visual recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 5212–5221.

[12] J. Ren, C. Yu, X. Ma, H. Zhao, S. Yi, et al., Balanced meta-softmax for long-tailed visual recognition, in: Advances in Neural Information Processing Systems, Vol. 33, 2020, pp. 4175–4186.

[13] C. Feng, Y. Zhong, W. Huang, Exploring classification equilibrium in long-tailed object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3417–3426.

[14] T. Wang, Y. Li, B. Kang, J. Li, J. Liew, S. Tang, S. Hoi, J. Feng, The devil is in classification: A simple framework for long-tail instance segmentation, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16, Springer, 2020, pp. 728–744.

[15] K. Cao, C. Wei, A. Gaidon, N. Arechiga, T. Ma, Learning imbalanced datasets with label-distribution-aware margin loss, in: Advances in Neural Information Processing Systems, Vol. 32, 2019.

[16] F. Wang, J. Cheng, W. Liu, H. Liu, Additive margin softmax for face verification, IEEE Signal Process. Lett. 25 (7) (2018) 926–930.

[17] M. Li, Y.M. Cheung, Z. Hu, Key point sensitive loss for long-tailed visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2022).

[18] V. Koltchinskii, D. Panchenko, Empirical margin distributions and bounding the generalization error of combined classifiers, Ann. Statist. 30 (1) (2002) 1–50.

[19] T. Wu, Z. Liu, Q. Huang, Y. Wang, D. Lin, Adversarial robustness under long-tailed distribution, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8659–8668.

[20] M.L. Zhang, X.Y. Zhang, C. Wang, C.L. Liu, Towards prior gap and representation gap for long-tailed recognition, Pattern Recognit. 133 (2023) 109012.

[21] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, Y. Kalantidis, Decoupling representation and classifier for long-tailed recognition, in: International Conference on Learning Representations.

[22] B. Kang, Y. Li, S. Xie, Z. Yuan, J. Feng, Exploring balanced feature spaces for representation learning, in: International Conference on Learning Representations, 2021.

[23] X. Zhao, J. Xiao, S. Yu, H. Li, B. Zhang, Weight-guided class complementing for long-tailed image recognition, Pattern Recognit. (2023) 109374.

[24] J. Liu, Y. Sun, C. Han, Z. Dou, W. Li, Deep representation learning on long-tailed data: A learnable embedding augmentation perspective, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 2970–2979.

[25] Y. Yang, Z. Xu, Rethinking the value of labels for improving class-imbalanced learning, in: Advances in Neural Information Processing Systems, Vol. 33, 2020, pp. 19290–19301.

[26] T. Li, L. Wang, G. Wu, Self supervision to distillation for long-tailed visual recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 630–639.

[27] Y. Zhang, B. Hooi, H. Lanqing, J. Feng, Self-supervised aggregation of diverse experts for test-agnostic long-tailed recognition, in: Advances in Neural Information Processing Systems.

[28] L. Xiang, G. Ding, J. Han, Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16, Springer, 2020, pp. 247–263.

[29] J. Cai, Y. Wang, J.N. Hwang, Ace: Ally complementary experts for solving long-tailed recognition in one-shot, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 112–121.

[30] Y. Zhou, Q. Hu, Y. Wang, Deep super-class learning for long-tail distributed image classification, Pattern Recognit. 80 (2018) 118–128.

[31] Y. Ma, M. Kan, S. Shan, X. Chen, Learning deep face representation with long-tail data: An aggregate-and-disperse approach, Pattern Recognit. Lett. 133 (2020) 48–54.

[32] H. Zhang, M. Cisse, Y.N. Dauphin, D. Lopez-Paz, Mixup: Beyond empirical risk minimization, in: International Conference on Learning Representations.

[33] Z. Zhong, J. Cui, S. Liu, J. Jia, Improving calibration for long-tailed recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16489–16498.

[34] A. Torralba, R. Fergus, W.T. Freeman, 80 Million tiny images: A large data set for nonparametric object and scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. 30 (11) (2008) 1958–1970.

[35] Y. Cui, M. Jia, T.Y. Lin, Y. Song, S. Belongie, Class-balanced loss based on effective number of samples, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9268–9277.

[36] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, S.X. Yu, Large-scale long-tailed recognition in an open world, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2537–2546.

[37] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[38] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, K. He, Accurate, large minibatch sgd: Training imagenet in 1 hour, 2017, arXiv preprint arXiv:1706.02677.

Xuesong Zhou received his B.S. degree in Software engineering from Hebei University, Baoding, China, in 2021. He is currently a M.S. candidate of College of Mathematics and Information Science, Hebei University, Baoding, China. His main research interests include machine learning and deep learning.

Junhai Zhai received his B.S. degree in Mathematics and M.S. degree in Computing Mathematics from Lanzhou University, Lanzhou, China, in June 1988 and June 2000 respectively, Ph.D. degree in Optical engineering from Hebei University, Baoding, China, in 2010. He is currently a Professor with College of Mathematics and Information Science, Hebei University, Baoding, China. His main research interests include machine learning, deep learning, big data processing.

Yang Cao received his B.S. degree in Software engineering from Qingdao University, Qingdao, China, in 2021. He is currently a M.S. candidate of College of Mathematics and Information Science, Hebei University, Baoding, China. His main research interests include multisource data fusion and time series prediction.