SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2024, Vol. 67, Iss. 6, 162102:1–162102:18
https://doi.org/10.1007/s11432-023-3922-2

Meta label associated loss for fine-grained visual recognition

Yanchao LI[1], Fu XIAO[1*], Hao LI[2], Qun LI[1] & Shui YU[3]

[1]*School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing* 210023*, China;*
[2]*School of Network Engineering, Zhoukou Normal University, Zhoukou* 466001*, China;*
[3]*School of Computer Science, University of Technology Sydney, Sydney* 2007*, Australia*

Received 29 March 2023/Revised 12 July 2023/Accepted 28 September 2023/Published online 15 May 2024

Abstract Recently, intensive attempts have been made to design robust models for fine-grained visual recognition, most notably are the impressive gains for training with noisy labels by incorporating a reweighting strategy into a meta-learning framework. However, it is limited to up or downweighting the contribution of an instance for label reweighting approaches in the learning process. To solve this issue, a novel noise-tolerant method with auxiliary web data is proposed. Specifically, first, the associations made from embeddings of well-labeled data with those of web data and back at the same class are measured. Next, its association probability is employed as a weighting fusion strategy into angular margin-based loss, which makes the trained model robust to noisy datasets. To reduce the influence of the gap between the well-labeled and noisy web data, a bridge schema is proposed via the corresponding loss that encourages the learned embeddings to be coherent. Lastly, the formulation is encapsulated into the meta-learning framework, which can reduce the overfitting of models and learn the network parameters to be noise-tolerant. Extensive experiments are performed on benchmark datasets, and the results clearly show the superiority of the proposed method over existing state-of-the-art approaches.

Keywords label associated loss, weighting noisy samples, fine-grained visual recognition, noise-tolerant learning, meta-learning

1 Introduction

Fine-grained visual analysis (FGVA) has garnered more and more attention in both academia and industry in the fields of image recognition, image retrieval, and image generation [1–7]. The objective of FGVA is to learn objects with subordinate categories, for example, species of dogs or models of cars [8]. It is a demanding task because of the fine-grained nature of small interclass and large intraclass variations [9]. Deep learning has been successfully implemented in FGVA, and fully supervised deep learning models can realize a promising performance with massive well-labeled data [10–12]. However, well-labeled fine-grained domain data are costly and limited, which inspires us to incorporate auxiliary web data into the weakly supervised setting.

Furthermore, exploiting side information is an effective approach to enhancing model performance, and web-supervised learning has also been explored for different tasks via incorporating web data into the trained model, such as visual categorization [4, 13–15], domain adaptation [16], and data argumentation [17, 18]. The inspiration behind introducing web data arises from the need to increase the fine-grained dataset with a larger and more diverse set of images. Web data offers a vast number of images that cover a wide range of variations. Although web data naturally include ground noise due to the lack of precise labeling, they can still be valuable in complementing well-labeled fine-grained datasets. As opposed to earlier investigations [19, 20], which mostly filter or remove noisy images among web datasets, these web data are treated as part of training data. Moreover, a previous study [21] suggested that it is not imperative to clean the web data as the amount is large. Therefore, coupling noisy label learning with fine-grained learning is very meaningful and saves much time in annotating a large amount of unlabeled data.

[*] Corresponding author (email: xiaof@njupt.edu.cn)

Learning with noisy labels has been proposed for visual analysis [22–26]. Strategies that utilize regularization techniques include weight decay [27], dropout [28], adversarial training [3], and maximum-entropy training [29]. However, they do not explicitly deal with noisy labels. Moreover, the noise transition matrix has been incorporated into deep neural networks (DNNs) [30], but this does not carefully address the fine-grained nature of the DNN. The objective of these approaches is to leverage the complementary strengths of both the well-labeled dataset and the web data to solve the label noise problem in fine-grained visual recognition. Reducing the influence of the gap between these two types of data could reduce the impact of label noise.

Meta-learning has been successfully utilized for many applications, including hyperparameter tuning, optimizer learning, adaptation to new tasks, and neural architecture search. Instance reweighting can be learned by developing an optimization problem of the main model, and it allows the main model to communicate with each other, and a better model can be learned [31]. However, one of the restrictions of label reweighting is that it is limited to up or downweighting the contribution of an instance in the meta-learning process: (1) it is effective for estimating or measuring the possibility of samples, and it produces promising results to some extent. On the other hand, estimating sample weights usually depends on extra knowledge; (2) learning with noisy labeled datasets based on deep networks causes degradation of model performance because they may easily overfit label noise and training set biases. This work leverages instance reweighting and meta-learning to learn with noisy labels.

In this work, a noise-tolerant method is proposed using auxiliary web data. Specifically, two noise-tolerant strategies are designed to enhance the robustness of models from input data to classification: (1) measuring the associations that make from embeddings of well-labeled data to those of web data and back at the same class, and (2) using its association probability as a weighting strategy into angular margin-based loss (AM loss), which makes the trained model robust to noisy datasets. Also, a bridge schema is proposed via the corresponding loss that encourages the learned embeddings to be coherent between well-labeled and web data, thereby reducing the influence of the gap between well-labeled and noisy web data. Finally, the formulation of the method is well encapsulated in a meta-learning framework that can reduce model overfitting during the conventional update and learn the network parameters to be noise-tolerant. Furthermore, the updated model (student model), aided by the teacher model, is constructed using a self-assembling method and yields consistent predictions.

Recently, AM losses have explored the intrinsic consistency with Softmax loss, which realizes promising results. Also, incorporating a weighting strategy into AM loss realizes a robust performance. The proposed association schema is utilized in weighted AM loss during the training phase, and the proposed model can reduce the effect of noisy data by adaptively adjusting the instance weights. In addition, instead of designing a specific model to clean noisy labels, a meta-learning-based noise-tolerant method is proposed to train with noisy labeled data. The main contributions of this work can be summarized as follows:

(1) A novel noise-tolerant method is proposed using auxiliary web data for fine-grained visual recognition. The designed association and bridge schemes make our algorithm robust to noisy labels, thereby reducing the influence of the gap between well-labeled and web data.

(2) The developed model is expressed as a meta-objective, which is robust to noisy labels by adaptively learning the updated models. Also, the teacher-student training paradigm generates consistent predictions.

(3) A novel instance reweighting fusion strategy is proposed, which allows the example weights learning and the main model to share with each other and a better model to be learned.

(4) The experimental study on benchmark datasets shows that the developed method is effective and realizes state-of-the-art results against the existing FGVA models.

This paper is structured as follows. Section 2 presents the related literature. Section 3 highlights the framework for the noise-tolerant method. Section 4 provides the extensive experiments and analysis. Sections 5 makes the discussions of our proposed work. Finally, the conclusion and future directions are highlighted in Section 6.

2 Related work

This work is closely associated with studies on weakly supervised learning and training with noisy data.

2.1 Weakly supervised fine-grained categorization

Recently, weakly supervised FGVA has been extensively explored [4, 5, 14, 32–35]. Several FGVA algorithms have been put forward for image analysis, such as image recognition [36], image retrieval [2], and image generation [3]. For instance, Zhang et al. [14] eliminated the irrelevant examples and employed highly informative examples for training and updating networks. Xu et al. [5] developed a discrimination-aware mechanism incorporated into proxy-based and pair-based losses for fine-grained representation learning. The method by [32] solves the issues of discriminative region diffusion, and an efficient feature-oriented model is proposed for fine-grained visual recognition. In [33], they exploited the region correlations of images and suggested a graph-propagation-based learning model for weakly supervised fine-grained recognition. To this end, many deep learning-based methods have been investigated to utilize fine-grained feature representation.

However, most of the FGVA methods have been learned from well-labeled clean datasets [37–39]. For noisy datasets, many methods deal with noisy labels, for instance, learning with noisy data [24, 40], learning with web data [4, 13, 17], and noise-tolerant learning [41–43]. The commonality between them is that they attempt to enhance the model's robustness. For instance, the method by [34] is a data augmentation method for the FGVA algorithm. However, there is a gap between well-labeled and web data due to no coherence between these two types of data. Moreover, the model tends to overfit the label noise from web data resulting from long-term training, and many DNN-based regularization tricks do not work well, such as weight decay and early stopping. Our developed method can learn with noisy web data. Furthermore, we describe the objective in a meta-learning framework that is noise-tolerant and can adaptively update the parameters during the training process. Our work attains comparable results on different datasets.

2.2 Training with noisy data

Web data certainly include considerable noise. Weighting training samples is a well-known method for learning with noisy datasets and has been applied for training DNN models [27, 44–46]. For instance, Li et al. [47] employed the knowledge graph to reduce the effect of noisy data. Han et al. [48] considered the structure prior and derived a probabilistic model for training with noisy samples. Also, incorporating the weighting strategy into deep networks has reached robust performance, and many different paradigms have been suggested for analysis, such as curriculum learning, MentorNet, and co-teaching. For instance, Ren et al. [27] designed a meta-learning-based method for weighting samples. Shu et al. [44] built a loss for learning weight in natural language processing tasks. To this end, incorporating a sample weighting strategy into deep network models exhibits promising performance on different tasks. However, it is a challenge to better incorporate the weighting training samples into deep networks. In this approach, the estimation is simplified, and the association probability is measured, which demonstrates that the learned embeddings are coherent between the web and the well-labeled datasets. The proposed association schema is utilized in weighted AM loss during the training phase and can reduce the effect of noisy examples by adaptively adjusting the instance weights.

In recent years, meta-learning methods for DNNs have shown promising performance [49–53]. The benefits of meta-learning include learning the robust network parameters and reducing overfitting during the conventional update. Normally, the noise-tolerant setting of meta-learning has two formulations: learning robust update rules [52] and finding good weight initializations [54]. The objective of these approaches is to learn the better parameters of the model from both the well-labeled and web datasets. Our method and traditional model-agnostic meta-learning (MAML) [54] belong to model-agnostic. However, unlike traditional MAML, our approach could be noise-tolerant when performing gradient updates on metatasks. Also, MAML trains the objective loss of the classification model on the metatest set, while our method employs a consistency loss by building the teacher and student network. Moreover, the teacher-student framework is utilized for training the student model on metatest data, which could be more tolerant of noisy labels. To the best of our knowledge, little work has been performed on fine-grained visual recognition.

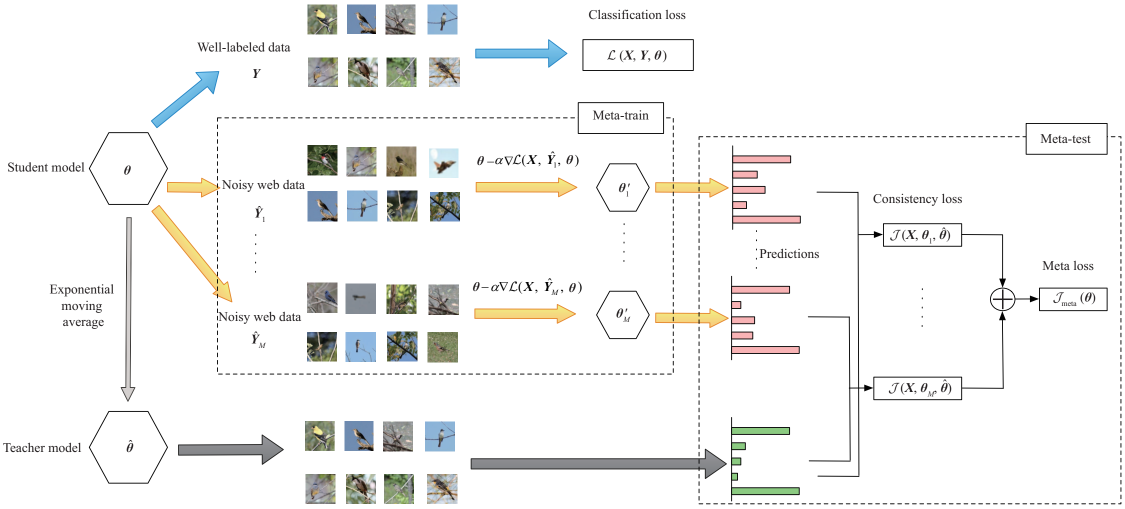
Classification loss
Well-labeled data
$\boldsymbol{Y}$
$\mathcal{L}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\theta})$
Meta-train
Meta-test
Student model
$\boldsymbol{\theta}$
Noisy web data
$\hat{\boldsymbol{Y}}_1$
$\boldsymbol{\theta} - \alpha\nabla\mathcal{L}(\boldsymbol{X}, \hat{\boldsymbol{Y}}_1, \boldsymbol{\theta})$
$\boldsymbol{\theta}'_1$
Consistency loss
$\mathcal{J}(\boldsymbol{X}, \boldsymbol{\theta}_1, \hat{\boldsymbol{\theta}})$
Predictions
Meta loss
$\mathcal{J}_{\text{meta}}(\boldsymbol{\theta})$
Exponential moving average
Noisy web data
$\hat{\boldsymbol{Y}}_M$
$\boldsymbol{\theta} - \alpha\nabla\mathcal{L}(\boldsymbol{X}, \hat{\boldsymbol{Y}}_M, \boldsymbol{\theta})$
$\boldsymbol{\theta}'_M$
$\mathcal{J}(\boldsymbol{X}, \boldsymbol{\theta}_M, \hat{\boldsymbol{\theta}})$
Teacher model
$\hat{\boldsymbol{\theta}}$

Figure 1 (Color online) Illustration of the proposed noise-tolerant method.

3 The proposed method

Given the web dataset $\mathcal{D}_w = \{(\boldsymbol{x}_i^w, y_i^w)\}_{i=1}^{N_w}$ and well-labeled dataset $\mathcal{D}_s = \{(\boldsymbol{x}_j^s, y_j^s)\}_{j=1}^{N_s}$, where N_s (N_w) is the number of well-labeled (web) images. Each image $\boldsymbol{x}_j^s$ $(\boldsymbol{x}_j^w)$ has the tag y_j^s (y_j^w) and the number of C classes. Note that the image $\boldsymbol{x}_j^w$ has the tag y_j^w, which is assigned by the keyword used to search websites. The combined datasets $\mathcal{D} = \{(\boldsymbol{x}_k, y_k)\}_{k=1}^{N}$, where $N = N_s + N_w$ and each image has an indicator tag I_n, which shows whether $\boldsymbol{x}_k$ comes from a well-labeled dataset $(I_n = 1)$ or a web dataset $(I_n = 0)$. The objective of this method is to design a robust model on the well-labeled dataset $\mathcal{D}_s$ via incorporating the abundant web dataset $\mathcal{D}_w \subset \mathcal{D}$.

In the following, the association schema, angular margin-based loss, bridge schema, and meta-learning of our proposed method will be introduced. Figure 1 displays an illustration of our proposed method. Before training on the conventional classification loss, a meta-loss is minimized for each minibatch of training data. To produce synthetic label noise resembling the original data's distribution, multiple minibatches with symmetric and asymmetric label noise (marked as the yellow arrow) are produced to preserve the underlying noise conditions. For each synthetic minibatch, the parameters are updated using gradient descent, and the updated model is needed to generate predictions consistent with a teacher model. The overall objective is to minimize consistency loss across all updated models with respect to θ.

Association schema. A general assumption behind association probability is that good embeddings will have high similarity if they are of the same class [28]. A batch of web and a batch of well-labeled examples are fed into the convolutional neural network (CNN), leading to embedding vectors (E^w and E^s). The objective of this proposed method is to optimize the parameters of a CNN that produces good embeddings and utilizes both well-labeled and web data. Computation traversing is conducted from E^s to E^w based on the mutual similarities and back. The transition obeys a probability distribution acquired from the similarity of its corresponding embeddings, which we refer to as an association. The association schema measures the probability from E^s to E^w and back to E^s, ending in the same class. This results in the measurement of the weight of the instance, which is incorporated into the angular margin-based loss later.

Definition 1. The similarity between embeddings E^s and E^w is defined as

$$M_{ij} := E_i^s \cdot E_j^w, \tag{1}$$

where E^s and E^w indicate the matrices whose rows index the instances in the batches, and the similarity measurement, such as Euclidean distance, can generally be replaced by any other similarity metric.

Definition 2. The probabilities from E^s to E^w by Softmaxing M over columns are

$$\begin{aligned} P_{ij}^{sw} &= P(E_j^w | E_i^s) := (\text{Softmax}_{\text{cols}}(M))_{ij} \\ &= \exp(M_{ij}) \Big/ \sum_{j'} \exp(M_{ij'}). \end{aligned} \tag{2}$$

Algorithm 1 Learning to reweight examples

```
1: Initialize: Randomly initialize $\boldsymbol{\theta}$, $\ell_c = \ell$, well-labeled web data $N_s$, $N_w, n, m$.
2: repeat
3:     Randomly sample batch data $(N_s, n)$ of size $n$ from $N_s$;
4:     Randomly sample batch data $(N_w, m)$ of size $m$ from $N_w$;
5:     Forward$((N_s, n), (N_w, m), \boldsymbol{\theta})$;
6:     Calculate the association probability $P^{sws}$;
7:     Calculate the weighted AM loss $\ell_c$;
8:     Backward$(\ell_c, \boldsymbol{\theta})$;
9: until Stopping criteria are met.
```

Therefore, the association probability of starting at E_i^s and ending at E_j^s is

$$P_{ij}^{sws} = (P^{sw}P^{ws})_{ij} = \frac{1}{K}\sum_k P_{ik}^{sw}P_{kj}^{ws}, \tag{3}$$

where K means the number of total associations. Note that this probability is the average of the association probabilities, which produces the best results in our experiments. Other variations of this probability can be exploited and result in the extension of this work.

Angular margin-based loss. Incorporating the weight fusion strategy into AM loss has demonstrated robust classification, such as face recognition [41]. However, it is difficult and complex to estimate and seamlessly incorporate the weights into the models. In our approach, the estimation is simplified, and the association probability is measured, which indicates that the learned embeddings are coherent between the web and the well-labeled dataset. To this end, the association schema is treated as a weighting strategy for AM loss. To shed light on the idea of weighting AM loss, first, the standard Softmax loss is given:

$$\ell = -\frac{1}{T}\sum_{i=1}^{T}\log\frac{\mathrm{e}^{\boldsymbol{W}_{y_i}^{\mathrm{T}}\boldsymbol{x}_i+b_{y_i}}}{\sum_{j=1}^{C}\mathrm{e}^{\boldsymbol{W}_j^{\mathrm{T}}\boldsymbol{x}_i+b_j}}, \tag{4}$$

where b indicates the bias, $\boldsymbol{W}_j$ refers to the j-th column of weights $\boldsymbol{W}$ of the layers, $\boldsymbol{x}_i$ and y_i refer to the i-th image with the label y_i, and T and C refer to batch size and class number, respectively. In most AM losses, the bias $b_j = 0$ and $\| \boldsymbol{W}_j \| = 1$ are set, and then the target logit is obtained using

$$\boldsymbol{W}_j^{\mathrm{T}}\boldsymbol{x}_i = \|\boldsymbol{x}_i\|\cos\theta_{i,j}, \tag{5}$$

where $\theta_{i,j}$ means the angle between $\boldsymbol{x}_i$ and $\boldsymbol{W}_j$. Then, $\|\boldsymbol{x}_i\| = s$ is set as a fixed value, and the traditional Softmax loss can be represented as

$$\ell = -\frac{1}{T}\sum_{i=1}^{T}\log\frac{\mathrm{e}^{s\cos\theta_{i,y_i}}}{\sum_{j=1}^{C}\mathrm{e}^{s\cos\theta_{i,j}}}. \tag{6}$$

Lastly, the association probability p_k of the association schema as a weighted strategy is implemented to the following equation:

$$\ell_c = -\frac{1}{T}\sum_{k=1}^{T}\log\frac{\mathrm{e}^{p_k s\cos\theta_{k,y_k}}}{\sum_{j=1}^{C}\mathrm{e}^{p_k s\cos\theta_{k,j}}}. \tag{7}$$

This Softmax loss function is merged into the bridge schema. In our proposed model, the weighting fusion skills are leveraged to calculate the gradients of loss or estimate the instance weights of the current batch. A detailed pseudocode is presented in Algorithm 1. Note that this implementation is general and can be extended to any deep learning model and other frameworks based on current packages.

Bridge schema. Learning the parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_f, \boldsymbol{\theta}_c\}$ from both the web and well-labeled datasets based on a DNN, e.g., CNN, where $\boldsymbol{\theta}_f$ means the parameters of feature learning and $\boldsymbol{\theta}_c$ refers to the parameters of a classifier that supports the C-way fashion, in which C is the number of categories. Moreover, the influence of noisy labels is considered from web data because they would cause performance degradation due to training loss that may overfit the label noise. Thus, AM loss is used, which is presented above. To construct a connection between the web and well-labeled datasets, a bridge schema is developed, which optimizes the learning representation $\boldsymbol{\theta}_f$ via incorporating a domain classifier to maximize the loss ℓ_s. Therefore, the logistic function can be defined for the domain classifier as follows:

$$g(\boldsymbol{x}, \boldsymbol{\theta}_s, \boldsymbol{\theta}_f) = \frac{1}{1+\mathrm{e}^{-\{\boldsymbol{\theta}_s,\boldsymbol{\theta}_f\}^{\mathrm{T}}\boldsymbol{x}}}, \tag{8}$$

where $\boldsymbol{\theta}_s$ means the parameters of its source classification. Then, the formulation of the domain classification is defined, where its log-likelihood loss function can be represented as

$$\begin{aligned}\ell_s(I; g(\boldsymbol{x}, \boldsymbol{\theta}_s, \boldsymbol{\theta}_f)) = \sum_{i=1}^{m} & -I_i \log(g(\boldsymbol{x}, \boldsymbol{\theta}_s, \boldsymbol{\theta}_f)) \\ & - (1 - I_i)\log(1 - g(\boldsymbol{x}, \boldsymbol{\theta}_s, \boldsymbol{\theta}_f)),\end{aligned} \tag{9}$$

where I_i refers to the input images from a web dataset ($I_i = 0$) or a well-labeled dataset ($I_i = 1$) and m is the minibatch size. The main idea behind the corresponding loss is that it can reduce the gap between the web and well-labeled datasets. To go a step further, this means that our model can differentiate whether the images come from the web or from well-labeled sources. Specifically, if the model can recognize the image source, the loss ℓ_s will decrease. Otherwise, the loss will increase when the model has difficulties recognizing whether the images come from the web or a well-labeled dataset. The objective of our method is to minimize the joint loss function, as shown below:

$$\mathcal{L}_\theta(\boldsymbol{\theta}_f, \boldsymbol{\theta}_c, \boldsymbol{\theta}_s) = \ell_c - \lambda \ell_s. \tag{10}$$

Meta-learning. Meta-learning is the process of learning to learn and contains the metatraining and metatest phases. In the training process, a minibatch of data $(\boldsymbol{X}, \boldsymbol{Y})$, where $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}$ and the corresponding labels $\boldsymbol{Y} = \{y_1, \ldots, y_k\}$, are sampled from various web datasets. Multiple minibatches $\{\hat{\boldsymbol{Y}}_1, \ldots, \hat{\boldsymbol{Y}}_M\}$ are generated. Subsequently, these batch data are used to update $\boldsymbol{\theta}$ via gradient descent.

$$\boldsymbol{\theta}'_m = \boldsymbol{\theta} - \alpha \nabla_\theta \mathcal{L}_\theta(\boldsymbol{X}, \hat{\boldsymbol{Y}}_m, \boldsymbol{\theta}), \tag{11}$$

where α is the step size and $\mathcal{L}_\theta(\boldsymbol{X}, \hat{\boldsymbol{Y}}_m, \boldsymbol{\theta})$ means the combined loss defined in (10).

In the metatest phase, to adapt the multiple web applications, the meta-objective is to train $\boldsymbol{\theta}$ for each batch of data. For this purpose, we focus on the self-assembling method proposed by Tarvainen & Valpola [55] because it serves as one of the foundations of our approach. It builds a teacher model to enhance consistent predictions. Two networks are introduced: a student network and a teacher network. The weights from the teacher network are determined by the exponential moving average (EMA) of the weights from the student network. The student network is trained to produce predictions consistent with the teacher network. In our proposed method, the teacher network is employed during metatesting to train the student network, improving its ability to effectively handle label noise.

Specifically, the EMA method [56] is used to calculate the parameters of the teacher model. Given the parameters of the student model $\boldsymbol{\theta}$ and those of the teacher model $\hat{\boldsymbol{\theta}}$, the performance of the teacher model is superior to that of the student model. At each training step, the update step can be expressed as

$$\hat{\boldsymbol{\theta}} = \gamma \hat{\boldsymbol{\theta}} + (1 - \gamma)\boldsymbol{\theta}, \tag{12}$$

where γ refers to a smoothing coefficient hyperparameter. The model enforces a consistency loss $\mathcal{J}(\boldsymbol{\theta}'_m)$ that encourages the updated model to provide robust predictions using the teacher model, which learns from clean, well-labeled data. Kullback-Leibler (KL) divergence is introduced to calculate the difference between the teacher model $f(\boldsymbol{X}, \hat{\boldsymbol{\theta}})$ and Softmax predictions from the updated model $f(\boldsymbol{X}, \boldsymbol{\theta}'_m)$.

$$\begin{aligned}\mathcal{J}(\boldsymbol{\theta}'_m) &= \frac{1}{k}\sum_{i=1}^{k} D_{\mathrm{KL}}(f(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}) || f(\boldsymbol{x}_i, \boldsymbol{\theta}'_m)) \\ &= \frac{1}{k}\sum_{i=1}^{k} \mathbb{E}(\log(f(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}})) - \log(f(\boldsymbol{x}_i, \boldsymbol{\theta}'_m))).\end{aligned} \tag{13}$$

Because we have different web datasets, consistency loss can be minimized via the parameters $\{\boldsymbol{\theta}'_1, \ldots, \boldsymbol{\theta}'_M\}$ from M updated models. Therefore, the average of all losses is defined as meta-loss:

$$\begin{aligned}\mathcal{L}_{\mathrm{meta}}(\boldsymbol{\theta}) &= \frac{1}{M}\sum_{m=1}^{M} \mathcal{J}(\boldsymbol{\theta}'_m), \\ s &= \frac{1}{M}\sum_{m=1}^{M} \mathcal{J}(\boldsymbol{\theta} - \alpha \nabla_\theta \mathcal{L}_\theta(\boldsymbol{X}, \hat{\boldsymbol{Y}}_m, \boldsymbol{\theta})).\end{aligned} \tag{14}$$

Algorithm 2 Noise-tolerant method based on meta-learning

```
Initialize: Randomly initialize $\boldsymbol{\theta}$, teacher model $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}$;
repeat
    Sample some $k$ sizes of batch data $(\boldsymbol{X}, \boldsymbol{Y})$ from $\mathcal{D}$;
    for $m = 1 : M$ do
        Update the parameters with gradient descent:
        $\boldsymbol{\theta}' = \boldsymbol{\theta} - \alpha \nabla_\theta \mathcal{L}_c(\boldsymbol{X}, \boldsymbol{Y}_m, \boldsymbol{\theta})$;
        Compute the consistency loss with the teacher model:
        $\mathcal{J}(\boldsymbol{\theta}'_m) = \frac{1}{k} \sum_{i=1}^{k} D_{\mathrm{KL}}(f(\hat{\boldsymbol{x}}_i, \hat{\boldsymbol{\theta}}) || f(\boldsymbol{x}_i, \boldsymbol{\theta}'_m))$;
    end for
    Compute $\mathcal{L}_{\mathrm{meta}}(\boldsymbol{\theta}) = \frac{1}{M} \sum_{m=1}^{M} \mathcal{J}(\boldsymbol{\theta}'_m)$;
    Update by meta-learning $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla \mathcal{L}_{\mathrm{meta}}(\boldsymbol{\theta})$;
    Compute the classification loss $\mathcal{L}_\theta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\theta})$;
    Update $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta \nabla \mathcal{L}_\theta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\theta})$;
    Update the teacher model: $\hat{\boldsymbol{\theta}} = \gamma \hat{\boldsymbol{\theta}} + (1 - \gamma) \boldsymbol{\theta}$;
until Stopping criteria are met.
```

Stochastic gradient descent (SGD) is performed to minimize meta-loss. The parameters θ of the student model can be updated as follows:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla \mathcal{L}_{\mathrm{meta}}(\boldsymbol{\theta}), \tag{15}$$

where η is the meta-learning rate. Taking the update from the meta-learning, we can use the original minibatches $(\boldsymbol{X}, \boldsymbol{Y})$ to optimize the classification loss via SGD:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta \nabla \mathcal{L}_\theta(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\theta}), \tag{16}$$

where β is the learning rate of the model. Algorithm 2 is shown above.

For the metagradient $\nabla \mathcal{L}_{\mathrm{meta}}(\boldsymbol{\theta})$, it involves a gradient via gradient. In this case, the second-order derivatives are needed to be calculated regarding $\boldsymbol{\theta}$. From the experimental results, we can reduce the computation complexity via the approximation method. As an example of this method [17], the first-order approximation is employed by omitting the second-order derivatives. Thus, the term $\alpha \nabla_\theta \mathcal{L}_\theta(\boldsymbol{X}, \hat{\boldsymbol{Y}}_m, \boldsymbol{\theta})$ in (14) can be treated as a constant for the first-order approximation. Finally, the update $\boldsymbol{\theta} - \alpha \nabla_\theta \mathcal{L}_\theta(\boldsymbol{X}, \hat{\boldsymbol{Y}}_m, \boldsymbol{\theta})$ can be treated as data-dependent noise, and it will regularize the model network during training.

Extensions. There are two potential extensions that we would like to try. The first is the ensemble technique, which can be incorporated into the student and teacher model to minimize the loss and improve the consistency loss, making it more effective.

$$\mathcal{J}(\boldsymbol{\theta}'_m) = \frac{1}{k} \sum_{i=1}^{k} D_{\mathrm{KL}}(c || f(\boldsymbol{x}_i, \boldsymbol{\theta}'_m)), \tag{17}$$

where c is the ensemble teacher (mentor) models. The second is active learning or label editing techniques, which mean to choose the wrong labels based on the loss function. For instance, we can remove the samples when the model reflects low confidence in its corresponding label. Then, the sampled batch data D' will have high quality, and the filtered training set is defined as follows:

$$\mathcal{D}' = (\boldsymbol{x}_i, y_i) \in \mathcal{D} | y_i \cdot f(\boldsymbol{x}_i, \boldsymbol{\theta}^*). \tag{18}$$

4 Experimental study

First, benchmark and large-scale datasets are introduced, which are commonly used in fine-grained visual tasks. Then, many baselines are compared and empirical results are shown for some parameters that influence our proposed method, including analyses of the effects of some hyperparameters and representation coherence. Moreover, the effect of the weighting strategy and learning with auxiliary web data is examined. Finally, the effect on different network structures is studied, and an ablation analysis of the model is performed.

4.1 Datasets and evaluation setup

The well-labeled datasets consist of four typical datasets used in the fine-grained classification task. The CUB-200-2011 dataset has 11788 images of 200 bird species. The Stanford Dogs dataset has 20580

Table 1 Characteristics of datasets

Dataset	Object	#Category	#Training	#Testing
CUB-200-2011	Birds	200	5994	5794
Stanford Dogs	Dogs	120	12000	8580
FGVC-Aircraft	Aircraft	100	6667	3333
MIT Indoor 67	Indoors	67	12496	3124
iNat	Species	5089	579184	95986

annotated images of dogs belonging to 120 species. The FGVC-Aircraft dataset has 10000 images of aircraft with 100 categories and is organized in a three-level hierarchy. The MIT Indoor 67 dataset has 67 indoor categories and 15620 images [57]. In our training, three extensive web datasets introduced by [18] are used, and these datasets are built by retrieving images from Google, Flickr, and Twitter through keyword searches, where the keywords align with the category labels found in public datasets. Then, images relevant to the respective classes are selected from the search results as web data. In practice, it is difficult for some datasets to crawl the web data because the names of scenes are very similar to the category names. For instance, the MIT Indoor 67 datasets are labeled with an additional "indoor" or "outdoor" category, which makes it very difficult to correct the web data based on its semantic meanings. For these four benchmark fine-grained classification datasets, 11788 bird images, 22000 dog images, 10000 aircraft images, and 15620 indoor scene images were separately collected. The test data certainly come from the original standard data in the following experiments.

For large-scale fine-grained visual recognition, the iNaturalist 2017 (iNat) dataset is employed in the visual recognition task. The iNat dataset has 675170 training and validation images from 5089 natural fine-grained categories. These categories belong to 13 supercategories, including Plant's categories, Insect, Bird, and Mammal. Using the same setting as the benchmark datasets, the same amount of data is collected as the training data. Table 1 summarizes the characteristics of the datasets.

Compared algorithms. Classification comparisons are conducted with state-of-the-art methods on four benchmark and one large-scale fine-grained datasets. Also, different CNN models are tested using our proposed method, such as AlexNet, CaffeNet, VggNet, and ResNet50. In the literature, a pretrained model with a finetuning strategy has shown promising results. Therefore, we also adopted these tricks in our experiments to train the initial models on both web and standard datasets.

To compare algorithms, we can classify the three classes from these baselines, i.e., noisy label learning baselines, webly-supervised learning baselines, and weakly supervised learning baselines. Among these baselines, webly-supervised learning baselines can use web images, whereas other baselines cannot incorporate auxiliary classes. In addition, to better augment the web data, the web datasets are processed according to methods and tools that can align the distribution of categories. For noisy label learning baselines, we use the methods of designed robust loss functions [22, 58], adversarial training [19, 49, 59], and meta-learning [23, 42]. For web-supervised learning baselines, we compared the following approaches: [4, 13, 16, 17, 34, 60]. The training data from the web source are treated as the source domain, while the test data from the web source are treated as the target domain. For weakly supervised learning baselines, we use the methods [21, 32–34, 36, 39, 60], and some of which cannot be directly adapted into our setting. We utilize several robust loss or weighting strategies and meta-learning frameworks.

For fair comparison, multimodal information is not used. We compensate the networks of dynamic MLP with various current fusion strategies, i.e., concatenation, addition, and multiplication, by augmenting the web data. Regarding the experiments performed on benchmark datasets, the dynamic MLP-C architecture is used, with fixed channel dimension and stage number set to the optimal configurations (d= 256, h= 64, N= 2). This selection remains consistent across all our experiments.

Evaluation setup. Our models are based on the TensorFlow framework and trained on NVIDIA TITAN X GPUs. ReLU is adopted into our architecture, and different network structures are used as our fully connected (FC) layer. Specifically, AlexNet, CaffeNet, and VggNet-16 hold for 4096-4096-2 hidden units, while we add an additional FC layer with two hidden units for ResNet50. We set different sizes of the minibatch for CaffeNet ($m = 64$), AlexNet ($m = 64$), VggNet ($m = 32$), and ResNet50 ($m = 12$). We set the start of the learning rate as 0.001 for the bird, dog, and aircraft datasets and 0.0001 for the indoor scene dataset. We reduced the learning rate after 25000 iterations of training. The parameters of our proposed method will be discussed in Subsection 4.5. We also set the maximum iteration number

Table 2 Comparison with various baselines trained with symmetric label noise ratio 0.1$^{a)}$

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	85.08	Krause et al. 2016 [22]	80.18	Fu et al. 2017 [36]	92.01	Dixit et al. 2015 [61]	72.03
Zheng et al. 2017 [62]	85.97	Fu et al. 2017 [36]	85.27	Zheng et al. 2017 [62]	92.03	Guo et al. 2016 [63]	83.02
Dubey et al. 2018 [29]	84.67	Dubey et al. 2018 [39]	83.53	Dubey et al. 2018 [39]	91.96	Herranz et al. 2016 [57]	80.28
Xu et al. 2018 [16]	84.17	Niu et al. 2018 [13]	83.62	Wang et al. 2018 [37]	92.10	Lin et al. 2017 [10]	77.93
Niu et al. 2018 [13]	84.23	Sun et al. 2019 [17]	84.66	Sun et al. 2019 [17]	92.89	Sun et al. 2019 [17]	78.87
Zheng et al. 2019 [64]	78.32	Hu et al. 2019 [34]	86.08	Hu et al. 2019 [34]	91.67	Hu et al. 2019 [34]	83.26
Liu et al. 2020 [58]	87.16	Liu et al. 2020 [58]	88.47	Liu et al. 2020 [58]	93.09	Liu et al. 2020 [58]	84.38
Sun et al. 2021 [4]	87.39	Sun et al. 2021 [4]	83.88	Sun et al. 2021 [4]	92.41	Sun et al. 2021 [4]	84.89
Yang et al. 2022 [65]	87.92	Yang et al. 2022 [65]	89.18	Yang et al. 2022 [65]	92.75	Yang et al. 2022 [65]	84.98
Li et al. 2023 [66]	87.89	Li et al. 2023 [66]	88.73	Li et al. 2023 [66]	92.43	Li et al. 2023 [66]	85.02
Ours	**88.23**	**Ours**	**90.15**	**Ours**	**93.36**	**Ours**	**86.25**

a) Best results are marked in bold.

Table 3 Comparison with various baselines trained with symmetric label noise ratio 0.3$^{a)}$

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	82.15	Krause et al. 2016 [22]	77.36	Fu et al. 2017 [36]	89.55	Dixit et al. 2015 [61]	69.68
Zheng et al. 2017 [62]	84.02	Fu et al. 2017 [36]	82.88	Zheng et al. 2017 [62]	89.47	Guo et al. 2016 [63]	79.73
Dubey et al. 2018 [29]	81.95	Dubey et al. 2018 [39]	82.77	Dubey et al. 2018 [39]	90.02	Herranz et al. 2016 [57]	78.16
Xu et al. 2018 [16]	81.86	Niu et al. 2018 [13]	81.27	Wang et al. 2018 [37]	89.26	Lin et al. 2017 [10]	76.42
Niu et al. 2018 [13]	81.62	Sun et al. 2019 [17]	81.93	Sun et al. 2019 [17]	90.68	Sun et al. 2019 [17]	77.05
Zheng et al. 2019 [64]	76.04	Hu et al. 2019 [34]	83.78	Hu et al. 2019 [34]	88.79	Hu et al. 2019 [34]	80.94
Liu et al. 2020 [58]	84.27	Liu et al. 2020 [58]	86.33	Liu et al. 2020 [58]	91.08	Liu et al. 2020 [58]	81.86
Sun et al. 2021 [4]	85.11	Sun et al. 2021 [4]	82.04	Sun et al. 2021 [4]	90.37	Sun et al. 2021 [4]	82.26
Yang et al. 2022 [65]	86.90	Yang et al. 2022 [65]	86.83	Yang et al. 2022 [65]	90.75	Yang et al. 2022 [65]	82.89
Li et al. 2023 [66]	86.54	Li et al. 2023 [66]	86.16	Li et al. 2023 [66]	90.47	Li et al. 2023 [66]	82.72
Ours	**86.98**	**Ours**	**88.05**	**Ours**	**91.08**	**Ours**	**84.17**

a) Best results are marked in bold.

to 220000 because this setting would result in the convergence of the model. We use ResNet50 as the default setting.

Two types of label noise are added: symmetric and asymmetric. Symmetric label noise employs typical uniform noise scenarios, and a one-hot vector is randomly injected to substitute the ground-truth label of a sample with a probability of r. The asymmetric label noise incorporates the notion that each label has a chance of being flipped to the next class, with a probability of r. The aim of this design is to mimic the structural characteristics of the genuine errors observed in related classes, for instance, dog→cat, horse→deer, automobile→truck, aircraft→bird. The parameterization of label transitions, denoted by $r \in [0, 1]$, assigns a probability of $1{-}r$ to the true class and a probability of r to the wrong class. The experiments were repeated 10 times. The average classification performance is recorded with different ratios.

4.2 Comparing with various baselines

The proposed method is compared with other state-of-the-art approaches on four public datasets. Note that the comparison methods utilize the same auxiliary web data. Tables 2–7 [4, 13, 16, 17, 22, 29, 34, 36, 37, 39, 57, 58, 61–66] list the results for symmetric and asymmetric label noise, demonstrating that the proposed method has better performance than the compared approaches. In Table 2, the methods of [17] and [4] also incorporate the web data, but their algorithms may result in performance degradation and overfit to the label noise. Our two-level noise-tolerant strategy plays an important role in enhancing the model's performance. On the one hand, the association mechanism makes our input data robust to noisy labels. On the other hand, incorporating the association mechanism into AM loss makes the proposed algorithm perform noise-tolerant classification. Compared with the method of [34], the methods of [58] and [5] propose the data augmentation network and contain the step of generating attention maps to represent the object's discriminative parts. However, poor accuracy in the first step would result in poor

Table 4 Comparison with various baselines trained with symmetric label noise ratio 0.5[a)]

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	78.25	Krause et al. 2016 [22]	72.40	Fu et al. 2017 [36]	86.52	Dixit et al. 2015 [61]	65.44
Zheng et al. 2017 [62]	80.15	Fu et al. 2017 [36]	78.56	Zheng et al. 2017 [62]	86.02	Guo et al. 2016 [63]	76.87
Dubey et al. 2018 [29]	77.62	Dubey et al. 2018 [39]	77.92	Dubey et al. 2018 [39]	86.02	Herranz et al. 2016 [57]	74.38
Xu et al. 2018 [16]	78.06	Niu et al. 2018 [13]	76.78	Wang et al. 2018 [37]	86.53	Lin et al. 2017 [10]	73.34
Niu et al. 2018 [13]	78.27	Sun et al. 2019 [17]	78.05	Sun et al. 2019 [17]	86.77	Sun et al. 2019 [17]	74.26
Zheng et al. 2019 [64]	72.93	Hu et al. 2019 [34]	77.80	Hu et al. 2019 [34]	87.63	Hu et al. 2019 [34]	75.24
Liu et al. 2020 [58]	80.82	Liu et al. 2020 [58]	82.76	Liu et al. 2020 [58]	87.35	Liu et al. 2020 [58]	78.61
Sun et al. 2021 [4]	81.92	Sun et al. 2021 [4]	80.75	Sun et al. 2021 [4]	88.11	Sun et al. 2021 [4]	79.82
Yang et al. 2022 [65]	86.15	Yang et al. 2022 [65]	86.22	Yang et al. 2022 [65]	89.68	Yang et al. 2022 [65]	82.03
Li et al. 2023 [66]	85.93	Li et al. 2023 [66]	85.27	Li et al. 2023 [66]	88.86	Li et al. 2023 [66]	82.08
Ours	**87.10**	**Ours**	**87.48**	**Ours**	**90.85**	**Ours**	**84.71**

a) Best results are marked in bold.

Table 5 Comparison with various baselines trained with asymmetric label noise ratio 0.1[a)]

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	85.12	Krause et al. 2016 [22]	80.25	Fu et al. 2017 [36]	92.26	Dixit et al. 2015 [61]	72.70
Zheng et al. 2017 [62]	86.31	Fu et al. 2017 [36]	85.93	Zheng et al. 2017 [62]	92.54	Guo et al. 2016 [63]	83.62
Dubey et al. 2018 [29]	84.92	Dubey et al. 2018 [39]	85.68	Dubey et al. 2018 [39]	93.02	Herranz et al. 2016 [57]	81.02
Xu et al. 2018 [16]	84.78	Niu et al. 2018 [13]	83.82	Wang et al. 2018 [37]	92.76	Lin et al. 2017 [10]	78.92
Niu et al. 2018 [13]	84.52	Sun et al. 2019 [17]	84.97	Sun et al. 2019 [17]	93.60	Sun et al. 2019 [17]	79.59
Zheng et al. 2019 [64]	78.93	Hu et al. 2019 [34]	86.92	Hu et al. 2019 [34]	92.15	Hu et al. 2019 [34]	84.38
Liu et al. 2020 [58]	87.65	Liu et al. 2020 [58]	89.15	Liu et al. 2020 [58]	94.17	Liu et al. 2020 [58]	85.27
Sun et al. 2021 [4]	88.27	Sun et al. 2021 [4]	85.02	Sun et al. 2021 [4]	93.14	Sun et al. 2021 [4]	86.08
Yang et al. 2022 [65]	87.95	Yang et al. 2022 [65]	89.40	Yang et al. 2022 [65]	92.89	Yang et al. 2022 [65]	85.16
Li et al. 2023 [66]	87.78	Li et al. 2023 [66]	89.12	Li et al. 2023 [66]	92.83	Li et al. 2023 [66]	84.97
Ours	**88.50**	**Ours**	**90.37**	**Ours**	**93.75**	**Ours**	**87.00**

a) Best results are marked in bold.

Table 6 Comparison with various baselines trained with asymmetric label noise ratio 0.3[a)]

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	82.69	Krause et al. 2016 [22]	77.89	Fu et al. 2017 [36]	90.02	Dixit et al. 2015 [61]	70.18
Zheng et al. 2017 [62]	84.05	Fu et al. 2017 [36]	83.38	Zheng et al. 2017 [62]	89.87	Guo et al. 2016 [63]	80.19
Dubey et al. 2018 [29]	82.29	Dubey et al. 2018 [39]	83.15	Dubey et al. 2018 [39]	90.47	Herranz et al. 2016 [57]	78.56
Xu et al. 2018 [16]	82.25	Niu et al. 2018 [13]	81.62	Wang et al. 2018 [37]	90.04	Lin et al. 2017 [10]	77.12
Niu et al. 2018 [13]	81.95	Sun et al. 2019 [17]	82.25	Sun et al. 2019 [17]	91.20	Sun et al. 2019 [17]	77.18
Zheng et al. 2019 [64]	76.66	Hu et al. 2019 [34]	84.37	Hu et al. 2019 [34]	90.06	Hu et al. 2019 [34]	81.75
Liu et al. 2020 [58]	84.88	Liu et al. 2020 [58]	86.78	Liu et al. 2020 [58]	92.05	Liu et al. 2020 [58]	83.14
Sun et al. 2021 [4]	86.37	Sun et al. 2021 [4]	82.76	Sun et al. 2021 [4]	91.92	Sun et al. 2021 [4]	83.11
Yang et al. 2022 [65]	86.93	Yang et al. 2022 [65]	86.90	Yang et al. 2022 [65]	91.02	Yang et al. 2022 [65]	83.05
Li et al. 2023 [66]	86.72	cLi et al. 2023 [66]	86.10	Li et al. 2023 [66]	90.15	Li et al. 2023 [66]	82.57
Ours	**87.85**	**Ours**	**88.97**	**Ours**	**92.26**	**Ours**	**85.39**

a) Best results are marked in bold.

performance in classification. Our meta-learning framework reduces the overfitting of models and learns the network parameters to be noise-tolerant. The same phenomenon as with the other symmetric label noise is shown in Tables 3 and 4. The results for these datasets offer the overall performance of our method versus the baselines. In the below experiments, the accuracy is recorded with different noise ratios.

Tables 5–7 present the comparison results with asymmetric label noise (noise ratio $r = 0.1$, 0.3, 0.5). The performance of our proposed method also exhibits competitive results compared with the baselines. Both the methods of [19,30] depend on a predefined noise transition matrix with ground truth. Moreover,

Table 7 Comparison with various baselines trained with asymmetric label noise ratio 0.5[a)]

CUB-200-2011		Stanford Dogs		FGVC-Aircraft		MIT Indoor 67	
Method	Acc (%)	Method	Acc (%)	Method	Acc (%)	Method	Acc (%)
Fu et al. 2017 [36]	78.17	Krause et al. 2016 [22]	72.36	Fu et al. 2017 [36]	86.12	Dixit et al. 2015 [61]	65.40
Zheng et al. 2017 [62]	80.06	Fu et al. 2017 [36]	78.45	Zheng et al. 2017 [62]	85.24	Guo et al. 2016 [63]	76.33
Dubey et al. 2018 [29]	77.61	Dubey et al. 2018 [39]	77.28	Dubey et al. 2018 [39]	85.39	Herranz et al. 2016 [57]	73.82
Xu et al. 2018 [16]	77.45	Niu et al. 2018 [13]	76.06	Wang et al. 2018 [37]	86.01	Lin et al. 2017 [10]	72.26
Niu et al. 2018 [13]	77.82	Sun et al. 2019 [17]	76.93	Sun et al. 2019 [17]	85.68	Sun et al. 2019 [17]	72.94
Zheng et al. 2019 [64]	72.29	Hu et al. 2019 [34]	77.42	Hu et al. 2019 [34]	86.04	Hu et al. 2019 [34]	73.96
Liu et al. 2020 [58]	80.10	cLiu et al. 2020 [58]	81.25	Liu et al. 2020 [58]	86.72	Liu et al. 2020 [58]	77.17
Sun et al. 2021 [4]	81.26	Sun et al. 2021 [4]	79.02	Sun et al. 2021 [4]	86.93	Sun et al. 2021 [4]	78.36
Yang et al. 2022 [65]	86.17	Yang et al. 2022 [65]	86.25	Yang et al. 2022 [65]	89.70	Yang et al. 2022 [65]	82.32
Li et al. 2023 [66]	85.88	Li et al. 2023 [66]	85.08	Li et al. 2023 [66]	88.75	Li et al. 2023 [66]	82.01
Ours	**86.82**	**Ours**	**87.45**	**Ours**	**90.72**	**Ours**	**84.02**

a) Best results are marked in bold.

the methods take the class distribution into the optimization method, which has a strong assumption. The scheme we proposed is general and can be incorporated into any deep network architecture and has the best performance under different noise scenarios. By comparing with the state-of-the-art methods, our method remains competitive. Furthermore, the teacher model of our proposed method significantly outperforms previous methods after three iterations.

4.3 Large-scale fine-grained recognition

To confirm the proposed learning schema on large-scale fine-grained recognition, several experiments were conducted on the iNat dataset. To attain a better result, the idea of finetuning from ImageNet pretrained networks is used. The iNat is a large-scale dataset with real-world noisy labels, which were added to our web data. For fair comparison, a random one-hot vector is used to inject label noise with a probability of r into the instances. Note that the noise level is greater than the injection ratio because the web data involve some noise. Motivated by their implementation settings, we use them during the training and make comparisons with previous methods. The base model is ResNet50 with random initialization. Furthermore, a human verified train subset is used as our probe data, which is included in each dataset. The comparison results in different ratios of symmetric and asymmetric noise ratios are shown in Tables 8 and 9. Our method attains 63.17% and 61.20% with symmetric and asymmetric noise ratios of 0.5, respectively. From Tables 8 and 9, it can be observed that using more web data realizes better performance on the iNat dataset.

The baselines include [22], which uses multiple crops and a larger web dataset with additional categories. The methods of [5, 34] describe the attention feature map as the discriminative components of objects. However, the separate steps of the current method may not be the perfect solution to address this issue because the aim of the method is not direct. Moreover, mistakes made in the previous step will affect the results of the next step. Our method does not need that, which makes our method more general. Moreover, compared with [58], our proposed method can achieve an improvement of +2.52% (3.42%) of accuracy with 0.5 symmetric (asymmetric) noise ratio, demonstrating that our proposed model is promising.

4.4 Effect of the AM loss function

As we clarified before, incorporating the association mechanism into AM losses could lead to a robust classification. The tricks have been applied in other noisy scenarios, such as LFW, CFP, and AgeDB. For comparison, the method is also applied with Softmax loss under the same noise ratio. In Table 10, the performance of using the AM loss function is better than that of simply using the Softmax loss function. This enhancement is mainly caused by incorporating weight fusion into AM loss. Specifically, two noise-tolerant strategies play an important role in ensuring robustness from the embeddings of input data to classification. These experiments show that our proposed method can realize robust classification.

Table 8 Classification accuracy (%) on a large-scale fine-grained dataset for different methods trained with different symmetric noise ratios[a)]

Method	Noise ratio				
	0.1	0.2	0.3	0.4	0.5
Zhang et al. 2016 [60]	56.94	56.29	54.83	51.65	50.02
Krause et al. 2016 [22]	58.17	57.38	55.26	55.02	53.24
Fu et al. 2017 [36]	57.92	57.18	57.03	55.84	54.17
Dubey et al. 2018 [39]	60.78	58.92	58.10	56.73	55.06
Niu et al. 2018 [13]	63.35	61.78	58.90	57.22	55.31
Sun et al. 2019 [17]	63.79	62.10	60.69	58.07	55.96
Liu et al. 2020 [58]	65.16	65.02	63.18	61.90	60.06
Sun et al. 2021 [4]	66.80	65.43	63.78	61.92	60.65
Yang et al. 2022 [65]	67.21	66.38	65.74	63.72	61.95
Li et al. 2023 [66]	67.16	66.33	65.26	63.09	61.72
Ours	**68.69**	**68.22**	**67.41**	**65.89**	**63.17**

a) Best results are marked in bold.

Table 9 Classification accuracy (%) on a large-scale fine-grained dataset for different methods trained with different asymmetric noise ratios[a)]

Method	Noise ratio				
	0.1	0.2	0.3	0.4	0.5
Zhang et al. 2016 [60]	57.03	56.60	54.98	52.01	50.22
Krause et al. 2016 [22]	57.92	57.63	56.87	55.26	53.81
Fu et al. 2017 [36]	58.11	57.08	55.26	54.10	52.19
Dubey et al. 2018 [39]	60.75	59.06	57.34	55.82	54.08
Niu et al. 2018 [13]	62.33	60.79	58.12	56.30	54.27
Sun et al. 2019 [17]	63.09	61.22	59.45	57.06	55.11
Liu et al. 2020 [58]	65.01	62.70	60.08	58.24	55.83
Sun et al. 2021 [4]	66.84	65.03	62.71	60.52	57.78
Yang et al. 2022 [65]	67.76	66.92	64.25	63.02	60.08
Li et al. 2023 [66]	67.43	66.84	64.09	62.83	60.02
Ours	**68.80**	**68.17**	**66.96**	**64.12**	**61.20**

a) Best results are marked in bold.

Table 10 Comparisons of purely Softmax loss (Simple) and AM loss function with symmetric (S) and asymmetric (A) labels (Noise)[a)]

Loss	Noise	Dataset				
	ratio	CUB-200-2011	Stanford Dogs	FGVC-Aircraft	MIT Indoor 67	iNat
Simple	0.2	87.42	88.92	93.05	85.63	66.79
Simple	0.4	85.86	87.89	91.24	84.06	62.03
Simple	0.6	84.01	86.02	88.78	81.88	56.48
Noise (A)	0.2	**88.59**	**90.17**	**93.89**	**87.02**	**68.22**
Noise (A)	0.4	**87.98**	**89.96**	**92.76**	**85.87**	**65.89**
Noise (A)	0.6	**86.73**	**88.15**	**91.15**	**84.09**	**60.82**
Noise (S)	0.2	**88.07**	**89.01**	**93.45**	**86.62**	**66.35**
Noise (S)	0.4	**87.01**	**88.73**	**91.89**	**85.05**	**65.01**
Noise (S)	0.6	**85.22**	**86.94**	**90.08**	**82.96**	**60.23**

a) Best results are marked in bold.

4.5 Effect of the parameters

How hyperparameters λ, M, and the ratio of the training set affect the performance of our proposed algorithm are investigated. A detailed analysis is described below.

Impact of λ values. The λ trades off two losses, and the findings are illustrated in Figure 2(a). It is found that the classification accuracy undergoes some oscillations as λ increases. Using small λ indicates that the source image data plays a more important role. When we set $\lambda = 0$, the performance of the algorithms was worse on the FGVC-Aircraft and MIT Indoor 67 datasets. On the other hand, using

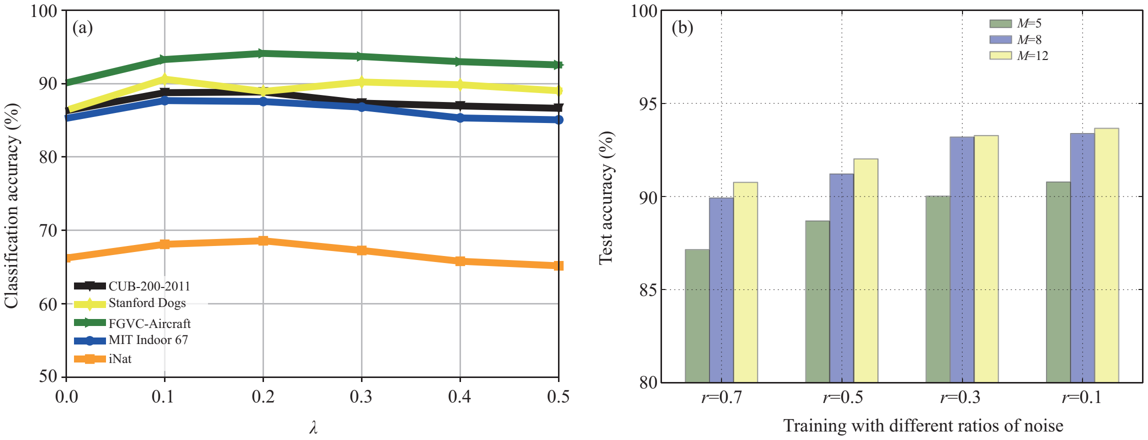

Figure 2 (Color online) Effect of parameters (a) λ and (b) M.

Table 11 Comparison with different ratios of training set, where -Web is mixing the web dataset into training data and -Ours is learning the representation coherence[a)]

Dataset	Ratio of training set				
	0.1	0.2	0.3	0.4	0.5
CUB-200-2011-Web	68.25	74.01	77.24	78.16	80.24
CUB-200-2011-Ours	**70.93**	**75.22**	**79.89**	**82.37**	**84.68**
Stanford Dogs-Web	67.46	72.79	75.10	76.87	79.35
Stanford Dogs-Ours	**69.82**	**74.05**	**79.37**	**82.39**	**85.86**
FGVC-Aircraft-Web	70.01	74.19	76.39	79.20	82.76
FGVC-Aircraft-Ours	**72.60**	**77.32**	**82.36**	**86.15**	**88.97**
MIT Indoor 67-Web	65.04	70.54	73.08	76.55	78.92
MIT Indoor 67-Ours	**66.17**	**71.10**	**75.09**	**79.26**	**82.03**
iNat-Web	51.02	52.89	54.85	55.92	57.40
iNat-Ours	**51.37**	**54.00**	**55.79**	**58.03**	**59.84**

a) Best results are marked in bold.

larger λ is not good for fine-grained classification. It probably suffers from small intraclass variations. The general trend is that the performance is good when we set λ between 0.1 and 0.3, while the performance of the method generally decreases as λ is greater than 0.3. For example, the performance is the highest in two cases when we set λ as 0.1. This λ parameter study provides some guidance for comparison experiments.

Impact of different ratios of training set. To solve the issue of the gap between well-labeled and web datasets, experiments on all four datasets are done with different ratios of training set. Better performance can be obtained by incorporating web data into our experiments. Table 11 lists the performance improvements at different noise ratios. The traditional approaches simply add the auxiliary data into the training data to enhance the model performance, and the improvements are not superior to our method, which can learn the representation coherence between the web and well-labeled data, because the web data may involve noise, especially for fine-grained domains, and it would affect the performance of models if we do not carefully deal with noisy labels. The experimental findings show that our approach can reduce the influence of the gap between well-labeled and web datasets.

Impact of M values. M is the number of minibatches $\{(\boldsymbol{X}, \boldsymbol{Y}_m)\}$ that are sampled for each minibatch $\{(\boldsymbol{X}, \boldsymbol{Y})\}$ of the training examples. We show the test accuracy on aircraft with $M = 5, 8, 12$ trained using labels with different noise ratios in Figure 2(b). The general phenomenon is that the accuracy of the model is enhanced as the number of M increases. Note that it is more significant as we increase the number of M from 5 to 8. However, from 8 to 12, the boosted results are not evident. Therefore, to balance the computation resource and generalization ability of the model, we set the number of M as 12 in our experiments.

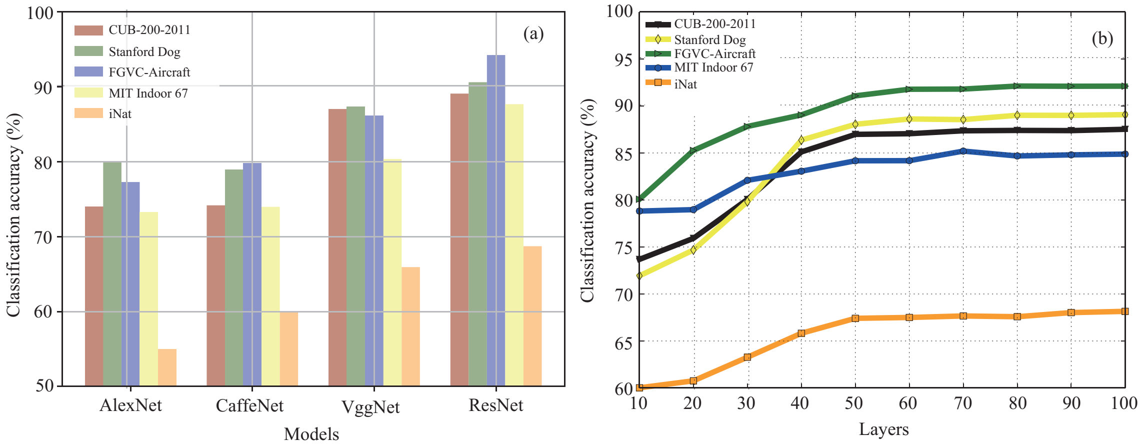

Figure 3 (Color online) (a) Scalability and (b) generalizability of our proposed model.

4.6 Scalability of our approach

In this subsection, given our focus on examining the scalability of our proposed method on the visual recognition task, our main concern here is whether the above noise-tolerant schemes could result in materially different scaling behaviors of our proposed method on the benchmark datasets. To address this problem, our method is assessed on different models, and the results are reported for the four models (AlexNet, CaffeNet, VggNet, and ResNet). The reason is that with good generalization of the model, the performance of our approach will be enhanced. The corresponding classification performance on the five benchmark datasets is plotted in Figure 3(a). These experimental results show that our approach has good scalability and can be further incorporated into other state-of-the-art models.

4.7 Learning with deeper layers and more data

The effect of the layer size of the model is attempted to be explored. The depths of the layers of the models are varied from 10 to 100 with a 10-layer interval. As the layers of the model increase, more data should be fed into the model. We then scaled up the training data using auxiliary web data. We expand the data by increasing the auxiliary web from 1 times the amount of basic training data to 10 times the amount of training data with an interval of 1 times the amount. Experiments were conducted on our five benchmark datasets, and the evaluation settings were the same as before. From Figure 3(b), it can be observed that the performance keeps improving as the model goes deeper. However, it converges when the number of layers of the model is around 60, attaining over 91.78% classification accuracy on the FGVA-Aircraft dataset. A similar convergence phenomenon of the model can be observed in the other four datasets. Thus, these experimental results indicate that the proposed model is stable from shallow to deep. Finally, considering the computation resources and massive parameters of the deep model, we should strike a balance between the computation and efficiency of the model.

4.8 Learning with auxiliary web data

Figure 4 shows the examples (left) correctly identified by our proposed method using noisy web data (right). Because the auxiliary web images sampled by the proposed method could cover the characteristics of all fine-grained categories, the bridge schema of the trained model could encourage the learned embeddings to be coherent between well-labeled and web data. The noise-tolerant strategies and meta-learning framework also play crucial roles in producing promising results. In addition, the updated model (student model), with the aid of the teacher model, is constructed by a self-ensembling method and produces consistent predictions. Thus, these results show that harnessing the noisy web data and incorporating the weighting strategy into AM loss are effective ways to improve the robustness of models.

4.9 Ablation analysis

To fully investigate our method, Table 12 provides a detailed ablation analysis of the different configurations of the key components. The individual objective components and their importance in the

Figure 4 (Color online) Our proposed method can differentiate images (left) using noisy web data (right) on fine-grained dog categories.

Table 12 Ablation study on the CUB-200-2011 dataset with noise ratio of 0.4[a)]

Association schema	Weighted AM loss	Bridge schema	Meta-learning	Accuracy (%)
–	–	–	–	72.84
✓	–	–	–	77.35
–	✓	–	–	77.37
–	–	✓	–	77.42
–	–	–	✓	77.36
✓	✓	–	–	78.40
✓	–	✓	–	79.97
✓	–	–	✓	81.65
–	✓	✓	–	81.46
–	✓	–	✓	81.22
–	–	✓	✓	82.74
✓	✓	✓	–	83.98
✓	✓	–	✓	84.31
✓	–	✓	✓	85.67
–	✓	✓	✓	84.82
✓	✓	✓	✓	**87.95**

a) ✓/– indicates the corresponding component is enabled/disabled. Best results are marked in bold.

CUB-200-2011 dataset are studied. ResNet50 is selected as the module. Based on our empirical observation, the association schema plays a significant role in preventing neural networks from overfitting to samples with incorrect labels. The model has an accuracy of 87.95% with the enabled association schema, while it has an accuracy of 84.82% without using the association schema components. Also, by incorporating the bridge schema, the method yields promising results as it can encourage the learned embeddings to be coherent between well-labeled and web data, thereby reducing the influence of the gap between well-labeled and web data. Compared with the method without using the bridge schema, our method offers significantly better results (+3.64%). In addition, using the weighted AM loss realizes 2.28% better performance than the baseline without using it. Furthermore, by developing the meta-learning schema, the method yields significantly better results (+3.97%). Finally, using all four components results in a

15.11% improvement. These experimental findings exhibit the importance of each of the four components. Similar experimental results of ablation analysis can be found on the other three benchmark and large-scale datasets.

5 Discussions

Learning from web data can help in constructing robust and generalizable models for generic image classification. By incorporating web data, models can learn from a broader range of visual variations and better adapt to real-world scenarios. This can aid in addressing the limitations of conventional datasets that may not cover the full diversity of visual appearances.

Moreover, there are many promising directions for the relationship between fine-grained visual recognition and learning from web data for generic image classification. One direction is to examine effective methods for mining and curating high-quality web data specifically tailored to fine-grained recognition tasks. Additionally, designing techniques to mitigate noise and label ambiguities in web data will be critical to guarantee reliable training results.

Another direction is to examine how to effectively leverage the knowledge learned from fine-grained visual recognition as a reference to the learning process from web data. This can include transfer learning approaches that transfer knowledge from fine-grained recognition models to generic image classification models trained on web data. To this end, the combination of fine-grained visual recognition and learning from web data holds great potential to advance the field of generic image classification, allowing better performance and broader applicability in real-world scenarios. In the future, we plan to examine a robust pairwise comparison method to enhance the consistency and effectiveness of the model via the teacher-student network module. Also, incorporating more effective weighting strategies into AM loss is another one of our directions.

6 Conclusion

In this work, a novel noise-tolerant learning method is proposed for fine-grained visual recognition. By reducing the influence of the gap between the well-labeled and web data, our method can train with large DNNs. The proposed method comprises many noise-tolerant schemes that take the association embeddings of input data and use its reweighting probability into AM loss to conduct a robust classification. Moreover, the problem of a meta-learning framework is described, which reduces overfitting and makes the parameters of the model noise-tolerant, and the self-ensembling method is applied to build the teacher model to enhance consistent predictions. Finally, the intensive experimental findings on the benchmark datasets verify that our proposed method produces a superior performance to the state-of-the-art approaches.

Acknowledgements This work was supported by National Natural Science Fund for Distinguished Young Scholars of China (Grant No. 62125203), National Natural Science Foundation of China (Grant Nos. 62302233, 62276143), Key Program of the National Natural Science Foundation of China (Grant No. 61932013), and Natural Science Foundation of Jiangsu Province of China (Grant Nos. BK20180470, BK20200739), Postdoctoral Science Foundation of Jiangsu Province of China (Grant No. 2021K172B), and China Postdoctoral Science Foundation (Grant No. 2021M691655).

References

1 Gao Y, Han X T, Wang X, et al. Channel interaction networks for fine-grained image categorization. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 10818–10825

2 Wei X S, Luo J H, Wu J, et al. Selective convolutional descriptor aggregation for fine-grained image retrieval. IEEE Trans Image Process, 2017, 26: 2868–2881

3 Xu T, Zhang P C, Huang Q Y, et al. AttnGAN: fine-grained text to image generation with attentional generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018. 1316–1324

4 Sun Z R, Yao Y Z, Wei X S, et al. Webly supervised fine-grained recognition: benchmark datasets and an approach. In: Proceedings of the IEEE International Conference on Computer Vision, 2021. 1297–1306

5 Xu F R, Wang M, Zhang W, et al. Discrimination-aware mechanism for fine-grained representation learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2021. 813–822

6 Wu L, Wang Y, Li X, et al. Deep attention-based spatially recursive networks for fine-grained visual recognition. IEEE Trans Cybern, 2019, 49: 1791–1802

7 Zhu H W, Ke W J, Li D, et al. Dual cross-attention learning for fine-grained visual categorization and object re-identification. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2022. 4682–4692

8 Wei X S, Song Y Z, Aodha O M, et al. Fine-grained image analysis with deep learning: a survey. IEEE Trans Pattern Anal Mach Intell, 2022, 44: 8927–8948

9 Sun G L, Cholakkal H, Khan S, et al. Fine-grained recognition: accounting for subtle differences between similar classes. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2020. 12047–12054

10 Lin T Y, RoyChowdhury A, Maji S. Bilinear convolutional neural networks for fine-grained visual recognition. IEEE Trans Pattern Anal Mach Intell, 2017, 40: 1309–1322

11 Ji R Y, Wen L Y, Zhang L B, et al. Attention convolutional binary neural tree for fine-grained visual categorization. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2020. 975–984

12 Huang Z X, Li Y. Interpretable and accurate fine-grained recognition via region grouping. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2020. 2642–2651

13 Niu L, Veeraraghavan A, Sabharwal A. Webly supervised learning meets zero-shot learning: a hybrid approach for fine-grained classification. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2018. 7171–7180

14 Zhang C Y, Yao Y Z, Liu H F, et al. Web-supervised network with softly update-drop training for fine-grained visual classification. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2020. 12781–12788

15 Okamoto N, Hirakawa T, Yamashita T, et al. Supplementary: deep ensemble learning by diverse knowledge distillation for fine-grained object classification. In: Proceedings of the European Conference on Computer Vision, 2022. 3872–3881

16 Xu Z, Huang S L, Zhang Y, et al. Webly-supervised fine-grained visual categorization via deep domain adaptation. IEEE Trans Pattern Anal Mach Intell, 2018, 40: 1100–1113

17 Sun X X, Chen L Y, Yang J F. Learning from web data using adversarial discriminative neural networks for fine-grained classification. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2019. 273–280

18 Yang J, Sun X, Lai Y K, et al. Recognition from web data: a progressive filtering approach. IEEE Trans Image Process, 2018, 27: 5303–5315

19 Hendrycks D, Mazeika M, Wilson D, et al. Using trusted data to train deep networks on labels corrupted by severe noise. In: Proceedings of the Advanced Neural Information Processing Systems, 2018. 10456–10465

20 Nguyen T D, Mummadi K C, Ngo P N T, et al. Self: learning to filter noisy labels with self-ensembling. In: Proceedings of the International Conference on Learning Representations, 2020. 817–828

21 Joulin A, van der Maaten L, Jabri A, et al. Learning visual features from large weakly supervised data. In: Proceedings of the European Conference on Computer Vision, 2016. 67–84

22 Krause J, Sapp B, Howard A, et al. The unreasonable effectiveness of noisy data for fine-grained recognition. In: Proceedings of the European Conference on Computer Vision, 2016. 301–320

23 Mirzasoleiman B, Cao K D, Leskovec J. Coresets for robust training of deep neural networks against noisy labels. In: Proceedings of the Advanced Neural Information Processing Systems, 2020. 2067–2076

24 Jiang L, Huang D, Liu M, et al. Beyond synthetic noise: deep learning on controlled noisy labels. In: Proceedings of the International Conference on Machine Learning, 2020. 2414–2423

25 Zhang M Y, Lee J, Agarwal S. Learning from noisy labels with no change to the training process. In: Proceedings of the International Conference on Machine Learning, 2021. 12468–12478

26 Bukchin G, Schwartz E, Saenko K, et al. Fine-grained angular contrastive learning with coarse labels. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2021. 8730–8740

27 Ren M Y, Zeng W Y, Yang B, et al. Learning to reweight examples for robust deep learning. In: Proceedings of the International Conference on Machine Learning, 2019. 4331–4340

28 Haeusser P, Mordvintsev A, Cremers D. Learning by association—a versatile semi-supervised training method for neural networks. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2017. 172–181

29 Dubey A, Gupta O, Raskar R, et al. Maximum-entropy fine grained classification. In: Proceedings of the Advanced Neural Information Processing Systems, 2018. 637–647

30 Tanaka D, Ikami D, Yamasaki T, et al. Joint optimization framework for learning with noisy labels. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2018. 5552–5560

31 Hong G Z, Mao Z Y, Lin X J, et al. Student-teacher learning from clean inputs to noisy inputs. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2021. 12075–12084

32 Wang Z H, Wang S J, Yang S H, et al. Weakly supervised fine-grained image classification via Guassian mixture model oriented discriminative learning. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2020. 1278–1287

33 Wang Z H, Wang S J, Li H J, et al. Graph-propagation based correlation learning for weakly supervised fine-grained image classification. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2020. 12289–12296

34 Hu T, Qi H G. See better before looking closer: weakly supervised data augmentation network for fine-grained visual classification. 2019. ArXiv:1901.09891

35 Ma H, Yang Z Y, Liu H Y. Fine-grained unsupervised temporal action segmentation and distributed representation for skeleton-based human motion analysis. IEEE Trans Cybern, 2022, 52: 13411–13424

36 Fu J L, Zheng H L, Mei T. Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2017. 4438–4446

37 Wang Y, Morariu V I, Davis L S. Learning a discriminative filter bank within a cnn for fine-grained recognition. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2018. 4148–4157

38 Sun M, Yuan Y C, Zhou F, et al. Multi-attention multi-class constraint for fine-grained image recognition. In: Proceedings of the European Conference on Computer Vision, 2018. 805–821

39 Dubey A, Gupta O, Guo P, et al. Pairwise confusion for fine-grained visual classification. In: Proceedings of the European Conference on Computer Vision, 2018. 70–86

40 van Rooyen B, Menon A, Williamson R C. Learning with symmetric label noise: the importance of being unhinged. In: Proceedings of the Advanced Neural Information Processing Systems, 2015. 10–18

41 Hu W, Huang Y Y, Zhang F, et al. Noise-tolerant paradigm for training face recognition cnns. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2019. 11887–11896

42 Ma X J, Huang H X, Wang Y S, et al. Normalized loss functions for deep learning with noisy labels. In: Proceedings of the International Conference on Machine Learning, 2020. 1787–1796

43 Peng Y X, Ye Z D, Qi J W, et al. Unsupervised visual-textual correlation learning with fine-grained semantic alignment. IEEE Trans Cybern, 2022, 52: 3669–3683

44 Shu J, Xie Q, Yi L X, et al. Meta-weight-net: learning an explicit mapping for sample weighting. In: Proceedings of the Advanced Neural Information Processing Systems, 2019. 1917–1928

45 Berthon A, Han B, Niu G, et al. Confidence scores make instance-dependent label-noise learning possible. In: Proceedings of the International Conference on Machine Learning, 2021. 825–836

46 Touvron H, Sablayrolles A, Douze M, et al. Grafit: learning fine-grained image representations with coarse labels. In: Proceedings of the International Conference on Computer Vision, 2021. 874–884

47 Li Y C, Yang J C, Song Y L, et al. Learning from noisy labels with distillation. In: Proceedings of the International Conference on Computer Vision, 2017. 1928–1936

48 Han B, Yao J C, Niu G, et al. Masking: a new perspective of noisy supervision. In: Proceedings of the Advanced Neural Information Processing Systems, 2018. 5841–5851

49 Li J N, Wong Y K, Zhao Q, et al. Learning to learn from noisy labeled data. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2019. 5051–5059

50 Zhu Y H, Liu C L, Jiang S Q. Multi-attention meta learning for few-shot fine-grained image recognition. In: Proceedings of the International Joint Conference on Artificial Intelligence, 2020. 1090–1096

51 Zheng G Q, Awadallah A H, Dumais S. Meta label correction for noisy label learning. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2021. 1290–1297

52 Xu Y J, Zhu L C, Jiang L, et al. Faster meta update strategy for noise-robust deep learning. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2021. 144–153

53 Guan J C, Liu Y, Lu Z W. Fine-grained analysis of stability and generalization for modern meta learning algorithms. In: Proceedings of the Advanced Neural Information Processing Systems, 2022. 4643–4652

54 Finn C, Abbeel P, Levine S. Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the International Conference on Machine Learning, 2017. 1126–1135

55 Tarvainen A, Valpola H. Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Proceedings of the Advanced Neural Information Processing Systems, 2017. 3658–3567

56 Reddi S J, Kale S, Kumar S. On the convergence of Adam and beyond. In: Proceedings of the International Conference on Learning Representations, 2019. 2763–2772

57 Herranz L, Jiang S Q, Li X Y. Scene recognition with CNNs: objects, scales and dataset bias. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2016. 571–579

58 Liu C B, Xie H T, Zha Z J, et al. Filtration and distillation: enhancing region attention for fine-grained visual categorization. In: Proceedings of the Association for the Advancement of Artificial Intelligence, 2020. 11555–11562

59 Wang Z, Hu G S, Hu Q H. Training noise-robust deep neural networks via meta-learning. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2020. 4523–4532

60 Zhang Y, Wei X S, Wu J X, et al. Weakly supervised fine-grained categorization with part-based image representation. IEEE Trans Image Process, 2016, 25: 1713–1725

61 Dixit M, Chen S, Gao D S, et al. Scene classification with semantic fisher vectors. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2015. 2974–2983

62 Zheng H L, Fu J L, Mei T, et al. Learning multi-attention convolutional neural network for fine-grained image recognition. In: Proceedings of the International Conference on Computer Vision, 2017. 5209–5217

63 Guo S, Huang W L, Wang L M, et al. Locally supervised deep hybrid model for scene recognition. IEEE Trans Image Process, 2016, 26: 808–820

64 Zheng H L, Fu J L, Zha Z J, et al. Looking for the devil in the details: learning trilinear attention sampling network for fine-grained image recognition. In: Proceedings of IEEE Conference Computer Vision and Pattern Recognition, 2019. 5012–5021

65 Yang L F, Li X, Song R J, et al. Dynamic MLP for fine-grained image classification by leveraging geographical and temporal information. In: Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, 2022. 10935–10944

66 Li L, Spratling M W. Data augmentation alone can improve adversarial training. In: Proceedings of the International Conference on Learning Representations, 2023. 3465–3475