Large-Scale Few-Shot Learning via Multi-Modal Knowledge Discovery

Shuo Wang[1,3], Jun Yue[3,*], Jianzhuang Liu[3], Qi Tian[4], and Meng Wang[1,2]

[1] School of Computer Science and Information Engineering,
Hefei University of Technology
[2] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center
{shuowang.hfut, eric.mengwang}@gmail.com
[3] Noah's Ark Lab, Huawei Technologies
[4] Huawei Cloud BU
jyue1991@gmail.com, {liu.jianzhuang, tian.qi1}@huawei.com

Abstract. Large-scale few-shot learning aims at identifying hundreds of novel object categories where each category has only a few samples. It is a challenging problem since (1) the identifying process is susceptible to over-fitting with limited samples of an object, and (2) the sample imbalance between a base (known knowledge) category and a novel category is easy to bias the recognition results. To solve these problems, we propose a method based on multi-modal knowledge discovery. First, we use the visual knowledge to help the feature extractors focus on different visual parts. Second, we design a classifier to learn the distribution over all categories. In the second stage, we develop three schemes to minimize the prediction error and balance the training procedure: (1) Hard labels are used to provide precise supervision. (2) Semantic textual knowledge is utilized as weak supervision to find the potential relations between the novel and the base categories. (3) An imbalance control is presented from the data distribution to alleviate the recognition bias towards the base categories. We apply our method on three benchmark datasets, and it achieves state-of-the-art performances in all the experiments.

Keywords: Large-scale few-shot learning, Multi-modal knowledge discovery

1 Introduction

In the past few years, convolutional neural networks (CNNs) have shown a powerful ability on a number of visual tasks, such as classification [13, 8], translation [34, 31, 6], detection [23, 22], reconstruction [17], and segmentation [37, 37]. Although CNNs have strong robustness to object and background variations, they can hardly show a good performance without large amounts of training data. Meanwhile, it is time-consuming and expensive to collect and label these data. On the contrary, a human can recognize and remember a new object from

* Corresponding Author

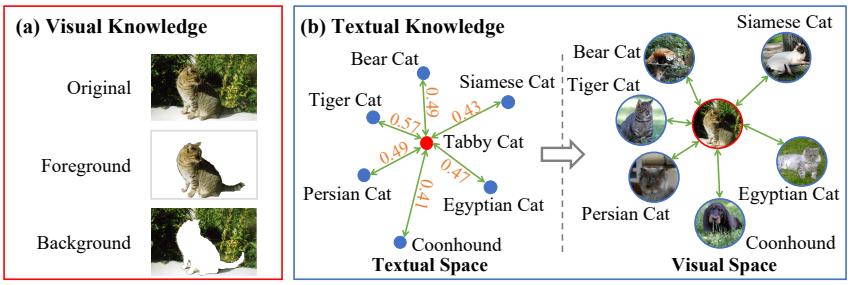

Fig. 1. (a) Given an image, we define its three visual parts, where the foreground focuses on the object, the background describes the environment related to the object, and the original image includes both the foreground and the background. (b) Given a novel label "Tabby Cat", we show the similarities between it and other labels from the base in the textual space. The similarity scores (in orange) are calculated by the word2vec method. This textual knowledge can be effectively used to help the recognition of novel objects as soft supervision information. For example, these scores are the largest among those between "Tabby Cat" and all the labels in the base. Based on them, these objects listed in (b) are considered as most similar to tabby cat, and their similarities with tabby cat can be exploited to help the recognition of a tabby cat image.

only a few samples (or even one) of it. Therefore, few-shot learning (FSL) is proposed to imitate this human ability. The FSL task can be divided into two categories, traditional FSL [30, 25, 21, 20] and large-scale FSL (LS-FSL) [32, 7, 15, 5]. Different from the traditional FSL which recognizes small N ($N \leqslant 20$) classes of novel objects, LS-FSL is a more realistic task that aims to identify hundreds of novel categories without forgetting those categories (called base categories) that have been recognized.

For the task of FSL or LS-FSL, we believe that the key problems are (1) how to extract more information from the available images, and (2) how to effectively use the base objects to help the recognition of novel objects. For the first problem, a popular strategy is using a CNN trained on the base categories to extract the global features of novel objects directly [7, 32]. It aims to yield a transferable feature representation (textures and structures) to describe a novel category. However, it is insufficient to represent the novel samples since their global features cannot well describe the distribution of their category with the limited samples. Therefore, to discover more information from the images, we define three visual parts (shown in Fig. 1(a)) computed by an unsupervised saliency detection method. They are used as the network input for training and inference. The effectiveness of this scheme will be elaborated in Section 3.1.

For the second problem, previous LS-FSL works [7, 32] train a classifier under the supervision of given labels (called hard labels in this paper) to learn the distribution over both the base and the novel categories. In our method, in addition to the hard labels, we introduce semantic soft labels generated from textual knowledge to help the network learn a more powerful classifier. More

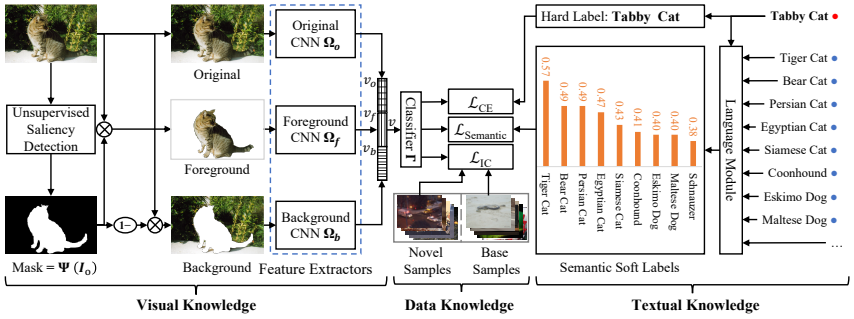

Fig. 2. The overview of the proposed framework, where $\mathcal{L}_{\text{CE}}$, $\mathcal{L}_{\text{Semantic}}$, and $\mathcal{L}_{\text{IC}}$ are three losses.

details can be found in Section 3.2. Here is an example shown in Fig. 1(b). The novel object can be guessed similar to a cat since the similarity between its label "Tabby Cat" and the label "Tiger Cat" in the base is relatively large (0.57). Besides, the score (0.41) gives the information that the input would be more similar to "Coonhound" than to other categories not shown in Fig. 1(b), such as "Car".

The overview of our framework is depicted in Fig. 2. First, we use the three visual parts of a given image as the input to three independently trained CNNs to extract the features from this image. Second, we calculate the similarities between the hard label of this image and other known labels in the base from the textual knowledge, and use these similarities to generate the semantic soft labels. Third, we design a classifier and train it with both the hard label and the soft labels. The main contributions of our method are fourfold.

(1) We introduce the strategy of extracting more visual information from images, and analyze its advantage for FSL and LS-FSL.

(2) We extract textual knowledge to help the classifier learn from language, which can also be used to improve existing LS-FSL methods.

(3) Two novel losses are designed for semantic knowledge discovery and sample imbalance control during training.

(4) Our method is simple yet powerful; it achieves state-of-the-art performances on popular LS-FSL and FSL datasets.

2 Related Work

2.1 Traditional Few-Shot Learning (FSL)

The methods [30, 25, 21, 20] based on meta-learning are proposed to solve the problem of FSL. They train a meta-learner from many FSL tasks (with base categories) without relying on ad hoc knowledge to suit for new FSL tasks (with novel categories). Metric-learning is another popular approach; it attempts to train a network which can make samples of the same class closer and samples

of different classes farther in the feature space [28, 30]. Sample hallucination is also useful to generate more training data [38, 36]. All the above methods consider small datasets like Omniglot [14], CIFAR [12], and Mini-ImageNet [30], and focus on the N-way-K-shot recognition problem that identifies N ($N \leqslant 20$) novel classes and each class has K ($K = 1$ or 5 usually) samples.

2.2 Large-Scale Few-Shot Learning (LS-FSL)

Recent works [5, 32, 7, 4, 15] start to pay attention to the more practical LS-FSL problem that learns hundreds of novel categories without forgetting the base categories. Specifically, [7] hallucinates new samples in the feature space by using a separate Multilayer Perceptron (MLP) to model the relationships between the foregrounds and the backgrounds of images. Wang *et. al.* train a meta-learner with hallucination to expand the training set and to classify the samples simultaneously [32]. The work in [15] clusters hierarchical textual labels both from the base and the novel categories to train a feature extractor, and uses the learned features to search the novel labels by the nearest neighbor (NN) method. Gidaris *et. al.* design a sample classification weight generator with attention mechanism and modify the classifier with the cosine similarity [4]. Peng *et. al.* imprint the weights of the FSL classifier from both visual and textual features [19]. The work in [33] hallucinates novel samples by a generative adversarial network (GAN) which transfers the textual features to the novel image features. The work in [5] combines meta-learning with a graph neural network (GNN) to model the relationships of different categories and predicts the parameters of novel classes.

Different from the sample hallucination methods [38, 36, 7, 32, 33], we do not generate hallucinated training samples; instead, we aim to extract more information from each image for training and inference. Compared with the most related method [15] that uses the knowledge to train the feature extractor (with NN as the classifier), we exploit the textual knowledge to train the classifier (with a pre-trained feature extractor). Besides, we discover the knowledge from data distribution and use it to balance the recognition processing.

3 The Proposed Approach

3.1 Visual Knowledge Discovery

For visual representation, [27] visualizes the responses on images from trained CNNs via gradient-based localization. The results show that CNNs trained with large-scale samples tend to use the object regions for the representation. In LS-FSL, the base categories usually have large-scale training samples (*e.g.*, about 1300 samples in one category). Therefore, a CNN trained on the base data is more inclined to focus on the textures and structures of the objects it learns. As shown in the column of "Original Response"[1] in Fig. 3(a), given a base sample in the "Partridge" category, the responsive regions on the original image focus

[1] The response on the original image is called original response.

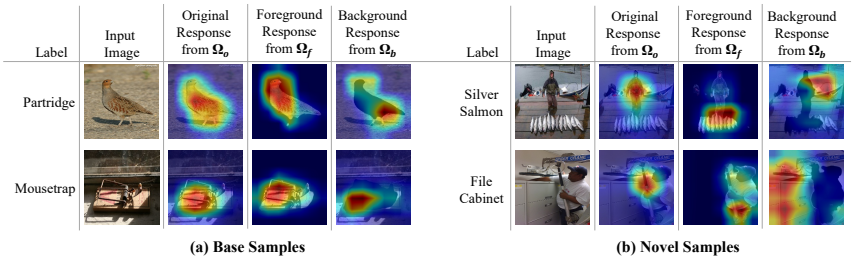

Fig. 3. The responsive regions of three CNNs (ResNets-50) visualized by Grad-CAM [27] from several samples in ImageNet-FS [7].

on the body of the bird since the CNN is trained with many partridge images. Similar result can be found for the mousetrap image. However, this CNN may deviate the responses from novel objects and overlook them. For example, it concentrates on the fisherman but not the fishes in the image with label "Silver Salmon" (Fig. 3(b)). Thus, it is important to make the responses more accurate or to enlarge the response regions of novel samples. However, it is difficult to make the responses focus on novel objects since there are no (or only a few) novel samples in the training procedure of a CNN.

Inspired by the work in [9] that humans have a remarkable ability to interpret a complex scene by selecting a subset (foreground) of the available sensory information primarily and then enlarging the vision to the other part (background) of the scene, we extract more visual knowledge from available samples to imitate this human behavior to enrich the representation. First, we employ an off-the-shelf unsupervised saliency detection network [37] to segment the salient region (foreground) from the background of an image. Let the unsupervised saliency detection network be $\boldsymbol{\Psi}$ and the original image be $\boldsymbol{I}_o$. Then the mask of the saliency regions is denoted as $\boldsymbol{\Psi}(\boldsymbol{I}_o)$ (see Fig. 2 for example, where $\boldsymbol{\Psi}(\boldsymbol{I}_o)$ is of the same size as $\boldsymbol{I}_o$ with 1 for the foreground and 0 for the background). Thus, the foreground $\boldsymbol{I}_f$ and the background $\boldsymbol{I}_b$ are calculated by

$$\boldsymbol{I}_f = \boldsymbol{\Psi}(\boldsymbol{I}_o) \otimes \boldsymbol{I}_o, \quad \boldsymbol{I}_b = (1 - \boldsymbol{\Psi}(\boldsymbol{I}_o)) \otimes \boldsymbol{I}_o, \tag{1}$$

where $\otimes$ denotes the Hadamard product.

Second, we train three independent CNNs, $\boldsymbol{\Omega}_o$, $\boldsymbol{\Omega}_f$, and $\boldsymbol{\Omega}_b$, to learn the representation of the three visual parts $\boldsymbol{I}_o$, $\boldsymbol{I}_f$, and $\boldsymbol{I}_b$, respectively, from all the base samples under the supervision of their hard labels. The reason to use three independent CNNs is because those parts have different distributions. To analyze the effectiveness of the visual knowledge discovery, we visualize the responsive regions of the foregrounds and backgrounds from the trained CNNs using the visualization method Grad-CAM [27] in Fig. 3. For the base samples, it is easy to see that $\boldsymbol{\Omega}_o$ and $\boldsymbol{\Omega}_f$ focus on the regions of the bird and the mousetrap. Although there is no object in the background $\boldsymbol{I}_b$, the responses of $\boldsymbol{\Omega}_b$ still concentrate on the edges of the bird and the mousetrap. In contrast, these CNNs perform

differently on the novel samples. For the two novel samples in Fig. 3(b), if only $\boldsymbol{\Omega}_o$ is used, then the responses are mainly on the fisherman and the man for the "Silver Salmon" and "File Cabinet" images, respectively. Obviously, the extracted feature representations are very likely to cause failure recognition. When $\boldsymbol{\Omega}_f$ is used, we can see that the response of "Silver Salmon" is shifted to the fishes, which is what we need. Why $\boldsymbol{\Omega}_b$ is required is because in many cases, such as the "File Cabinet" image, the result of the unsupervised saliency detection does not give the main object corresponding to the label; instead, the object is considered as the background. Therefore, $\boldsymbol{\Omega}_b$ is necessary to extract useful features in these cases, as shown in the lower-right sub-image of Fig. 3(b).

In our framework, the features extracted by $\boldsymbol{\Omega}_o$, $\boldsymbol{\Omega}_f$, and $\boldsymbol{\Omega}_b$ are denoted as

$$v_o = \boldsymbol{\Omega}_o(\boldsymbol{I}_o), \quad v_f = \boldsymbol{\Omega}_f(\boldsymbol{I}_f), \quad v_b = \boldsymbol{\Omega}_b(\boldsymbol{I}_b), \tag{2}$$

which are then concatenated together as $v = [v_o, v_f, v_b]$ to describe a sample and for the training of the classifier.

3.2 Textual Knowledge Discovery

Humans can recognize a new category with a few samples of it because they have seen many other related objects or learnt them from textual knowledge, and thus are already familiar with their salient features. Inspired by this, to help the recognition of a novel category, we find its similar categories from the base by using textual knowledge. For example, in Fig. 1(b), the similarity scores between the novel label "Tabby Cat" and the labels from the base in the textual space can describe their similarities to a large extent. Compared with the hard labels of the novel categories, these scores provide more diverse and informative supervision for recognizing the novel samples. To effectively use this textual knowledge to help our network learn a better classifier, we extract the semantic knowledge to enrich the supervision information.

The classifier in our method has two purposes: (1) learning the novel categories without forgetting the base categories, and (2) using the base knowledge to help learn the novel categories. To achieve these, we design a C-way classifier $\boldsymbol{\Gamma}$ to learn the prediction distribution from both the base and the novel categories, where C is the total number of the base and the novel categories.

Given a feature v extracted by the trained CNNs, the prediction by the classifier is denoted as $p = \boldsymbol{\Gamma}(v)$, where p is a C-dimensional vector. We design our semantic soft label supervision based on the textual knowledge. Given the labels of a novel sample k and the base samples, we first express these labels as vectors by the available word2vec method [15]. Second, we compute the similarities between the novel label and the base labels using the cosine similarity with their vector representations. Then, we obtain a C_{base}-dimensional vector ℓ^k, where C_{base} is the number of the base categories, and the components of ℓ^k are the similarity scores. $\ell^k \in \mathbb{R}^{C_{\text{base}}}$ provides non-sparse supervision measurements to describe the similarities between the novel and the base objects. We call ℓ^k the semantic soft label for the novel sample k. Next, we design a semantic soft loss based on ℓ^k.

In this work, the classifier $\boldsymbol{\Gamma}$ is a simple network, *e.g.*, as simple as one fully-connected layer. Its prediction (for the novel sample k) $p^k \in \mathbb{R}^C$ is normalized by the sigmoid function, generating a normalized prediction vector $s^k \in \mathbb{R}^C$ with $s_i^k = \text{sigmoid}(p_i^k)$, where $s_i^k \in (0,1)$ and p_i^k, $i = 1, \ldots, C$, are the components of s^k and p^k, respectively. We then define the semantic soft loss for the novel sample k as

$$\mathcal{L}_k = -\frac{1}{|C_{\text{base}}|} \sum_{j \in \text{base}} \boldsymbol{\gamma} \log s_j^k, \ \boldsymbol{\gamma} = \begin{cases} 1, \text{if } \ell_j^k \geqslant \alpha, \\ 0, \text{if } \ell_j^k < \alpha, \end{cases} \tag{3}$$

where ℓ_j^k is the similarity score between the label of the novel sample k and the j^{th} base category label, and α is a threshold controlling the usage of the textual knowledge. Minimizing $\mathcal{L}_k$ can be loosely considered as maximizing the normalized log likelihood of s_j^k's under $\ell_j^k \geqslant \alpha$, implying that these normalized predictions s_j^k's should be large because they are more similar to the novel label. During training, if there are N novel samples in a training batch, the semantic soft loss for this batch is

$$\mathcal{L}_{\text{Semantic}} = \sum_{k=1}^{N} \mathcal{L}_k. \tag{4}$$

3.3 Imbalance Control from Data Distribution

In LS-FSL, [7] shows that a classifier trained under the supervision of hard labels without other assistant strategies bias the recognition towards the base categories. Specifically, the mean accuracy of the novel categories is much worse than that of the base categories. This is because each base category can use many samples to well describe its feature distribution, while a novel category has only a few training samples. To alleviate the effect of the imbalanced samples between the novel and the base categories, we first oversample the samples from the novel categories in each training batch. Second, we regard the distribution of the dataset as the prior knowledge and then design an imbalance control strategy to bias the predictions towards novel samples.

Given a training batch with B base samples and N novel samples, the predictions of these samples from the classifier are denoted as $\{p^b\}_{b=1}^B$ and $\{p^n\}_{n=1}^N$, where $p^b \in \mathbb{R}^C$ and $p^n \in \mathbb{R}^C$ are the predictions of the b^{th} base sample and the n^{th} novel sample, respectively. Then, the imbalance control loss $\mathcal{L}_{\text{IC}}$ is defined as

$$\mathcal{L}_{\text{IC}} = \sum_{n=1}^{N} \sum_{b=1}^{B} \max\left(\frac{\langle p^b, p^n \rangle}{\|p^b\| \cdot \|p^n\|} + \beta, 0\right), \tag{5}$$

where $\beta \in [0,1]$ is a hyper-parameter to determine the strength of the imbalance control, and $\langle \bullet, \bullet \rangle$ is the inner product between two vectors.

Without the imbalance control, due to much more training data from the base categories, both p^b and p^n have relatively large predictions for the base

Table 1. Data partitions of ImageNet-FS [7] for different experiments, where 389 base categories are used to train the feature extractors, ALL-S_1 and ALL-S_2 are used for the ablation studies and the comparisons with other methods, respectively.

	ALL-S_1 (493)	ALL-S_2 (507)
Base Categories (389)	BASE-S_1: 193	BASE-S_2: 196
Novel Categories (611)	NOVEL-S_1: 300	NOVEL-S_2: 311

categories, meaning that p^b and p^n have a relatively large correlation. Using the proposed imbalance control by minimizing $\mathcal{L}_{\text{IC}}$, we can reduce these correlations, thus making the prediction towards the novel categories when the input is a novel sample. Note that imposing this loss has little effect on base samples because there are much more training data in the base categories.

3.4 Hard Label Supervision and Total Loss

The hard labels are also used to train the classifier with the cross-entropy loss. Given a training batch with $B+N$ samples, the cross-entropy loss between the predictions $\{p^h\}_{h=1}^{B+N}$ and their hard labels $\{L^h\}_{h=1}^{B+N}$ is calculated by

$$\mathcal{L}_{\text{CE}} = \sum_{h=1}^{B+N} \text{CrossEntropy}(\text{softmax}(p^h), L^h). \tag{6}$$

Finally, the total loss for a training batch is defined as

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \mu_1 \mathcal{L}_{\text{Semantic}} + \mu_2 \mathcal{L}_{\text{IC}}, \tag{7}$$

where μ_1 and μ_2 are two weighting factors.

4 Experiments

In this section, we evaluate our method on three tasks, LS-FSL, traditional FSL, and improving other LS-FSL methods. We use the pre-trained unsupervised saliency detection network [37] to split an image into three visual parts, and use the pre-trained word2vec [15] to represent the labels with vectors. In our experiments, the three feature extractors are of the same structure. It should be mentioned that this saliency detection network [37] is trained unsupervisedly on the MSRA-B dataset [16] without using any object masks and category labels; in other words, no extra visual supervision information is introduced to help the FSL and LS-FSL tasks.

4.1 Large-Scale Few-Shot Learning

4.1.1 Experiments on the ImageNet-FS Benchmark

Dataset and evaluation. ImageNet-FS [7] contains 1000 categories from the ImageNet dataset [24]. It is divided into 389 base categories and 611 novel categories, where 193 base categories and 300 novel categories are used for validating

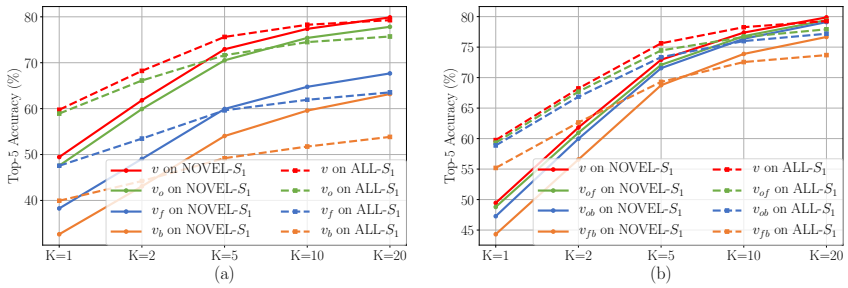

Fig. 4. Top-5 accuracies (%) using different visual parts (v_o, v_f, or v_b) and their concatenations (v_{of}, v_{ob}, v_{fb}, or v) on NOVEL-S_1 and ALL-S_1. The feature extractors are ResNets-10.

the hyper-parameters, and the remaining 196 base categories and 311 novel categories are used for classifier learning and testing. Denote the former 493 categories as ALL-S_1 and the latter 507 categories as ALL-S_2. There are about 1300 samples in a base category. For novel categories, there are 5 settings with $K = 1$, 2, 5, 10, and 20 training samples per category. The evaluation of this benchmark contains two parts: (1) NOVEL-S_2: the Top-5 recognition accuracy of the 311 testing novel categories, and (2) ALL-S_2: the Top-5 recognition accuracy of the 507 (both the base and the novel) categories. More details of the settings can be found from [7]. In Table 1, we summarize the data partitions for different experiments in this section.

Training the feature extractors. To compare with other methods, we use ResNets-10 [8] or ResNets-50 [8] as the feature extractors $\boldsymbol{\Omega}_o$, $\boldsymbol{\Omega}_f$, and $\boldsymbol{\Omega}_b$. We respectively train $\boldsymbol{\Omega}_o$, $\boldsymbol{\Omega}_f$, and $\boldsymbol{\Omega}_b$ using the original images, the foreground images, and the background images with their hard labels from all the 389 base categories ([7] also uses all the 389 base categories to train its feature extractor). During this training, we optimize the parameters of these feature extractors with the squared gradient magnitude (SGM) loss [7] using SGD [1] for 200 epochs with a batch size = 256. The learning rate starts at 1 and is divided by 10 for every 50 epochs. The weight decay is fixed at 0.0001. Note that the three classifiers for training the three feature extractors are discarded after this training, which are different from the classifier discussed next.

Training the classifier. Our classifier $\boldsymbol{\Gamma}$ has only one fully-connected layer with normalized weights [26]. It classifies the features from both the base and the novel categories. It is trained with our loss $\mathcal{L}$ in Eq. (7) for 90 epochs. The batch size $(B + N)$ is set to 1000 with $B = 500$ and $N = 500$. We use the Adam optimization [10] with the starting learning rate of 0.001 and the weight decay of 0.0001. The learning rate is divided by 10 after every 30 epochs.

Ablation study — the effectiveness of the visual knowledge discovery. In this ablation study, the classifier is trained with only the $\mathcal{L}_{\text{CE}}$ loss. To evaluate the effectiveness of the visual knowledge discovery, we train seven classifiers with the features of v_o, v_f, v_b, v_{of}, v_{ob}, v_{fb}, or v from ALL-S_1, where v_{of}, v_{ob},

Table 2. Top-5 accuracies (%) on NOVEL-S_1 and ALL-S_1 with different α in the textual knowledge discovery. The feature extractors are ResNets-10. "$\mathcal{L}_{\text{CE}}$ Only" denotes the network without using the textual knowledge.

	NOVEL-S_1/ALL-S_1				
	$K = 1$	$K = 2$	$K = 5$	$K = 10$	$K = 20$
$\alpha = 0.0$	50.2/59.7	61.8/68.3	72.7/75.7	78.4/78.7	80.0/80.0
$\alpha = 0.1$	50.6/59.8	62.1/68.4	72.9/75.9	78.6/78.8	79.9/80.2
$\alpha = 0.2$	51.0/59.7	62.2/68.2	73.0/75.9	78.6/78.9	80.1/80.2
$\alpha = 0.3$	50.9/59.8	**62.3**/68.2	73.0/75.9	78.5/78.8	79.9/80.1
$\alpha = 0.4$	51.0/59.9	62.1/68.2	73.0/75.9	78.5/78.8	79.8/80.0
$\alpha = 0.5$	**51.1**/**60.1**	62.2/**68.7**	**73.8**/76.0	**78.8**/**79.0**	**80.9**/80.3
$\alpha = 0.6$	50.6/59.9	62.0/68.3	73.0/75.9	78.5/78.8	80.1/80.2
$\alpha = 0.7$	50.5/59.7	61.8/68.2	72.7/75.7	78.5/78.8	80.3/80.3
$\alpha = 0.8$	50.1/59.5	61.7/68.1	72.9/75.9	78.4/78.7	80.2/**80.4**
$\alpha = 0.9$	50.2/59.6	61.8/68.2	73.2/**76.2**	78.4/78.8	80.1/80.2
$\mathcal{L}_{\text{CE}}$ Only	49.5/59.7	61.8/68.2	72.9/75.6	77.3/78.3	79.8/79.3

v_{fb}, and v represent the concatenations $[v_o, v_f]$, $[v_o, v_b]$, $[v_f, v_b]$, and $[v_o, v_f, v_b]$, respectively. These features are extracted by the ResNet-10 feature extractors. The recognition performances on the $K = 1$, 2, 5, 10, and 20 settings and on the evaluations of NOVEL-S_1 and ALL-S_1 are shown in Fig. 4. Fig. 4(a) indicates that both the foregrounds and the backgrounds can help the network classify the novel and the base samples. As expected, the foregrounds are more useful than the backgrounds, and the original images give more information than either foregrounds or backgrounds. Comparing Fig. 4(a) with Fig. 4(b), we can see that the performances of different concatenations are better than their individual features. More importantly, the combination v of v_o, v_f, and v_b provides the best performance, which validates the effectiveness of our visual knowledge discovery.

Ablation study — the textual knowledge discovery. In this ablation study, the classifier is trained with the combined features v and with $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{Semantic}}$ losses ($\mu_1 = 1$, $\mu_2 = 0$). Since the threshold α controls the strength of the textual knowledge usage, we conduct an experiment with different α on NOVEL-S_1 and ALL-S_1. The results are shown in Table 2. For comparison, we also give the results without the textual knowledge in the last row of Table 2. First, we can see that the textual knowledge discovery is effective and stable; the network with it consistently outperforms the network without it for different α. Second, the best value of $\alpha = 0.5$ is selected when the feature extractors are ResNets-10. We also have another similar experiment when the feature extractors are ResNets-50 (omitted here), where the best value of α is 0.3.

In Fig. 5, we show several examples of the results by the network with the textual knowledge discovery (denoted as "$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Semantic}}$") and the network without it (denoted as "$\mathcal{L}_{\text{CE}}$ Only"). It is easy to see that "$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Semantic}}$" obtains the top-ranked results that are more relevant to the input objects. For example, when the input novel image is a kind of dog ("Cardigan Welsh Corgi" here), all the top 7 results by "$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Semantic}}$" are dog labels, but the second and the third results ("Plastic Bag" and "Paper Towel") by "$\mathcal{L}_{\text{CE}}$ Only" are

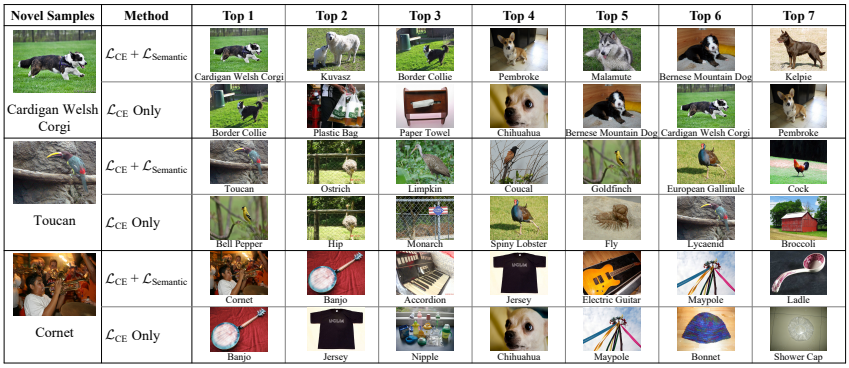

Fig. 5. The recognition results of several novel samples by the networks with and without the textual knowledge discovery, denoted as "$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{Semantic}}$" and "$\mathcal{L}_{\text{CE}}$ Only", respectively. In this experiment, $K = 1$. We randomly select one image from each label category for easy understanding of the objects corresponding to the labels.

Table 3. Top-5 accuracies (%) in the evaluation of the imbalance control on NOVEL-S_1 and ALL-S_1. The feature extractors are ResNets-10.

	Method	$K = 1$	$K = 2$	$K = 5$	$K = 10$	$K = 20$
NOVEL-S_1	$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{IC}}$	**50.1**	**62.0**	**73.4**	**78.1**	**80.7**
	$\mathcal{L}_{\text{CE}}$ Only	49.5	61.8	72.9	77.3	79.8
ALL-S_1	$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{IC}}$	**60.1**	**68.5**	**75.9**	**78.6**	**79.9**
	$\mathcal{L}_{\text{CE}}$ Only	59.7	68.2	75.6	78.3	79.3

not relevant. This figure clearly shows the effectiveness of the textual knowledge discovery.

Ablation study — imbalance control from data distribution. In this ablation study, the classifier is trained with the combined features v and with the $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{IC}}$ losses ($\mu_1 = 0$, $\mu_2 = 1$). We try different values of $\beta \in [0, 1]$, and find that the network performance is insensitive to them. Thus, we set $\beta = 1$. In Table 3, we compare the network using the imbalance control ("$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{IC}}$") and the network without it ("$\mathcal{L}_{\text{CE}}$ Only"). In all the cases, "$\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{IC}}$" outperforms "$\mathcal{L}_{\text{CE}}$ Only", indicating the usefulness of the imbalance control.

Comparisons with other LS-FSL methods. In this experiment, we compare our method with state-of-the-art LS-FSL ones. The hyper-parameters of our model are set to $\alpha = 0.5$ when ResNets-10 are used as the feature extractors, $\alpha = 0.3$ when ResNets-50 are used, $\beta = 1$, $\mu_1 = 1$, and $\mu_2 = 1$. The compared methods include Prototypical Nets (PN) [28], Matching Networks (MN) [30], Logistic Regression [7], Prototype Matching Nets (PMN) [7], SGM [7], LwoF [4], wDAE-GNN [5], Nearest Neighbor (NN) [15], LSD [2], and KTCH [15].

All the results on NOVEL-S_2 and ALL-S_2 are listed in Table 4. Our method outperforms others in all the cases. Compared with these methods, our improvements for novel categories (NOVEL-S_2) are larger than those for both the base and the novel categories (ALL-S_2). Besides, our improvements when using

Table 4. Top-5 accuracies (%) by different methods on NOVEL-S_2 and ALL-S_2 [7]. Here, "from [32]" means the accuracy numbers of the corresponding method are from [32]. "*H.*" means data hallucination.

Method with ResNet-10	NOVEL-S_2					ALL-S_2				
	$K=1$	$K=2$	$K=5$	$K=10$	$K=20$	$K=1$	$K=2$	$K=5$	$K=10$	$K=20$
Prototypical Nets (from [32])	39.3	54.4	66.3	71.2	73.9	49.5	61.0	69.7	72.9	74.6
Matching Networks (from [32])	43.6	54.0	66.0	72.5	76.9	54.4	61.0	69.0	73.7	76.5
Logistic Regression + *H.* [7]	40.7	50.8	62.0	69.3	76.5	52.2	59.4	67.6	72.8	76.9
SGM + *H.* [7]	44.3	56.0	69.7	75.3	78.6	54.8	62.6	71.6	76.0	78.2
PMN + *H.* [32]	45.8	57.8	69.0	74.3	77.4	57.6	64.7	71.9	75.2	77.5
LwoF [4]	46.2	57.5	69.2	74.8	78.1	58.2	65.2	72.7	76.5	78.7
wDAE-GNN [5]	48.0	59.7	70.3	75.0	77.8	59.1	66.3	73.2	76.1	77.5
Ours	**51.8**	**63.1**	**73.6**	**78.1**	**80.9**	**60.1**	**68.5**	**75.9**	**78.9**	**80.5**
Method with ResNet-50	NOVEL-S_2					ALL-S_2				
	$K=1$	$K=2$	$K=5$	$K=10$	$K=20$	$K=1$	$K=2$	$K=5$	$K=10$	$K=20$
Nearest Neighbor (from [15])	49.5	59.9	70.1	75.1	77.6	-	-	-	-	-
Prototypical Nets (from [32])	49.6	64.0	74.4	78.1	80.0	61.4	71.4	78.0	80.0	81.1
Matching Networks (from [32])	53.5	63.5	72.7	77.4	81.2	64.9	71.0	77.0	80.2	82.7
SGM + *H.* [7]	52.8	64.4	77.3	82.0	84.9	63.7	71.6	80.2	83.3	85.2
PMN + *H.* [32]	54.7	66.8	77.4	81.4	83.8	65.7	73.5	80.2	82.8	84.5
LSD [2]	57.7	66.9	73.8	77.6	80.0	-	-	-	-	-
KTCH [15]	58.1	67.3	77.6	81.8	84.2	-	-	-	-	-
Ours	**58.5**	**69.7**	**79.2**	**83.0**	**85.5**	**66.7**	**74.8**	**81.5**	**83.4**	**85.5**

ResNets-10 as the feature extractors are more significant than those when using ResNets-50. With ResNets-10, compared with the best previous method wDAE-GNN, ours outperforms it by at least 3.1% in accuracy for $K = 1$, 2, 5, 10, and 20 on NOVEL-S_2, which is significant for LS-FSL.

4.1.2 Experiments on the ImNet Benchmark

Dataset and evaluation. ImNet [11] is another LS-FSL dataset, which is also selected from ImageNet. It contains 1000 base categories and 360 novel categories. For novel categories, there are 5 settings with $K = 1$, 2, 3, 4, and 5 training samples per category. The evaluation of this benchmark is to recognize the samples from these 360 novel categories. More details are described in [11].

Training the feature extractors and the classifier. The feature extractors are three ResNets-50. The training of them is similar to that in the experiments on ImageNet-FS. Besides, the training of the classifier $\boldsymbol{\Gamma}$ is also similar to that in the experiments on ImageNet-FS.

Comparisons with other LS-FSL methods. The compared methods include Nearest Neighbor (NN) [15], SGM [7], PPA [21], LSD [2], and KTCH [15]. The Top-5 accuracies by our and these methods on the novel categories are listed in Table 5. We can see that our method again performs best on this dataset in all the cases. Compared with the previous best model KTCH, our improvements for $K = 1$, 2, 3, 4, and 5 are 4.4%, 5.7%, 5.9%, 5.9%, and 6.7%, respectively, showing the power of our approach.

Table 5. Top-5 accuracies (%) by different methods on the novel categories from ImNet [11]. All the methods use ResNet-50 for feature extraction. All the accuracy numbers of other methods are from [15].

Method	Novel Categories				
	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
Nearest Neighbor (from [15])	34.2	43.6	48.7	52.3	54.0
SGM (from [15])	31.6	42.5	49.0	53.5	56.8
PPA (from [15])	33.0	43.1	48.5	52.5	55.4
LSD (from [15])	33.2	44.7	50.2	53.4	57.6
KTCH [15]	39.0	48.9	54.9	58.7	60.5
Ours	**43.4**	**54.6**	**60.8**	**64.6**	**67.2**

Table 6. Top-1 accuracies (%) by different methods on the testing novel categories of Mini-ImageNet with 95 confidence intervals. †: Using the validation set (in addition to the base set) for feature extractor training.

Method	Feature Extractor	$K = 1$	$K = 5$
MAML [3]	Conv-4-64	48.70 ± 1.84%	63.10 ± 0.92%
PN [28]	Conv-4-64	49.42 ± 0.78%	68.20 ± 0.66%
RelationNet [29]	Conv-4-64	50.40 ± 0.80%	65.30 ± 0.70%
MetaGAN [38]	Conv-4-64	52.71 ± 0.64%	68.63 ± 0.67%
SalNet [36]	Conv-4-64	57.45 ± 0.88%	72.01 ± 0.67%
MetaNet [18]	ResNets-12	57.10 ± 0.70%	70.04 ± 0.63%
PPA† [21]	WRN-28-10	59.60 ± 0.41%	73.74 ± 0.19%
LEO† [25]	WRN-28-10	61.76 ± 0.08%	77.59 ± 0.12%
LwoF (from [5])	WRN-28-10	60.06 ± 0.14%	76.39 ± 0.11%
wDAE-GNN† [5]	WRN-28-10	62.96 ± 0.15%	78.85 ± 0.10%
Ours	WRN-28-10	**64.40** ± 0.43%	**83.05** ± 0.28%

4.2 Traditional Few-Shot Learning

Dataset and evaluation. For traditional FSL, we apply our model on the Mini-ImageNet dataset [30]. Mini-ImageNet consists of 100 categories from ImageNet and each category has 600 images. It is divided into three parts: 64 base categories, 16 novel categories for validation, and the remaining 20 novel categories for testing. This dataset is evaluated on several 5-way-K-shot classification tasks. In each task, 5 novel categories are sampled first, then K samples in each of the 5 categories are sampled for training, and finally 15 samples (different from the previous K samples) in each of the 5 categories are sampled for testing. To report the results, we sample 2000 such tasks and compute the mean accuracies over all the tasks.

Training the feature extractors. We use the 2-layer wide residual networks (WRN-28-10) [35] as the feature extractors. They are trained with the cross-entropy loss using the Adam optimization for 200 epochs with a batch size = 256 on all the 64 base categories. The learning rate starts at 0.001 and is divided by 10 for every 50 epochs. The weight decay is fixed at 0.0001.

Training the classifier. The training of the classifier $\boldsymbol{\Gamma}$ on this dataset is similar to that in the experiments on previous datasets. It is trained with our loss $\mathcal{L}$ in Eq. (7) for 40 epochs. The batch size $(B+N)$ is set to 100 with $B = 50$ and $N = 50$. We use the Adam optimization with the starting learning rate of 0.001 and the weight decay of 0.0001. The learning rate is divided by 10 after

Table 7. Top-5 accuracies (%) on ImNet.

Method	Novel Categories				
	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
SGM [7]	31.4	42.7	49.1	53.2	56.4
SGM $+\mathcal{T}$	**33.5**	**44.1**	**50.1**	**54.5**	**57.3**
KTCH [15]	36.0	47.0	52.9	57.2	60.4
KTCH $+\mathcal{T}$	**40.1**	**50.5**	**56.6**	**60.8**	**63.3**

every 10 epochs. We conduct an experiment with different α on the validation set (omitted here) and find the best value of α is 0.2.

Comparisons with other traditional FSL methods. As shown in Table 6, we compare our method with MAML [3], PN [28], RelationNet [29], MetaGAN [38], SalNet [36], MetaNet [18], PPA [21], LEO [25], LwoF [4], and wDAE-GNN [5]. Again, our method outperforms them. Specifically, compared with the best previous method wDAE-GNN, we obtain 1.44% and 4.2% accuracy improvements for $K = 1$ and $K = 5$, respectively. Note that wDAE-GNN uses both the base and the validation categories for feature extractor training, while ours are trained with only the base categories.

4.3 Textual Knowledge Discovery on Other Methods

Our textual knowledge discovery can be used to improve other LS-FSL methods. In this section, we show two examples based on SGM [7] and KTCH [15]. We train our classifier (with the $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{Semantic}}$ losses) using the features[2] extracted by SGM or KTCH on the ImNet dataset. The two new models are denoted as SGM $+\mathcal{T}$ and KTCH $+\mathcal{T}$, respectively, where "$\mathcal{T}$" means the textual knowledge discovery. The recognition results by SGM, KTCH, and the two new models are shown in Table 7. We can see that the new models SGM $+\mathcal{T}$ and KTCH $+\mathcal{T}$ respectively improve SGM and KTCH by significant margins.

5 Conclusion

In this paper, we have proposed three schemes to tackle the problem of large-scale few-shot learning (LS-FSL): (1) visual knowledge discovery for better object representation, (2) textual knowledge discovery for finding the relations between novel and base categories, and (3) imbalance control from data distribution to alleviate the recognition bias towards the base categories. Our method is simple yet effective. The extensive experiments have shown that our model achieves state-of-the-art results on both LS-FSL and traditional FSL benchmarks. Besides, the proposed textual knowledge discovery can also be used to improve other LS-FSL methods.

Acknowledgments. This work is supported by the National Key Research and Development Program of China under grant 2018YFB0804205, and National Nature Science Foundation of China (NSFC) under grants 61732008 and 61725203.

[2] The SGM features are obtained from the released code by the authors of [7], while the KTCH features are provided by an author of [15].

References

1. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: COMPSTAT (2010)
2. Douze, M., Szlam, A., Hariharan, B., Jégou, H.: Low-shot learning with large-scale diffusion. In: CVPR (2018)
3. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: ICML (2017)
4. Gidaris, S., Komodakis, N.: Dynamic few-shot visual learning without forgetting. In: CVPR (2018)
5. Gidaris, S., Komodakis, N.: Generating classification weights with gnn denoising autoencoders for few-shot learning. In: CVPR (2019)
6. Guo, D., Wang, S., Tian, Q., Wang, M.: Dense temporal convolution network for sign language translation. In: IJCAI (2019)
7. Hariharan, B., Girshick, R.: Low-shot visual recognition by shrinking and hallucinating features. In: ICCV (2017)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
9. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. TPAMI (11) (1998)
10. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
11. Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. In: CVPR (2017)
12. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS (2012)
14. Lake, B.M., Salakhutdinov, R., Tenenbaum, J.B.: Human-level concept learning through probabilistic program induction. Science **350**(6266) (2015)
15. Li, A., Luo, T., Lu, Z., Xiang, T., Wang, L.: Large-scale few-shot learning: Knowledge transfer with class hierarchy. In: CVPR (2019)
16. Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.Y.: Learning to detect a salient object. TPAMI **33**(2) (2010)
17. Meng, N., Wu, X., Liu, J., Lam, E.Y.: High-order residual network for light field super-resolution. In: AAAI (2020)
18. Munkhdalai, T., Yu, H.: Meta networks. In: ICML (2017)
19. Peng, Z., Li, Z., Zhang, J., Li, Y., Qi, G.J., Tang, J.: Few-shot image recognition with knowledge transfer. In: ICCV (2019)
20. Qi, H., Brown, M., Lowe, D.G.: Low-shot learning with imprinted weights. In: CVPR (2018)
21. Qiao, S., Liu, C., Shen, W., Yuille, A.L.: Few-shot image recognition by predicting parameters from activations. In: CVPR (2018)
22. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: CVPR (2016)
23. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS (2015)
24. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV **115**(3) (2015)

25. Rusu, A.A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., Hadsell, R.: Meta-learning with latent embedding optimization. In: ICLR (2019)
26. Salimans, T., Kingma, D.P.: Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In: NeurIPS (2016)
27. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: ICCV (2017)
28. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: NeurIPS (2017)
29. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: CVPR (2018)
30. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: NeurIPS (2016)
31. Wang, S., Guo, D., Zhou, W.g., Zha, Z.J., Wang, M.: Connectionist temporal fusion for sign language translation. In: ACM MM (2018)
32. Wang, Y.X., Girshick, R., Hebert, M., Hariharan, B.: Low-shot learning from imaginary data. In: CVPR (2018)
33. Xian, Y., Sharma, S., Schiele, B., Akata, Z.: f-vaegan-d2: A feature generating framework for any-shot learning. In: CVPR (2019)
34. You, Q., Jin, H., Wang, Z., Fang, C., Luo, J.: Image captioning with semantic attention. In: CVPR (2016)
35. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: BMVC (2016)
36. Zhang, H., Zhang, J., Koniusz, P.: Few-shot learning via saliency-guided hallucination of samples. In: CVPR (2019)
37. Zhang, J., Zhang, T., Dai, Y., Harandi, M., Hartley, R.: Deep unsupervised saliency detection: A multiple noisy labeling perspective. In: CVPR (2018)
38. Zhang, R., Che, T., Ghahramani, Z., Bengio, Y., Song, Y.: Metagan: An adversarial approach to few-shot learning. In: NeurIPS (2018)